

ДЖОШ КАУФМАН

ПЕРВЫЕ 20 ЧАСОВ. КАК БЫСТРО НАУЧИТЬСЯ... ЧЕМУ УГОДНО



Посвящается Леле

Josh Kaufman

THE FIRST 20 HOURS

How to Learn Anything. . Fast

All rights reserved including the right of reproduction in whole or in part in any form.

This edition published by arrangement with Portfolio, a member of Penguin Group (USA)

Inc.

© Worldly Wisdom Ventures LLC, 2013

© Гольдберг Ю., перевод на русский язык, 2014

© Оформление, издание на русском языке. ООО «Издательская Группа «Азбука-Аттикус», 2014

К читателю

*Жизнь коротка, искусство — не постичь.
Джеффри Чосер, «Птичий парламент»*

«Я столько всего хочу... но у меня так мало времени». Вот история современной жизни.

Остановитесь на минуту и подумайте, чем бы вы хотели заняться. Что там у вас в списке? А что вам мешает исполнить свои желания?

Скорее всего, две вещи: время и отсутствие навыков.

Истина не очень приятная: самые интересные занятия в жизни всегда требуют определенного навыка. Для овладения им необходимо потратить время и приложить усилия — время, которого у нас нет, и усилия, которые мы не склонны прикладывать.

«Я займусь этим потом, когда будет свободная минутка».

Откровенно говоря, проще сидеть перед телевизором или бродить по Всемирной паутине, что и делает большинство людей, чьи желания по этой причине так и остаются в области грез.

А вот еще одна неприглядная истина: многие занятия не доставляют удовольствия, пока вы не добьетесь в них определенных успехов. Каждый навык имеет собственный, как я его называю, *барьер фрустрации* — период времени, когда у вас совсем ничего не выходит, и вы остро ощущаете этот факт. Зачем тратить время, если вы знаете, что не сможете этому как следует научиться?

Правда, было бы здорово, знай вы способ овладевать новыми навыками, не испытывая такого страха? Быстро преодолевать барьер фрустрации и переходить к тому этапу, который доставляет радость? Тратить меньше времени на утомительную борьбу с растерянностью и сомнениями, а больше — на удовольствие?

Возможно ли приобрести новые навыки без особых мучений, таким образом, чтобы расходовать меньше времени и сил?

Мой опыт свидетельствует: да, возможно.

Эта книга о моем личном стремлении постичь искусство и науку *быстрого приобретения навыков* — то есть как научиться чему-то новому настолько быстро, насколько это возможно. Цель книги — помочь вам овладеть новыми навыками за рекордно короткое время.

Я на собственном опыте убедился, что для преодоления барьера фрустрации — то есть от полного неумения делать то, чему вы пытаетесь научиться, до заметного прогресса — требуется около 20 часов.

Эта книга описывает системный подход к приобретению новых навыков настолько быстро, насколько это возможно. Предлагаемый метод универсален. Не имеет значения, хотите ли вы изучить иностранный язык, сочинить роман, нарисовать портрет, начать свой бизнес или пилотировать самолет. Потратив всего 20 часов на обучение основам навыка, вы будете удивлены достигнутым успехом.

Мы начнем с принципов быстрого приобретения навыка: как овладеть новым делом за максимально короткое время. Теория и практика не очень сложны, и обучение не займет много времени.

Затем я объясню, как применять эти принципы в реальном мире, продемонстрировав, каким образом я приобрел шесть новых навыков, потратив на каждый 20 часов или даже меньше, тренируясь не более полутора часов в день.

- Разработка индивидуального комплекса упражнений йоги.
- Разработка компьютерной программы для интернет-приложения.
- Переучивание печатанию на компьютере слепым методом.
- Изучение древней сложной игры.
- Игра на музыкальном инструменте.
- Виндсерфинг.

Надеюсь, эта книга вдохновит вас стряхнуть пыль со старого списка ваших «желаний», пересмотреть его и приступить к изучению чего-то нового.

Джош Кауфман

1

Портрет автора как фаната обучения

Каждое утро я встаю с намерением изменить мир и прекрасно провести время. Иногда из-за этого мне трудно спланировать день.

Э. Б. Уайт, писатель и эссеист, автор книг «Паутинка Шарлотты» и «Элементы стиля»

Меня зовут Джош Кауфман, и я заядлый любитель обучения.

Шкафы у меня дома и в офисе до отказа набиты книгами, инструментами и разного рода неиспользуемым оборудованием, которое медленно покрывается слоем пыли.

Мой список того, чему необходимо научиться, включает сотни пунктов. Моя корзина покупок на сайте Amazon.com в настоящее время содержит 241 наименование — все это книги, которые я бы хотел прочитать. Зайдя в книжный магазин, я не могу выйти оттуда без трех или четырех новых книг, которые дополнят те 852 тома, что у меня уже есть.

Каждый день ко мне приходит идея очередного проекта или эксперимента, и я добавляю ее в постоянно удлиняющийся перечень под названием «когда-нибудь/возможно». При взгляде на список всего, чему я хотел бы научиться, меня охватывает растерянность, поэтому я делаю это не очень часто.

Я хочу знать, как повысить эффективность моего издательского бизнеса. Я хочу научиться снимать и редактировать видео. Я хочу подготовить аудиопрограмму. Я хочу понять, как улучшить мои семинары и сделать мои курсы эффективнее.

У меня есть идеи по выпуску нового продукта, но я не знаю, как его создать. Я придумал новые компьютерные программы, но не умею их писать. Сюжетов книг в моей голове столько, что у меня не хватит на них ни времени, ни сил.

Я хочу научиться рисовать. Научиться сплавиться по реке на байдарке. Удиль рыбу нахлестом. Я хочу научиться скалолазанию, хочу уметь играть на гитаре, укулеле, пианино и электрической скрипке.

Мне нравятся некоторые игры, например го, но я так и не научился в них играть. Правила других игр, в частности шахмат, мне известны, однако я играю в них не очень хорошо и поэтому — не получая большого удовольствия — довольно редко.

Меня привлекает гольф, но каждая партия оборачивается тренировкой стойкости духа, когда мне приходится за смехом скрывать свое замешательство.

(Обычно я говорю, что играю в гольф-марафон: к концу партии пробегаю марафонскую дистанцию.)

У меня складывается впечатление, что каждый день я добавляю какой-нибудь новый навык к списку того, что мне хотелось бы уметь, *ad infinitum*[1]. Столькому нужно еще научиться, а времени так мало!

По складу характера я отношусь к тем, кто все делает своими руками. Если требуется решить какую-то проблему, я скорее займусь этим сам, чем обращусь за помощью. Даже если кто-то справится с этим быстрее и лучше, мне не хочется лишать себя удовольствия чему-нибудь научиться.

Усложняет ситуацию и то обстоятельство, что у моей жены Келси свой бизнес — разработка курсов непрерывного повышения квалификации для преподавателей йоги. Мы оба с удовольствием занимаемся бизнесом, и поэтому постоянно заняты.

А чтобы сделать жизнь еще интереснее, мы подарили этому миру нашу дочь Лелу. Когда я пишу эти строки, Леле девять месяцев.

Еще до рождения Лелы мы с Келси решили, что если у нас будут дети, то их воспитание должно стать для нас самым главным делом. Одной из основных причин, по которым я оставил свою работу менеджера в корпорации Fortune 500, была потребность в гибком графике, желание работать дома, и самому распоряжаться своим временем и как можно больше этого самого времени проводить с семьей.

Мы с Келси поровну делим родительские обязанности. Поскольку в нашей семье два бизнеса, то Келси работает по утрам, я же занимаюсь Лелой. После обеда дочь переходит на попечение Келси, а я работаю до ужина. Это дает мне 25 рабочих часов в неделю плюс то время, которое мне удастся выкроить, пока Лела спит.

После рождения Лелы я едва успевал справляться с работой, а о приобретении новых навыков не могло быть и речи. Заядлого любителя обучения это могло свести с ума. Я не хотел полностью отказываться от учебы и возможности личностного роста даже после появления новых семейных обязанностей. Свободного времени у меня не очень много, но я собирался распорядиться им с максимальной разумностью.

Так возник интерес к тому, что я называю быстрым приобретением навыка, — методам быстрого обучения новым умениям.

Я хочу и дальше учиться чему-то новому, но не желаю, чтобы процесс обучения затягивался до бесконечности. Я хочу быстро усваивать основы и добиваться заметных результатов, не испытывая постоянной фрустрации.

Уверен, вы меня поймете. Сколько «свободного» времени у вас остается ежедневно после того, как закончена работа и выполнены семейные обязанности? Вам не кажется, что в сутках должно быть 36 или 48 часов, чтобы у вас оставалось время на то, чтобы сесть и научиться чему-то новому?

Есть известное выражение: «работать с умом». Выяснилось, что процесс приобретения нового навыка имеет отношение не к количеству затраченных часов, а к тому, на что вы тратите эти часы.

Будь ты проклят, Малкольм Гладуэлл!

В 2008 году Малкольм Гладуэлл написал книгу под названием «Гении и аутсайдеры. Почему одним все, а другим ничего?»[2] («Outliers: The Story of Success»). В ней автор попытался объяснить, почему одни люди оказываются успешнее других.

Одна из идей, к которой Гладуэлл обращается снова и снова, называется «правилом 10 тысяч часов». Исследование, проведенное доктором К. Андерсом Эриксоном из Университета штата Флорида, показало, что для достижения высокого уровня мастерства требуется в среднем 10 тысяч часов целенаправленных тренировок [1].

10 тысяч часов эквивалентны восьми часам ежедневных занятий на протяжении приблизительно трех с половиной лет, без перерывов, выходных и отпусков. Если предположить, что в году 260 полноценных рабочих дней, то это полная занятость в течение пяти лет, при условии, что во все 100 % времени вы вкладываете все 100 % своей энергии и сил.

На практике такой уровень сосредоточенности требует чрезвычайного напряжения. Даже у мастеров мирового класса в высшей степени конкурентных профессий (например, у музыкантов и спортсменов) хватает сил лишь приблизительно на три с половиной часа интенсивных ежедневных занятий. Это значит, что для достижения мастерства может потребоваться 10 лет.

Другими словами, исследования доктора Эриксона указывают, что, если вы хотите приобрести новый навык, приготовьтесь к долгому пути. Чтобы стать в какой-либо области лучшим в мире, даже ненадолго, потребуются годы и годы упорных тренировок. Если же вы не желаете тратить столько времени и сил, вас затмят те, кто готов на это.

Книга «Гении и аутсайдеры» стремительно взлетела в верхние строчки списков бестселлеров из категории научно-популярной литературы и продержалась там три месяца. В мгновение ока «правило 10 тысяч часов» стало постулатом.

Неужели вы не только должны найти время для тренировок... но еще и обязаны потратить на них именно 10 тысяч часов? Большинство считают себя счастливыми, если удастся выкроить несколько часов в неделю. Зачем вообще заваривать всю эту кашу, если для обучения требуется столько времени?

Взгляните на мои великие деянья![3]

Прежде чем отступить, подумайте вот о чем.

В исследовании доктора Эриксона есть один аспект, который очень легко не заметить: это высокий уровень мастерства. Если вы собираетесь стать следующим Тайгером Вудсом, то вам, вероятно, придется потратить не менее 10 тысяч часов на интенсивную и систематическую отработку всех элементов игры в гольф. Почти все профессиональные гольфисты начинали играть в юном возрасте и непрерывно тренировались на протяжении как минимум семи лет. Чтобы стать игроком мирового класса, требуется время.

А если вы не ставите перед собой цель выиграть турнир Профессиональной ассоциации гольфа? Если вы просто хотите научиться играть, чтобы достойно выглядеть на поле, не испытывать неловкости, получать удовольствие и, возможно, иметь шансы на успех в местном гольф-клубе?

Это совсем другое дело. Мастерство мирового класса может потребовать 10 тысяч часов упорных тренировок, но для овладения навыком, *достаточным для удовлетворения своих потребностей*, как правило, не нужно прилагать такие колоссальные усилия.

Данное утверждение не ставит под сомнение ценность того, что Эриксон называет «намеренной практикой»: интенсивные и систематические упражнения по совершенствованию навыка. Намеренная практика — это основа приобретения навыка. Вопрос в том, *сколько* намеренной практики требуется для достижения поставленной цели. Как правило, гораздо меньше, чем вам кажется.

Качество, а не количество

Ключом к быстрому приобретению навыка является принцип *достаточности*. В данной книге мы будем обсуждать умение что-либо делать, а не мастерство мирового класса. Мы попробуем выйти на крутой участок кривой обучения и как можно быстрее подняться по ней.

Оставьте 10 тысяч часов профессионалам. Мы начнем с 20 часов целенаправленных, разумных и сосредоточенных усилий.

Мы хотим добиться требуемых результатов с минимальными затратами сил. Возможно, вы никогда не выиграете золотую медаль чемпионата мира, но получите не менее ценные для вас награды за гораздо меньшее время.

Если потом вы захотите усовершенствовать свои умения, то значительно повысите шансы на успех, начав с быстрого приобретения навыков. Осознав, во что вы ввязываетесь, усвоив основы, тренируясь разумно и выработав удобный

график тренировок, вы будете быстрее и увереннее продвигаться по кривой обучения и в рекордно короткое время достигнете мастерства.

Что такое быстрое приобретение навыка?

Быстрое приобретение навыка — это процесс. Вы разбиваете навык, которым намерены овладеть, на минимально возможные элементы, причем выявление этих элементов является самым главным, а затем целенаправленно упражняетесь в их выполнении. Все просто.

Быстрое приобретение навыка можно представить в виде четырех основных этапов.

- **Разбиение** навыка на элементы, мельчайшие из возможных.
- **Изучение** каждого элемента — достаточно подробное, чтобы придать тренировкам осмысленность и иметь возможность исправления ошибок.
- **Устранение** физических, психологических и эмоциональных препятствий, мешающих тренировкам.
- **Тренировка** самых важных элементов в течение как минимум 20 часов.

Вот так. Быстрое приобретение навыка — это не ракетные технологии. Вы просто выбираете, что именно тренировать, определяете наилучший метод тренировки, выделяете на нее время и упражняетесь, пока не достигнете желаемого уровня мастерства.

Тут нет никакой магии — только разумные, долговременные усилия, направленные на то, что вам интересно. Небольшая подготовка позволит вам овладевать новыми навыками быстро и с меньшими усилиями.

Это не значит, что результат будет мгновенным. Кстати, стремление к мгновенному удовлетворению — одна из главных причин, почему люди так медленно приобретают новые навыки.

Заблуждение «Матрицы»

Помните сцену из фильма «Матрица», когда герой Киану Ривза открывает глаза, моргает несколько раз и шепчет: «Я знаю кун-фу»?

Вынужден вас разочаровать: быстрое приобретение навыка не *настолько* быстрое.

В этом Голливуд оказал нам медвежью услугу. Разумеется, было бы здорово научиться управлять вертолетом Bell 212 за пять секунд, загрузив программное обеспечение прямо в мозг, но современная наука не может

угнаться за фантастикой.

До тех пор пока программирование мозга не станет реальностью, «быстрый» означает лишь существенно меньшее время, чем требуется для обучения навыку так, как это обычно делает большинство людей: вслепую, бессистемно и непоследовательно.

Один из первых профессиональных навыков, которым я овладел, это умение разработать полезный и работоспособный сайт. Начав в 1996 году с основ на сайте Angelfire.com, я научился читать и писать на языке HTML, использовать CSS, редактировать изображение в Adobe Photoshop, конфигурировать интернет-серверы и поддерживать систему, которая публикует мои работы.

Я не учился всему этому в старших классах школы или в колледже. И хотя в университете нам читали базовый курс по информационным системам в бизнесе, лекции в аудитории и моя нынешняя повседневная работа практически не пересекаются.

Я овладел искусством создания сайтов методом проб и ошибок, обучаясь по ходу дела. Столкнувшись с новой технологией или инструментом, способным улучшить мой сайт или облегчить работу, я экспериментировал. Прошло довольно много времени, прежде чем мое мастерство выросло.

Тем не менее даже бессистемный подход к приобретению навыка создания сайтов сделал свое дело: я получил заказ, требовавший этих навыков, и теперь они, уже приобретенные, помогают мне зарабатывать на жизнь. Можно сказать, цель достигнута.

Все это далось мне тяжелым трудом. Вне всякого сомнения, моего уровня компетенции в данной области можно достичь не за 15 лет, а гораздо быстрее, если применить системный подход. Разумно спланировав занятия, вы подниметесь до моего уровня компетентности примерно за месяц.

Вот что я имею в виду под быстрым приобретением навыка. Если вы сможете узнать о веб-дизайне столько же, сколько знаю я, не за 15 лет, а за один месяц сосредоточенных усилий, это серьезный прогресс. И это вполне достижимо.

Быстрота приобретения нового навыка в значительной степени определяется тем, сколько времени вы готовы посвящать намеренной практике и разумному экспериментированию и насколько высоки требования к желаемому уровню мастерства.

Не ждите мгновенных результатов. Просто рассчитывайте, что затраченное время будет гораздо, гораздо меньшим, чем если приступать к процессу обучения, не имея стратегии.

Прежде чем подробно рассматривать сам метод, необходимо осознать следующее: быстрое приобретение навыка не имеет ничего общего с тем, как вас «учили учиться» в школе. Теоретические занятия и аттестация практически

никак не пересекаются с приобретением навыка, не говоря уже о его быстром приобретении.

Приобретение навыка и обучение

Подобно многим американским старшеклассникам, я изучал иностранный язык. Каждый день на протяжении нескольких лет мне преподавали испанский. Оценки у меня были высокими — в основном пятерки.

Сегодня я могу вспомнить только *hola, ¿cúmo estás* и *muuy bien*^[4] и, хоть убей, не способен поддержать разговор по-испански. (Я даже не знаю, как сказать, что у меня выдался не самый удачный день.)

Противоположный пример — мой друг Карлос Мисели, выросший в испаноязычной Аргентине. В старших классах школы Карлос решил, что ему нужно научиться бегло говорить по-английски, и сознательно старался как можно больше разговаривать с людьми, для кого английский был родным. В процессе освоения языка он открыл для себя Skype и создал собственный сайт, чтобы иметь возможность регулярно практиковаться в письменном и устном английском.

Карлос не учился на курсах. Он не знает правил английской грамматики. И даже не в состоянии объяснить, *как* выучил язык. Но это не имеет значения. Карлос бегло говорит и пишет по-английски, а остальное не важно.

Доктор Стивен Крашен из Университета Южной Калифорнии — признанный специалист в области обучения второму языку. Одна из главных идей Крашена заключается в том, что *усвоение* языка отличается от *обучения* языку.

В школе я много узнал *об* испанском языке. Выучил тысячи испанских слов, спряжение глаголов, правила грамматики. Усвоил все необходимое, чтобы с блеском сдать экзамены.

Однако эти экзамены не имели никакого отношения к моей способности осмысленно говорить по-испански, а также понимать естественный темп речи испаноговорящего человека. Если бы я хотел свободно владеть языком, то несколько недель разговоров на испанском дали бы лучший результат, чем четыре года учебы.

Однако в то время я не ставил себе целью свободное владение испанским. Моей целью было получить высший балл на выпускном экзамене. Карлос, наоборот, не утруждал себя теорией, а начал сразу с практики. Вместо спряжения глаголов он упражнялся в том, что действительно важно, — общаться с другими людьми на английском.

Бесспорно, в смысле эффективности и долговременного результата подход Карлоса оказался гораздо лучше моего.

Истинная ценность обучения

Это не значит, что вам не нужны теоретические знания о навыке, который вы приобретаете. Обучение может быть чрезвычайно важным, но несколько в другом, неожиданном аспекте. Теоретические знания о предмете обучения помогают «редактировать» или «исправлять ошибки» в процессе тренировки.

Если вы умеете спрягать испанские глаголы, вам будет легче составить правильную фразу в разговоре с человеком, для которого испанский язык родной. Зная слова из основного словаря, вы будете лучше понимать собеседника, а попав в затруднительное положение, быстрее подберете подходящее слово или фразу.

Доктор Крашен называет это «гипотезой монитора». Обучение помогает планировать, редактировать и исправлять ошибки в процессе овладения навыком. Вот в чем заключается ценность обучения. Беда в том, что мы путаем обучение с приобретением навыка.

Если вы хотите приобрести новый навык, необходимо тренироваться в соответствующем контексте. Обучение делает практику эффективнее, но не заменяет ее. Если речь идет о практических действиях, одного обучения недостаточно.

Приобретение навыка и тренировка

Существует также огромная разница между приобретением навыка и тренировкой. В данном контексте тренировка означает совершенствование навыка, которым вы уже овладели посредством повторения.

Примером может служить марафонский бег. Большинство людей учатся бегать в детстве. Тут никакого нового навыка не нужно — ставим попеременно одну ногу впереди другой, пока не преодолеем все 42 километра.

Однако сначала требуется приложить довольно много сил, чтобы укрепить свой организм и достичь необходимой физической готовности, позволяющей пробежать марафон. Этот процесс выработки выносливости и называется тренировкой. Чем больше вы тренируетесь, тем сильнее становитесь — и тем быстрее пробегаете марафон.

В занятиях марафонским бегом присутствует и элемент обучения: как записаться для участия в забеге, как квалифицироваться на крупные соревнования, например Бостонский марафон, чего ожидать на дистанции, какими должны быть оптимальный темп бега, экипировка и так далее.

Например, такая мелочь, как трение футболки о кожу, не будет большой проблемой, если вы бежите на 5 тысяч метров, и большинство бегунов об этом

даже не задумываются. Но чрезмерное трение превращается в *огромную* проблему, если дистанция — марафон.

Не сумев подготовиться заранее, вы столкнетесь с такой печально известной проблемой, как стертые в кровь соски. Это неприятно, больно... и вполне предотвратимо. (Не верите? Поищите в Google.)

Тренировка и обучение, несомненно, помогут вам добежать до финиша, но это не приобретение навыка, без которого тренировка и обучение невозможны или бесполезны. Подготовка и создание благоприятных условий для занятий могут облегчить приобретение навыка, но ни в коем случае не заменяют практику.

А вот повторное обучение бегу *относится* к категории приобретения навыка. Такие методики, как ChiRunning [2], помогают бегуну овладеть техникой, которая минимизирует усилия и повышает эффективность движений. Немного попрактиковавшись, спортсмен может обновить основной навык бега, который затем укрепляется последующими тренировками.

Приобретение навыка и образование с аттестацией

Несмотря на благородные усилия преподавателей и профессоров во всем мире, современные методы в сфере образования и аттестации почти никак не связаны с приобретением навыка.

Приобретение навыка требует практики. Оно требует значительных периодов непрерывных, сфокусированных усилий. Оно требует творческого подхода, гибкости и свободы устанавливать собственное мерило успеха.

К сожалению, большинство современных методов в области образования и аттестации требуют простого соответствия нормам. Главная (по умолчанию) цель состоит в удостоверении соответствия набору критериев, по большей части случайных, которые устанавливаются комиссиями с целью подтверждения определенных качеств, которые якобы требуются некой третьей стороне.

Творческий подход, гибкость и свобода эксперимента — главные элементы быстрого приобретения навыка — несовместимы с процессом аттестации. Если стандарты слишком гибкие, их уже нельзя назвать стандартами, не правда ли?

К сожалению, жесткие процедуры в области образования и аттестации могут *мешать* приобретению навыка. Главная проблема заключается в цене нереализованных возможностей: если требования по получению аттестации настолько высоки, что не оставляют времени на практические занятия, то программы аттестации могут принести больше вреда, чем пользы.

Возьмем, к примеру, энергичного и мотивированного человека, желающего создать компанию по разработке программного обеспечения.

Базовый курс информатики и вычислительной техники престижного университета обычно рассчитан как минимум на четыре года [3]. За это время свежееиспеченный специалист проведет тысячи часов за изучением алгоритмов и анализом компиляторов, успешно сдаст дюжину экзаменов, но ни на шаг не приблизится к созданию собственной компании. Наш несчастный студент запомнит большой объем информации о программировании, но по-прежнему не будет знать, как создать именно такую компьютерную программу, которую люди посчитают полезной и захотят купить.

Создание компании по разработке программного обеспечения требует приобретения новых навыков. Это изучение языков программирования, создание и поддержка компьютерных систем, анализ доступных инструментов и программ, создание прототипов, поиск первых пользователей, получение необходимого финансирования, а также решение обычных задач управления бизнесом.

Есть ли связь между созданием стартапа и получением свидетельства об образовании? Несомненно. Но следует отметить вот что: бо2льшая часть образовательного процесса направлена на выполнение определенных стандартов. Помогут ли эти стандарты приобрести навыки, необходимые для работы в реальном мире? Этот вопрос для разработчиков стандартов третьестепенен — в лучшем случае.

В моей первой книге «Сам себе MBA. Самообразование на 100 %»[5] («The Personal MBA: Master the Art of Business») (2010) я объяснял, почему решил отказаться от диплома о бизнес-образовании в пользу самостоятельного изучения современных принципов ведения бизнеса и создания собственной компании. Заменяв занятия в бизнес-школе построением собственного бизнеса, я очень многому научился и сэкономил 150 тысяч долларов. С учетом поставленных целей самостоятельное приобретение навыков построения бизнеса оказалось во всех отношениях предпочтительнее бизнес-школы.

Если вы хотите добиться успехов в том, что требует эффективности в реальной жизни, необходимо практиковать этот навык в соответствующем контексте. Одного теоретического обучения никогда не бывает достаточно.

Нейрофизиология навыка: пластичность мозга и мышечная память

И последнее, что следует знать, прежде чем мы перейдем к сути быстрого приобретения навыка: вы должны полностью осознавать свою способность овладеть новыми навыками.

Как ни странно это звучит, но человеку легко поверить в ограниченность своих возможностей, в то, что вам от природы либо даны способности, талант, дар, либо, увы, не даны.

В книге «Образ мышления: новая психология успеха» («Mindset: The New Psychology of Success») (2007) психолог Кэрол Двек приводит результаты исследований, которые указывают на то, что люди обычно придерживаются одного из двух взглядов на работу своего мозга.

По мнению доктора Двек, приверженцы «фиксированной» теории предполагают, что способности и таланты являются врожденными, что мы появляемся на свет с определенными возможностями и изменить ничего нельзя. Если человеку с таким типом мышления «не дается математика», то дополнительные занятия математикой — пустая трата времени. К чему утруждать себя, если ты все равно не добьешься успеха?

Те же, кто придерживается теории «роста», предполагают, что навыки и возможности расширяются посредством практики и настойчивых усилий. Ошибаясь в решении математических задач, человек с таким складом ума считает причиной неуспеха не отсутствие способностей, а недостаток практики. Если проявить настойчивость и упорно трудиться, овладение математикой на нужном уровне — это лишь вопрос времени.

Для тех, кто попал в ловушку «фиксированного» мышления, у меня есть хорошая новость: многие исследования (и их число постоянно растет) демонстрируют, что любой мозг способен при помощи упражнений совершенствовать свои способности и навыки. Генетическая предрасположенность действительно существует, однако она играет незначительную роль в сравнении с мощью целенаправленной, разумной практики. Вы можете развить любой навык, если захотите приложить усилия.

Человеческий мозг *пластичен* — этот термин нейробиологи используют для указания на физические изменения, которые происходят в мозге под воздействием окружающей среды, ваших действий и их последствий. В процессе практики при обучении новому навыку, двигательному или интеллектуальному, между нейронами мозга изменяются связи.

По словам доктора Джона Медины, автора книги «Правила развития мозга вашего ребенка» («Brain Rules For Baby», 2009), «одновременно возбуждаемые нейроны образуют связи», и в результате в нервных цепях мозга формируются абсолютно новые структуры. Со временем в процессе упражнений в ответ на обратную связь возбуждаемые нейроны образуют все более эффективные цепи.

Если вы работаете над *моторным навыком* (то есть навыком, который предполагает физическое движение), то в самом начале действие получается медленным и неуклюжим. Вам приходится думать над каждым движением, и вы часто совершаете досадные ошибки. Для овладения основами требуются целенаправленные усилия.

Однако со временем мышечная координация становится автоматической, более согласованной с мыслительным процессом. У вас появляется способность

уделять больше внимания мелочам, и вы учитесь адаптировать свои действия в соответствии с обратной связью, поступающей от окружающей среды.

Вы начинаете уделять больше времени тому, что важно, не отвлекаясь на второстепенное. В конечном итоге вы уже способны выполнять требуемое действие, не сосредоточиваясь на каждой мелочи.

В научной литературе этот процесс называется «трехступенчатой моделью» приобретения навыка [4], которая применима и к двигательным, и к интеллектуальным навыкам. Вот эти три стадии.

1. Когнитивная (первая) стадия — понимание того, что вы пытаетесь сделать, осмысление процесса и разбиение навыка на поддающиеся выполнению элементы.

2. Ассоциативная (промежуточная) стадия — практика, восприятие обратной связи и корректировка действий на основе этой связи.

3. Автономная (завершающая) стадия — эффективное и рациональное выполнение действия, без обдумывания и излишнего внимания к процессу.

Нейрофизиологический процесс приобретения навыка протекает непрерывно, даже когда вы читаете это предложение. Разум не может пребывать в неподвижном состоянии. Ваш мозг постоянно учится, декодирует информацию и объединяет новые навыки.

Доктор Двек в книге «Новая психология успеха» пишет: «Ваш мозг подобен мышце — чем больше вы его нагружаете, тем сильнее он становится». Чем больше вы упражняетесь, тем более рациональным, эффективным и автоматическим становится навык.

Это очень важно, когда речь заходит о быстром приобретении навыка. Если ваши разум и тело способны научиться действовать иначе и лучше, то можно выяснить, как *ускорить* этот процесс.

2

Принципы быстрого приобретения навыка

Я понял, что для того, чтобы стать мастером карате, не нужно выучивать 4000 приемов, надо просто повторить 4000 раз каждое из базовых движений.

Чет Холмс, автор книги «Невероятная машина продаж»^[6] («The Ultimate Sales Machine»)

Теперь, когда мы поняли, что на самом деле означает приобретение навыка, давайте выясним, как сделать это быстро. Цель данной главы — предложить удобную пошаговую инструкцию по приобретению нового навыка.

Я считаю, что к этим принципам следует относиться как к способу культивирования «временной одержимости». Быстрое приобретение навыка происходит естественным образом, когда ваше любопытство и интерес перевесят все сомнения, по крайней мере временно.

Рассматривайте эти принципы как пути идентификации навыка, заслуживающего временной одержимости, сосредоточения на нем и устранения отвлекающих моментов или препятствий, которые мешают эффективной практике.

Вот 10 главных принципов быстрого приобретения навыка.

1. Выберите привлекательный проект.
2. Сосредоточьтесь на каком-то одном навыке.
3. Определите целевой уровень мастерства.
4. Разбейте навык на элементы.
5. Приготовьте все необходимое для занятий.
6. Устраните препятствия для занятий.
7. Выделите специальное время для занятий.
8. Создайте быстрые петли обратной связи.
9. Занимайтесь по расписанию, короткими интенсивными интервалами.
10. Уделяйте внимание количеству и скорости.

Многие из этих принципов могут быть выведены просто исходя из здравого смысла — и это действительно так. Важно помнить: простого знания этих принципов недостаточно. Вы должны применять их на практике, если хотите сорвать куш.

1. Выберите привлекательный проект

Карл Поппер считается одним из величайших философов XX века. Именно он ввел в обиход понятие *фальсифицируемости научной теории*. Выражаясь простым языком, если вы не в состоянии опровергнуть что-либо путем наблюдения или эксперимента, значит, это не наука.

Поппер сказал много умных вещей, но одно из самых мудрых, по-моему, следующее его утверждение: «Лучшее, что может случиться с человеческим существом, — это найти проблему, влюбиться в нее и жить, пытаясь решить ее, если только не появится другая проблема, еще более привлекательная».

Если вам нужна формула, которая поможет прожить насыщенную, плодотворную жизнь, воспользуйтесь этой — не ошибетесь.

Быстрое приобретение навыка требует выбора привлекательной проблемы или проекта. Чем сильнее вас вдохновляет какое-либо умение, тем быстрее вы им овладеете.

На практике поиск привлекательного проекта — дело сугубо индивидуальное. Например, в моем нынешнем списке навыков, которыми я хочу овладеть, отсутствует умение говорить и писать на пекинском диалекте китайского языка, поскольку в данный момент у меня нет в этом настоящей потребности, но есть множество других интересных проектов. Если мне придет в голову переселиться в ту часть Китая, где говорят на пекинском диалекте, это умение может стать для меня привлекательным — но пока я никуда ехать не собираюсь.

С другой стороны, мне очень хочется научиться играть в го, древнейшую в мире стратегическую настольную игру, которую придумали в Китае больше трех тысяч лет назад. Это прекрасная игра, и желание научиться ей возникло у меня еще много лет назад, при первом знакомстве.

Чтобы научиться играть в го, нужен серьезный подход. Правила просты, но точное прочтение узоров из чередующихся белых и черных камешков на доске — это настоящее искусство. В шахматном мире уже много лет доминируют компьютеры, но даже самые мощные вычислительные машины не могут тягаться с опытным игроком в го.

Тому, что для вас важно, вы естественным образом учитесь быстрее. В настоящее время мне интереснее научиться играть в го, чем овладеть пекинским диалектом китайского языка, и поэтому я сначала займусь го, а китайский оставлю на потом.

Если сначала вы сфокусируетесь на овладении *самым важным* для вас навыком (то есть самым привлекательным проектом), то освоите его гораздо быстрее.

2. Сосредоточьтесь на каком-то одном навыке

Одна из первых ошибок в процессе приобретения новых навыков — это попытка овладеть несколькими навыками одновременно.

Арифметика тут проста: приобретение нового навыка требует критической массы затраченного времени и сфокусированного внимания. Если для занятий и обучения у вас есть всего лишь час или два в день и вы распределите свое время и силы на 20 разных навыков, то каждому достанется слишком мало, чтобы добиться заметных успехов.

Одним людям применять этот принцип сложнее, другим легче. Лично у меня всегда был темперамент «человека Возрождения»: в любой момент времени я хочу научиться сотням вещей в сотнях самых разных областей. Поэтому мне трудно отказаться от изучения всего нового, о чем я узнаю или слышу [1].

Однако, когда я пытаюсь научиться всему сразу, из этого ничего не выходит. Никакого прогресса не наблюдается, и я трачу слишком много времени на переключение между разными занятиями, расстраиваюсь и беру за что-то другое. Таков рецепт для чрезвычайно медленного приобретения навыков.

Выберите один и только один новый навык, которым желаете овладеть. Сконцентрируйте всю свою свободную энергию на приобретении этого навыка, а остальные временно отложите. Дэвид Аллен, автор книги «Как привести дела в порядок» [7] («Getting Things Done») (2002), рекомендует завести список, который он называет «может быть, когда-нибудь»: список дел, которыми вы хотели бы заняться когда-нибудь в будущем, но которые недостаточно важны, чтобы сейчас сосредоточиваться на них. Добавляя пункт в этот список, вы на время избавляете себя от необходимости осуществлять или обдумывать идею, пока не решите присвоить ей статус востребованной.

Я могу повторять это бесконечно. Сосредоточение на одном главном деле является абсолютно необходимым условием для быстрого приобретения навыка. Вы не отказываетесь навсегда от других занятий — просто откладываете их на будущее.

3. Определите целевой уровень мастерства

Целевой уровень мастерства — это простая фраза определяет, что значит «достаточно хорошо». Насколько хорошо вы хотели бы овладеть навыком, над которым работаете?

Ваш целевой уровень мастерства — это краткое описание желаемого уровня овладения навыком. Рассматривайте его как формулировку (одной фразой) того, чего вы хотите достичь, и того, что вы будете уметь, когда завершите занятия. Чем конкретнее сформулирован целевой уровень, тем лучше.

Определение целевого уровня мастерства поможет вам представить, как выглядит определенный уровень владения навыком. Точное описание того, насколько хорошо вы хотите или должны освоить навык, укажет путь к достижению цели. Как говорил Чарльз Кеттеринг, изобретатель электрической системы зажигания автомобиля, «правильно сформулированная проблема — это наполовину решенная проблема».

Определение целевого уровня мастерства зависит в первую очередь от того, почему вы решили овладеть этим навыком. Если вы хотите получить удовольствие, то ваша цель — тот момент, когда вы перестанете раздражаться и начнете получать удовольствие от самого занятия. Если ваша цель — мастерство, то какой минимальный уровень мастерства вас устроит на первых порах?

Определив начальный целевой уровень, вы всегда можете затем повысить планку и продолжить занятия. Самые высокие целевые уровни кажутся недостижимыми, однако их достижение не выходит за границы возможного.

Как правило, чем ниже целевой уровень мастерства, тем быстрее вы можете приобрести соответствующий навык. Если вы нацелились на достижения мирового класса, такой прием может вам показаться жульничеством: вы просто опускаете планку, чтобы быстрее «победить», правда?

Совершенно верно, только это не жульничество. Не забывайте, что цель быстрого приобретения навыка — вовсе не результаты мирового класса. Мы стремимся не к совершенству, а к максимально быстрому достижению эффективности и результативности.

Важно отметить, что некоторые занятия связаны с риском, и поэтому в целевой уровень мастерства обязательно следует включать требования безопасности. Травма (или смерть) в процессе овладения новым навыком лишает ваши занятия смысла.

4. Разбейте навык на элементы

Большинство из того, что мы считаем навыками, на самом деле состоит из более мелких элементов. Выбрав навык, на котором вы хотите сосредоточиться, разбейте его на минимально возможные составляющие. Например, умение играть в гольф состоит из множества компонентов: выбора подходящей клюшки, первого удара с колышка, удара из рва с песком, заведения мяча в лунку и так далее.

После того как навык в должной мере разбит на составляющие, проще выделить самый важный элемент. Сосредоточившись сначала на главных элементах, вы добьетесь большего прогресса и затратите меньше сил.

Разбиение навыка на элементы также помогает избежать чувства

растерянности. Вам не обязательно отрабатывать все элементы навыка одновременно. Эффективнее будет сосредоточиться на тех составляющих, которые обещают наибольшую общую отдачу.

Кроме того, разбиение навыка на элементы до начала занятий позволяет выявить те составляющие, которые не имеют большого значения для новичков. Исключив необязательные элементы или приемы из первой стадии процесса, вы получите возможность вкладывать больше времени и сил в овладение первостепенными составляющими навыка.

5. Приготовьте все необходимое для занятий

Большинство навыков требуют определенной предварительной подготовки. Сложно играть в теннис без теннисной ракетки или научиться управлять вертолетом, если у вас нет доступа на аэродром.

Какие инструменты, оборудование и окружающая обстановка обеспечат вам эффективные занятия? Как получить самое лучшее из того, что вы можете найти и себе позволить?

Перед началом занятий не поленитесь определить критически важное оборудование — это сэкономит драгоценное время. Обеспечив себя необходимыми ресурсами до начала занятий, вы потратите время с максимальной пользой.

6. Устраните препятствия для занятий

На пути к цели вы можете столкнуться с множеством препятствий, которые затрудняют приобретение нового навыка. Вот несколько самых распространенных примеров.

— Приложение значительных усилий до начала занятий. Вы неправильно расположили инструменты, неверно их выбрали или не выполнили условия предварительной подготовки.

— Непостоянная доступность ресурсов. Например, вы берете оборудование напрокат или ваш ресурс имеет ограниченное время доступа.

— Внешние отвлекающие факторы. Это могут быть телевизор, телефонные звонки, электронные письма.

— Эмоциональные барьеры. Страх, сомнения и растерянность.

Каждое из этих препятствий затрудняет начало занятий и, как следствие, снижает скорость достижения результата.

Расчет на то, что одна лишь сила воли позволит вам преодолеть эти барьеры, — заведомо проигрышная стратегия. Ежедневные запасы воли

ограничены, и лучше использовать ее разумно.

Оптимальный способ применения силы воли для приобретения нового навыка — устранение с ее помощью препятствий, мешающих занятиям. Организовав свое окружение таким образом, чтобы максимально облегчить начало занятий, вы уменьшите время приобретения навыка.

7. Выделите специальное время для занятий

Время, которое вы тратите на приобретение нового навыка, нужно откуда-то брать. К сожалению, обычно мы хотим овладеть новыми навыками и одновременно не отказываться от того, что доставляет нам удовольствие, например просмотра телепрограмм и так далее.

Мы говорим себе: я займусь этим, когда найду время.

Но дело в том, что «найти» время невозможно — это миф.

Никто никогда чудесным образом не «находит» время, подобно тому как мы случайно находим 20-долларовую купюру, забытую в кармане куртки.

Если вы надеетесь, что найдете время для какого-то дела, оно так и останется несделанным. Если вам требуется время, вы должны его *создать*.

Каждый день вы можете использовать 24 часа, или 1440 минут — не больше, и не меньше. Это максимум. Если вы спите приблизительно 8 часов в день, то в вашем распоряжении остается 16 часов. Часть этого времени вы потратите на себя и своих близких. Еще часть — на основную работу.

То, что останется, — это время для приобретения навыков. Если вы хотите совершенствовать свои навыки максимально быстро, то чем продолжительнее отрезки времени, которые вы можете выделить для занятий, тем лучше.

Наиболее эффективный подход к выделению времени для приобретения навыков — определить, когда время тратится с наименьшей пользой, а затем избавиться от подобного времяпрепровождения. В качестве эксперимента я рекомендую вам несколько дней вести короткий дневник, записывая в него, как вы проводите время.

Результаты эксперимента вас удивят: если вы примете пару трудных решений и сократите нерациональные действия, то у вас останется гораздо больше времени для приобретения новых навыков. И чем больше времени вы будете ежедневно посвящать занятиям, тем быстрее овладеете ими. Я рекомендую освободить время для полуторачасовых — по меньшей мере — ежедневных занятий, максимально сократив время, проводимое с наименьшей пользой.

Я также рекомендую *заранее настроиться* как минимум на 20 часов занятий. Приступив к занятиям, вы не должны останавливаться, пока не достигнете 20-часовой отметки. Если дело не движется, не ослабляйте усилий

— нельзя опускать руки, пока вы не достигли целевого уровня мастерства или не потратили 20 часов. Если вам жалко тратить на приобретение данного навыка 20 часов, выберите себе другое занятие.

Причина этого проста: первые этапы процесса приобретения навыка обычно кажутся труднее, чем они есть на самом деле. Вы часто теряетесь, сталкиваетесь с неожиданными проблемами и препятствиями. Вместо того чтобы отступать перед первыми же трудностями, заранее настройтесь заниматься 20 часов — и вам будет легче проявить упорство.

Рассматривайте такой подход как тренировку твердости характера: не позволяйте какой-нибудь глупой мелочи помешать выполнить желаемое. Вы либо разрешите проблему, либо продолжите упорно трудиться, пока не достигнете 20-часовой отметки. И окажетесь в гораздо лучшем положении для принятия дальнейших решений.

8. Создайте быстрые петли обратной связи

«Быстрая обратная связь» означает получение точной информации о своих действиях с максимально возможной скоростью. Чем больше времени необходимо для получения точной обратной связи, тем медленнее приобретается навык.

Возьмем, к примеру, искусство сыровара. Сложный химический процесс, благодаря которому получается превосходный сыр, нередко длится несколько месяцев или даже лет, и ускорить этот процесс, не испортив результат, никак невозможно. Если для того, чтобы определить, хорош ваш сыр или плох, требуется шесть месяцев, то такая задержка в обратной связи затрудняет быстрое приобретение навыка.

Быстрая обратная связь естественным образом приводит к быстрому приобретению навыка. Если обратная связь приходит мгновенно или с очень небольшой задержкой, то вам гораздо легче связать эту информацию со своими действиями и внести необходимые корректировки.

Лучшая обратная связь почти мгновенная. Вот почему такие занятия, как программирование, могут затягивать: вы вносите изменение — и через несколько миллисекунд компьютер отвечает, помогло ли оно. Если обратная связь вас не удовлетворила («аварийное завершение программы»), внесите еще одно изменение и повторите попытку.

Потенциальных источников обратной связи великое множество. Как объяснял в своей статье в журнале *The New Yorker* [2] известный хирург и теннисист-любитель Атул Гаванде, опытные тренеры и наставники способны предоставить вам мгновенную обратную связь относительно ваших действий и порекомендовать необходимые изменения.

Тренер — не единственный источник обратной связи. Регистрирующие

устройства, например видеорекамеры, помогут вам увидеть себя и свои действия. Такие инструменты, как компьютерные программы, учебные пособия и другие средства, также способны мгновенно указать, где вы совершили ошибку или допустили промах.

Чем больше источников быстрой обратной связи вы будете использовать в процессе занятий, тем быстрее овладеете навыком.

9. Занимайтесь по расписанию, короткими интенсивными интервалами

Наш мозг создан для того, чтобы учиться: замечать закономерности, моделировать возможные варианты действий, вычислять, что должно произойти дальше. Однако он не предназначен для точной оценки времени — того, как долго будет продолжаться то или иное действие или сколько времени вы потратите на то или иное занятие.

На первых этапах овладения новым навыком очень легко дать завышенную оценку времени, потраченному на занятия. Когда дело у вас получается неважно (и вы это знаете), время ползет как черепаха, и вам кажется, что вы занимаетесь дольше, чем на самом деле.

Решение этой проблемы — придерживаться расписания. Купите таймер [3] и поставьте его на 20 минут. Правило тут только одно: запустив отсчет, вы должны немедленно приступить к занятиям и не останавливаться, пока не истечет установленное время. Никаких исключений.

Этот простой прием позволит справиться и с более продолжительными периодами непрерывных занятий, даже когда вы устали или недовольны собой.

Чем больше будет продолжительных непрерывных занятий, тем быстрее вы овладеете навыком. Выделив время для трех-пяти таких занятий в день, вы за очень короткий период добьетесь существенного прогресса.

10. Уделяйте внимание количеству и скорости

Когда вы только начинаете осваивать новый навык, то испытываете искушение сосредоточиться на достижении идеальной техники — это путь к разочарованию. Ваша техника, естественно, будет далека от идеальной.

Вместо того чтобы пытаться приблизиться к идеалу, сосредоточьтесь на том, чтобы заниматься как можно больше и с максимальной скоростью, поддерживая «достаточно хорошую» форму.

В книге «Гид по выживанию для современного художника» [8] («Art & Fear») (2001) Дэвид Бейлс и Тед Орланд приводят очень интересную историю о важности объема:

Преподаватель гончарного искусства на первом занятии объявил классу,

что намерен разделить его на две группы. Ученики, расположившиеся в левой части студии, сказал он, будут оцениваться исходя исключительно из количества проделанной работы, а ученики справа — из ее качества.

Процедура была простой. В последний день занятий преподаватель принес напольные весы и взвесил работы «количественной» группы: 50 фунтов горшков оценивались на «отлично», 40 фунтов — на «четверку» и так далее. Членам же «качественной» группы требовалось изготовить только по одному горшку — но безупречному, — чтобы получить «пятерку».

В конце аттестации обнаружился любопытный факт: работы наилучшего качества были делом рук учеников «количественной» группы. Похоже, что в то время как «количественная» группа усердно «штамповала» горшки — и училась на своих ошибках, — «качественная» группа предавалась размышлениям на тему безупречности. Итогом стали грандиозное теоретизирование и груда мертвой глины[9].

Навык — это результат целенаправленных, настойчивых занятий, и на первом этапе объем и скорость важнее качества. Чем чаще тренировки и выше их скорость, тем быстрее приобретается требуемый навык.

Это не значит, что во время занятий вы не должны обращать внимания на правильность действий. Некоторые навыки, особенно те, что предполагают движения или действия, требуют определенного качества формы. Если вы обучаетесь живописи и поставили перед собой цель писать реалистические портреты, то сто полотен в день в стиле Джексона Поллока вам не помогут. В данном случае важна техника.

В первую очередь нужно убедиться, что применяемая вами в процессе занятий техника достаточна для удовлетворения целевого уровня мастерства. Если ваша техника удовлетворительная (как минимум на 80–90%), увеличьте скорость для более быстрого овладения навыком.

Таковы 10 простых принципов, которые позволят вам вырабатывать желаемый навык самым эффективным и результативным способом из всех возможных.

Неужели это работает?

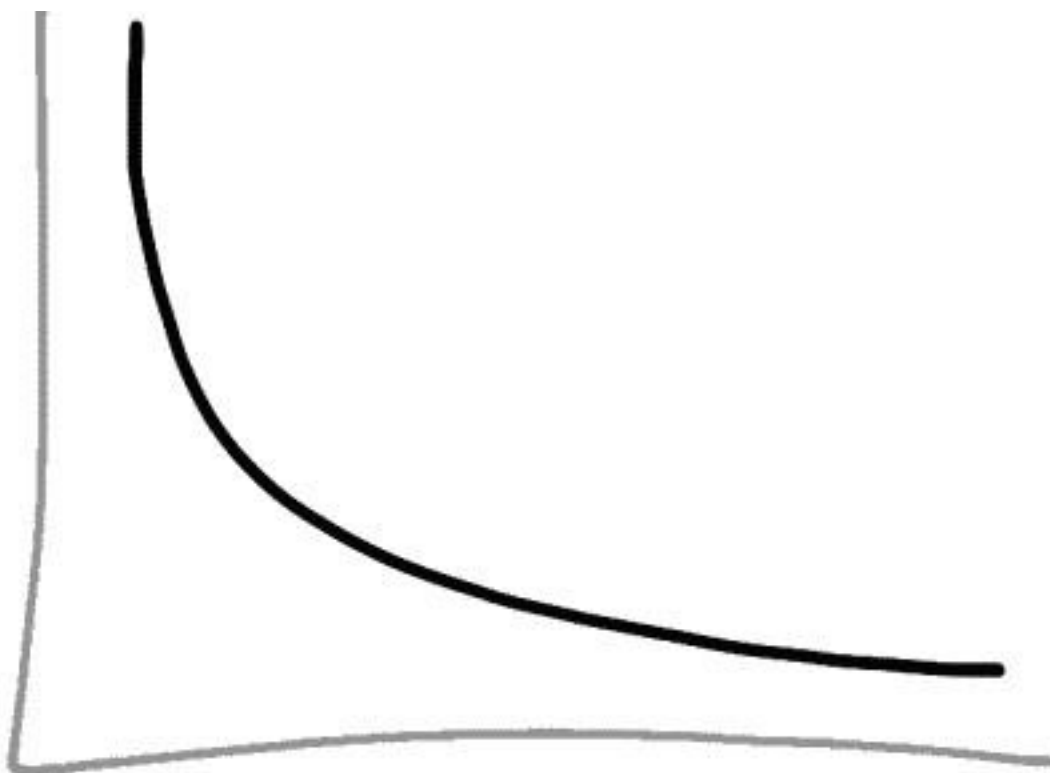
Поможет ли такой метод быстрее приобретать новые навыки? Исследования не оставляют в этом сомнений.

В научных изысканиях, посвященных приобретению когнитивных и моторных навыков, ученые обнаружили одну и ту же закономерность: когда испытуемые начинают тренировать новый навык, их производительность за очень короткий промежуток времени всегда существенно повышается. Не требуется много занятий, чтобы из «очень медленного и чрезвычайно

неумелого» превратиться в «относительно быстрого и довольно ловкого».

В специальной литературе эта закономерность известна как «степенной закон практики». Данный эффект был известен еще в 1924 году [4] и подтвержден не один раз в исследованиях как двигательных, так и интеллектуальных навыков [5]. В одной из работ даже утверждается, что «любую теорию приобретения навыка, которая не включает степенную функцию обучения, можно сразу же отвергнуть» [6].

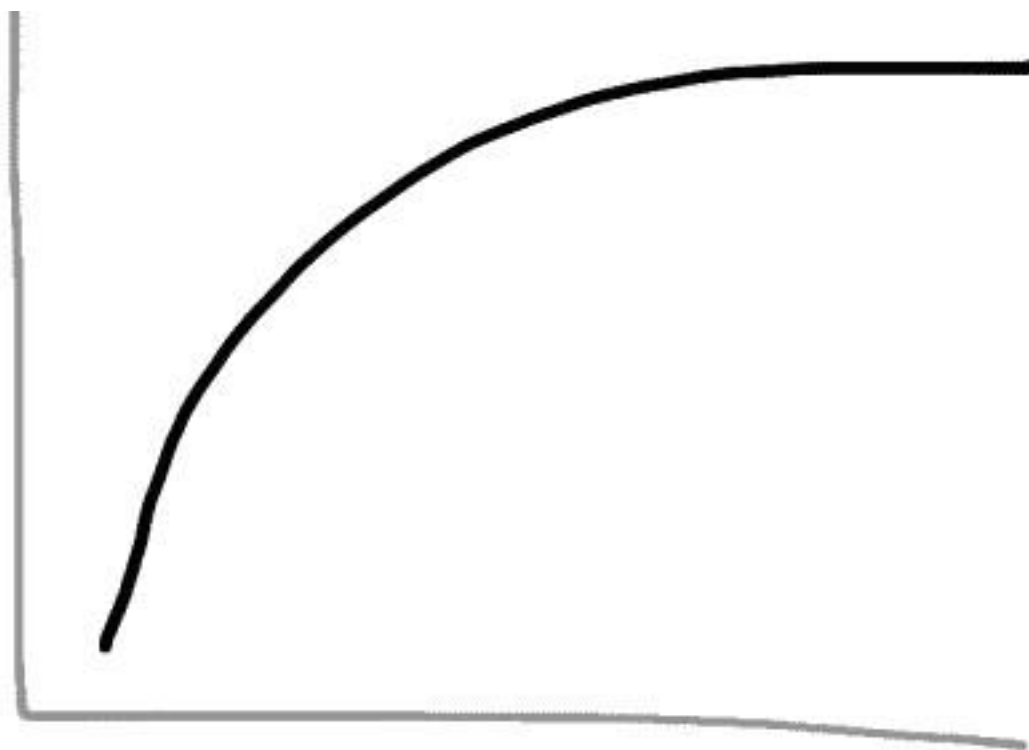
В научных исследованиях кривая «степенного закона практики» изображается следующим образом (время выполнения работы — по оси y , время занятий — по оси x):



Время занятий

С увеличением времени занятий кривая времени выполнения работы уходит вниз. Чем больше мы тренируемся, тем меньше времени требуется для достижения цели.

Интересно отметить, что если на оси y откладывать «успехи» (то есть определить производительность в более общих терминах, а не в единицах времени), то мы получим знаменитую *кривую обучения*:



Время занятий

Общая закономерность, которую отражает кривая обучения, выглядит следующим образом. В самом начале у вас ничего не получается, но первые успехи приходят довольно быстро, после того как вы овладеете основными элементами навыка. По достижении определенного уровня, через относительно небольшой промежуток времени, скорость овладения навыком падает, и следующие успехи приходят гораздо медленнее.

Вопреки распространенному мнению, крутая кривая обучения — это хорошо, а не плохо. График отчетливо иллюстрирует почему: крутая кривая обучения указывает на очень быстрое приобретение навыка. Чем круче кривая, тем больше успехов в единицу времени.

Представленный выше список принципов быстрого обучения можно рассматривать как способ сознательно сделать свою кривую обучения как можно круче. Сами принципы — это приемы, которые облегчают преодоление первых двух теоретических стадий процесса обучения (*когнитивной* и *ассоциативной*).

Когда вы приступаете к занятиям, ваши навыки естественным образом заметно улучшаются за очень короткий промежуток времени. Фокус в том, чтобы начать занятия как можно быстрее. Не *думать* о занятиях, не *волноваться* из-за них, а просто *заниматься*.

Очень часто у людей возникает ощущение, что они тратят много времени на какое-либо занятие, но при этом совсем мало упражняются. Если вы давно хотели чему-то научиться, мечтали достичь мастерства, но никак не могли

приступить к занятиям, это значит, что вы много лет расходовали мыслительную и эмоциональную энергию, но не продвинулись ни на шаг. Если вы не знаете своей цели или не имеете точной стратегии для ее достижения, то можете потратить массу сил на бесплодные блуждания.

Мои 10 принципов разработаны для того, чтобы помочь вам избавиться от этой непродуктивной пробуксовки и заменить ее действиями, которые играют главную роль в процессе приобретения навыка. Чем больше времени и сил вы вложите в преодоление первых двух стадий процесса обучения и меньше времени потратите на то, что вам ничем не поможет, тем быстрее вы овладеете навыком. Все очень просто.

А как насчет погружения?

Разумеется, предложенный метод приобретения новых навыков не единственный, но, вне всякого сомнения, наиболее гибкий. Другие методы могут дать аналогичные результаты, однако потребуют гораздо больших компромиссов.

Самый известный из общих методов быстрого приобретения навыка — это *погружение*. Он требует перемены обстановки таким образом, чтобы она способствовала непрерывной вынужденной практике. Если вы, например, желаете говорить по-французски, то изучение языка методом погружения потребует прожить во Франции несколько недель или месяцев.

В общем случае погружение эффективно. Переехав во Францию, вы будете вынуждены ежедневно упражняться во французском языке — каждую минуту и секунду, все время, пока вы там находитесь. По прошествии нескольких ужасных дней привыкания к новой обстановке вы заметите, что быстро делаете успехи.

Эффективность метода погружения обусловлена тем, что вам приходится преодолевать первые, самые главные часы обучения: от окружения никуда не убежать, и вы упражняетесь автоматически, помимо своей воли.

Недостаток погружения состоит в том, что навык, которым вы намерены овладеть, должен стать для вас главным на протяжении продолжительного времени. Если вы можете бросить все дела, упаковать чемоданы и отправиться во Францию, то изучение французского языка методом погружения — это хорошая стратегия.

Однако у большинства из нас есть обязанности, которыми мы не можем (или не хотим) пренебречь: семья, работа, выплаты по ипотеке и тому подобное. В подобных случаях использование метода погружения затруднительно или неосуществимо.

При худшем из возможных сценариев идея погружения становится

препятствием: если вы будете ждать возможности погружения, чтобы овладеть новым навыком, то можете потерять годы драгоценного времени.

Используйте возможности погружения, когда они появляются, но не рассчитывайте на них. Методы, предлагаемые в этой книге, помогут вам приобретать новые навыки даже в тех случаях, когда у вас есть всего один или два свободных часа в день.

Восстановление старых навыков

Важно также отметить, что эти принципы полезны и в том случае, когда навык, которым вы стремитесь овладеть, не является для вас совсем новым. Предлагаемые приемы применимы и для восстановления старых навыков, причем за рекордно короткое время.

Так, например, в старших классах школы я учился играть на трубе — довольно много занимался, и у меня неплохо получалось. Закончив школу и поступив в колледж, я больше не брал в руки инструмент.

Если у меня появится желание снова играть на трубе, то восстановление старого навыка не потребует продолжительных занятий. Я уже знаю все необходимые элементы и поэтому сосредоточусь на амбушюре (способе сложения губ, когда они обхватывают мундштук), чтении нот, восстановлению в памяти соответствующих положений пальцев, а также на теории музыки (тактах, темпе, звучании и экспрессии).

На восстановление навыка мне понадобится несколько часов занятий. Больше времени уйдет на то, чтобы выделить эти часы, устранить препятствия и упражняться по расписанию.

Хорошее начало — половина дела

Иногда хочется бросить гитару. Ты ненавидишь гитару. Но если ты справишься с собой, то будешь вознагражден.

Джими Хендрикс, гитарист-виртуоз

При овладении каждым новым навыком вам не обязательно использовать все вышеперечисленные принципы, но некоторые из них непременно окажутся полезными.

На мой взгляд, эти принципы следует воспринимать как контрольный список. Решив научиться чему-то новому, просто просмотрите список и решите, какие принципы применимы к вашему конкретному проекту.

Вот он, ваш контрольный список для быстрого приобретения навыка.

1. Выберите привлекательный проект.
2. Сосредоточьтесь на каком-то одном навыке.
3. Определите целевой уровень мастерства.
4. Разбейте навык на элементы.
5. Приготовьте все необходимое для занятий.
6. Устраните препятствия для занятий.
7. Выделите специальное время для занятий.
8. Создайте быстрые петли обратной связи.
9. Занимайтесь по расписанию, короткими интенсивными интервалами.
10. Уделяйте внимание количеству и скорости.

Вот и все. Примените этот контрольный список к тому навыку, которым вы хотите овладеть, и ваши занятия станут более рациональными и эффективными.

Как я уже говорил, мой метод — не ракетные технологии. Это стратегия и подготовка применительно к навыку, который вы хотите усовершенствовать, основанные на здравом смысле.

Теперь давайте посмотрим, как обучение и исследования могут сделать процесс приобретения навыка еще более эффективным.

3

10 принципов эффективного обучения

Ни одна проблема не устоит под натиском мысли.

Вольтер

В главе 1 мы уже обсуждали разницу между обучением и приобретением навыка. Однако это нисколько не умаляет важности обучения. Потратив немного времени на изыскания, прежде чем приступить к практике, вы сэкономите драгоценное время, энергию и душевные силы.

Теория сделает практические занятия более эффективными, позволит больше времени посвящать важным элементам, которые необходимо освоить первыми.

Из этого следуют 10 принципов эффективного обучения.

1. Изучите навык, о котором идет речь, и связанные с ним области.
2. Признайте, что вы ничего не понимаете.
3. Определите ментальные модели и мысленные зацепки.
4. Представьте результат, противоположный желаемому.
5. Поговорите с людьми, которые этим занимаются, чтобы знать, что вас ожидает.
6. Устраните из своего окружения все, что вас отвлекает.
7. Для запоминания используйте интервальные повторения и закрепление.
8. Создайте поддерживающие структуры и контрольные списки.
9. Сформулируйте и уточните прогнозы.
10. Уважайте свой организм.

1. Изучите навык, о котором идет речь, и связанные с ним области

Потратьте 20 минут на поиски в интернете, сходите в книжный магазин или в местную библиотеку за книгами и другими источниками информации об интересующем вас навыке. Ваша цель — найти не меньше трех книг, DVD с инструкциями, пособий или других источников информации о навыке, которым вы пытаетесь овладеть.

Только не впадайте в панику — вам не нужно тратить кучу времени на запоминание этих сведений. Наоборот: время, потраченное на чтение или просмотр, сокращает продолжительность практических занятий.

Вы не готовитесь к экзамену. Цель этих предварительных изысканий — определить главные элементы навыка, критически важные составляющие, а

также инструменты, необходимые для максимально быстрого успеха. Чем больше вы заранее узнаете о навыке, тем лучше подготовитесь. Ваша цель — как можно быстрее собрать широкий спектр информации о новом навыке, сформировав точное описание процесса его приобретения.

Для быстрого приобретения навыка беглый просмотр полезнее углубленного чтения. Отмечая в уме идеи и средства, повторяющиеся в разных источниках, вы убеждаетесь, что можете доверять выявленным закономерностям, и получаете возможность должным образом подготовить практические занятия.

Если, например, вы хотите испечь вкусный круассан, возьмите несколько кулинарных книг о выпечке. Вам не придется заново изобретать весь процесс — вы обнаружите, что существующая технология совершенствовалась лучшими кондитерами на протяжении многих лет. Если вы увидите, что один и тот же прием или процесс встречается в нескольких источниках, велика вероятность того, что эта информация действительно заслуживает внимания.

Обнаружив самые полезные — по всей вероятности — приемы, вы можете экспериментировать с ними на своей кухне, что избавит вас от многочисленных ошибок.

2. Признайте, что вы ничего не понимаете

В процессе предварительных исследований вы можете натолкнуться на теории, приемы и идеи, которые вам непонятны. Иногда какой-либо аспект навыка кажется очень важным, но вы понятия не имеете, что он означает. Вы читаете незнакомые слова или понимаете: другие делают то, что вы не в состоянии осознать.

Не паникуйте. Растерянность при первой встрече с незнакомым — это нормально. На самом деле, это даже хорошо. Не бегите от нее.

Предварительные исследования — один из лучших способов выявить главные элементы навыка и основные идеи. Однако велика вероятность, что вы не поймете их смысла сразу. Понимание придет позже, когда вы начнете практические занятия.

Доктор Стивен Крашен, специалист по обучению иностранным языкам, о котором я упоминал выше, называет это *постижимым потоком*. Новая информация, которую вы получаете, по определению, не очень понятна, поскольку не связана с предыдущими знаниями. Со временем, после приобретения некоторого опыта, та же самая информация становится постижимой. Как говорил известный учитель йоги Т. К. В. Дешикачар: «Осознание неведения само по себе является формой ясности».

Признание того, что вы пребываете в растерянности, очень важно. Оно помогает выяснить, *что именно* вы не понимаете, а значит, определить

направление исследований, которые помогут вам избавиться от этой растерянности.

Если вы понимаете больше половины того, с чем сталкиваетесь в предварительных исследованиях, значит, вы обучаетесь достаточно быстро. Если у вас возникают сомнения и колебания относительно темпа исследований, значит, вы на правильном пути. Если проблема или проект действительно вам интересны, то чем больше ваша растерянность на первом этапе процесса, тем сильнее внутренняя мотивация во всем разобраться и тем быстрее вы обучаетесь.

Нежелание признавать свое неведение — серьезный психологический барьер на пути к быстрому приобретению навыка. Чувствовать себя дураком неприятно, но, если вы напомним себе, что в процессе занятий такое ощущение пройдет, эта мысль поможет вам как можно быстрее перейти от неведения к ясности.

3. Определите ментальные модели и мысленные зацепки

В процессе исследования вы естественным образом начинаете замечать закономерности: идеи и приемы, которые встречаются вам снова и снова.

Эти концепции называются *ментальными моделями*, и они очень важны. Ментальные модели — это основа обучения, способ понимания и обозначения объекта или связи, существующих в нашем мире. По мере накопления точных ментальных моделей вам становится легче предвидеть, что произойдет после выполнения того или иного действия. Ментальные модели также значительно облегчают обсуждение своего опыта с другими людьми.

Вот вам пример: недавно я помогал отцу создать сайт. Приступив к делу, я попытался объяснить ему свои действия. Отец в итоге был недоволен нами обоими: я все время использовал такие слова, как «сервер», а он понятия не имел, о чем идет речь.

Но как только папа понял, что сервером называют специальный компьютер, предоставляющий пользователю доступ к веб-странице, и что сервер — это другой компьютер, а не тот, на котором мы работаем, ему стало гораздо легче понять, что же мы на самом деле делаем. В данном случае сервер представляет собой ментальную модель — знание этого термина облегчило отцу понимание процесса нашей работы над сайтом.

Кроме того, вы должны заметить, что некоторые вещи кажутся вам знакомыми. Это *мысленные зацепки* — аналогии и метафоры, которые можно использовать для запоминания новых концепций.

В случае с сервером представьте себе библиотеку. Когда вы идете в библиотеку и спрашиваете какую-либо книгу, библиотекарь просматривает стеллажи с сотнями или даже тысячами книг, чтобы отыскать именно ту,

которая вам нужна. Обнаружив книгу, он несет ее вам. Если нужной книги нет, библиотекарь говорит: «Я не могу найти книгу, которую вы ищете».

Именно так работают интернет-серверы. Получив запрос на конкретную веб-страницу, сервер ищет ее в своей памяти. Если страница найдена, она открывается для пользователя. Если же сервер не может ее найти, то посылает вам сообщение: «Ошибка 404: Страница не найдена». Представление сервера как «компьютерного библиотекаря» помогает понять работу системы.

Чем больше ментальных моделей и *мысленных зацепок* вы сумеете идентифицировать в процессе предварительных исследований, тем легче будет использовать их во время практических занятий.

4. Представьте результат, противоположный желаемому

Попытка представить, что приобретение нового навыка закончится катастрофой, а не триумфом, на первый взгляд выглядит нелогично.

Что если вы все сделали неправильно? Что если вы получили худший из возможных результатов?

Этот метод решения проблем называют *инверсией*, и он полезен для освоения основ практически любого дела. Изучив противоположность того, что вы хотите достичь, вы получаете возможность выявить важные элементы, которые в первый момент не кажутся очевидными.

Возьмем, к примеру, сплав по реке на байдарках. Что я должен знать, чтобы суметь сплавиться на байдарке по большой, быстрой, изобилующей порогами реке?

Обратимся к инверсии. Что будет, если все пойдет не так, как надо?

- Я перевернусь, окажусь под водой и не смогу выбраться.
- Вода попадет в байдарку, и она утонет, в результате чего я лишусь средства передвижения.
- Я потеряю весло и лишусь маневренности.
- Я ударюсь головой о камень.
- Я выпаду из байдарки, окажусь в водовороте и не смогу выплыть.

Если я умудрюсь сделать все это одновременно посреди бурной реки, то, скорее всего, распрощаюсь с жизнью — худший из возможных сценариев.

Такое мрачное направление мыслей полезно, поскольку указывает на несколько элементов сплава на байдарках по реке, которые могут оказаться жизненно важными:

- Умение возвращать перевернувшуюся байдарку на киль, не выпадая из нее.

- Умение не дать байдарке утонуть, если необходимо ее покинуть.
- Умение избежать потери весла в бурной воде.
- Умение безопасно огибать большие камни.
- Изучение русла реки перед сплавом, чтобы избежать опасных участков.

Ментальная модель определяет и список покупок: нужно потратиться на спасательный жилет, шлем и другое снаряжение, которое обеспечит безопасность.

Теперь вместо общего представления: 1) сплавиться по реке; 2) получить удовольствие и при этом; 3) остаться в живых — у меня имеется конкретный список элементов, которыми нужно овладеть, а также необходимых действий, чтобы получить удовольствие, не вывести из строя снаряжение и выйти живым из путешествия по реке.

Инверсия — эффективный метод.

5. Поговорите с людьми, которые этим занимаются, чтобы знать, что вас ожидает

Первый этап обучения позволит сформировать адекватные ожидания: как, скорее всего, будут выглядеть действия новичка?

Приступая к усвоению нового навыка, вы можете совершить весьма распространенную ошибку — недооценить сложность задачи или количество элементов, которыми необходимо овладеть для достижения успеха. А если навык к тому же предполагает возможность приобрести вес в обществе, то его притягательность вполне способна исказить ожидания.

Многие люди, стремившиеся стать рок-звездами, брали в руки гитару, но затем обнаруживали, что очень трудно одновременно хорошо играть, петь, не фальшивя, и потрясающе выглядеть. Отчасти проблема состоит в том, что «быть рок-звездой» — это не один навык, а целый набор взаимосвязанных мини-навыков и для овладения каждым из них нужно упорно трудиться.

Разговор с людьми, которые уже овладели интересующим вас навыком, поможет развеять мифы и неверные представления еще до того, как вы начнете вкладывать в занятия время и силы. Точно зная, что вас ждет в процессе тренировок, вы сможете поддерживать интерес к занятиям и избежите разочарования на первых этапах процесса.

6. Устраните из своего окружения все, что вас отвлекает

Отвлечения — это злейший враг быстрого приобретения навыка. Отвлечения убивают внимание во время занятий, а отсутствие сосредоточенности ведет к замедлению темпов приобретения навыка (или

вообще к отсутствию прогресса). Вы можете предвосхитить такой поворот событий, потратив несколько минут, чтобы еще до начала занятий устранить (или ослабить) как можно больше отвлекающих факторов.

Главные источники отвлечения можно разделить на две группы: электронные и биологические.

Телевизор, телефон и интернет — все это электронные отвлечения. Выключите их, заблокируйте, выдерните шнуры из розеток или любым другим способом удалите их из своего окружения, если они не являются необходимыми для самих занятий.

Исполненные благих намерений члены семьи, коллеги и домашние животные — это биологические отвлечения. Людей выключить не получится, но можно заранее сообщить им, что вы будете недоступны во время занятий, повысив таким образом вероятность того, что они проявят к вам уважение и не будут мешать.

Чем меньше отвлекающих факторов во время занятий, тем быстрее вы овладеете навыком.

7. Для запоминания используйте интервальные повторения и закрепление

Чтобы использовать материал, который вы усвоили во время занятий, необходимо быстро вспоминать связанные с ним идеи. Многие навыки требуют различной степени запоминания.

Тут есть одна ловушка: наша память несовершенна. Узнав что-либо новое, вы скорее всего это забудете, если не повторите через определенный промежуток времени. Такое повторение закрепляет освоенное и помогает мозгу сохранить его в долговременной памяти.

Исследователи обнаружили, что память подчиняется *кривой затухания*: новые концепции нуждаются в регулярном закреплении, но чем дольше вы знакомы с той или иной концепцией, тем реже требуется освежать ее в памяти, чтобы она не стерлась.

Интервальные повторения и закрепление — это техника запоминания, которая поможет вам систематически, на регулярной основе освежать в памяти важные концепции и информацию. Идеи, которые трудно запомнить, повторяются чаще, тогда как более легкие или более старые — реже.

Компьютерные программы для создания карточек, такие как Anki [1], SuperMemo [2] и SmartR [3], значительно упрощают интервальные повторения и закрепление. Системы интервальных повторений используют принцип считывания «карточки», и вам придется создавать эти карточки самому. Создавая карточки в процессе разделения навыка на элементы, вы одним выстрелом убиваете двух зайцев.

Для просмотра имеющихся карточек требуется лишь несколько минут в

день. Систематизируя процесс просмотра и упорядочивая поиск в памяти, эти системы помогают запомнить новые идеи, методы и процессы за рекордно короткое время. Постоянно просматривая карточки, вы очень быстро запомните необходимые концепции и идеи.

Очень важно отметить, что приобретение навыка обычно требует большей вовлеченности, чем обучение. Если нужно запомнить концепции, идеи или словарь, чтобы сдать экзамен, то вам достаточно всего лишь интервальных повторений [4].

Однако наилучший эффект эта техника дает в тех случаях, когда требуется быстро вспомнить информацию. Если вы набираете словарный запас, чтобы выучить новый язык, то интервальные повторения и закрепление вам очень помогут. В тех случаях, когда быстрое запоминание не критично, лучше отказаться от метода «карточек» в пользу практических занятий и экспериментов.

8. Создайте поддерживающие структуры и контрольные списки

Многие навыки требуют определенного порядка действий, который включает в себя планирование, подготовку, поддержку, завершение и так далее. Создание простой системы — лучший способ обеспечить эти важные элементы с минимумом дополнительных усилий.

Контрольные списки удобны для запоминания действий, которые необходимо выполнять во время каждого практического занятия. Это способ систематизации процесса и позволяет сосредоточиться на важных вещах.

Поддерживающие структуры обеспечивают одинаковый подход к выбранному навыку во время каждого занятия. Представьте баскетболиста, который готовится выполнить штрафной бросок. Вытереть руки о майку, расслабить плечи, взять мяч у судьи, три раза ударить о площадку, застыть на три секунды, затем бросить в кольцо. Это и есть поддерживающая структура.

Создание поддерживающих структур повышает эффективность занятий. Они также облегчают визуализацию, и это помогает воспользоваться таким приемом, как мысленное повторение, который полезен при освоении отдельных видов двигательных навыков.

9. Сформулируйте и уточните прогнозы

Процесс приобретения навыка включает экспериментирование: вы пробуете новое и смотрите, что из этого получается.

Настоящей проверкой полезности обучения служит прогноз. Можете ли вы, основываясь на полученных знаниях, заранее предсказать, к чему приведут изменения в ваших действиях или новый эксперимент?

Привычка делать и проверять прогнозы поможет вам быстрее овладеть навыком. Это разновидность научного метода, включающего четыре ключевых элемента.

- **Наблюдения** — что вы в данный момент видите?
- **Знания** — что вы уже знаете об интересующем вас предмете?
- **Гипотезы** — что, по вашему мнению, усовершенствует ваш навык?
- **Тесты** — что вы собираетесь еще попробовать?

Я рекомендую использовать блокнот или любое другое средство записи, чтобы во время занятий отслеживать результаты экспериментов и формировать гипотезы. Следя за своими прогнозами и генерируя новые идеи, вы сделаете эксперименты более плодотворными.

10. Уважайте свой организм

Мозг и тело — это биологические системы, и потребности у них тоже биологические: пища, вода, физические упражнения, отдых и сон. Существует и опасность переутомления, которое снижает эффективность занятий. Не получая необходимого, разум и тело не дадут нужного результата.

По мнению Тони Шварца, автора книг «Жизнь на полной мощности»^[10] («The Power of Full Engagement») (2004) и «Быть идеальным во всем» («Be Excellent at Anything») (2011), оптимальный цикл обучения требует около 90 минут сосредоточенных усилий. После этого вашему мозгу и телу потребуется перерыв. Используйте эту возможность для того, чтобы заняться спортом, перекусить, вздремнуть или переключиться на что-то другое.

Этот принцип прекрасно согласуется с занятиями по часам. Если перед началом практических или теоретических занятий вы поставите таймер на промежуток от часа до полутора, то по окончании отведенного времени не забудете сделать перерыв.

При необходимости можно также разбить занятия на несколько частей с короткими перерывами между ними: 20 минут занятий — 10-минутный перерыв, 20 минут занятий — 10-минутный перерыв и так далее.

Подводя итоги

*У нас нет правил. Мы пытаемся достичь результата.
Томас Эдисон, изобретатель*

Для каждого приобретаемого навыка вам не понадобятся все изложенные

выше принципы, но некоторые из них наверняка окажутся полезными.

На мой взгляд, следует относиться к этим принципам как к дополнительному контрольному списку. Решив приобрести новый навык, вы просто просматриваете список и отмечаете, какие принципы применимы к вашему проекту.

Итак, контрольный список для эффективного обучения.

1. Изучите навык, о котором идет речь, и связанные с ним области.
2. Признайте, что вы ничего не понимаете.
3. Определите ментальные модели и мысленные зацепки.
4. Представьте результат, противоположный желаемому.
5. Поговорите с людьми, которые этим занимаются, чтобы знать, что вас ожидает.
6. Устраните из своего окружения все, что вас отвлекает.
7. Для запоминания используйте интервальные повторения и закрепление.
8. Создайте поддерживающие структуры и контрольные списки.
9. Сформулируйте и уточните прогнозы.
10. Уважайте свой организм.

Применив этот контрольный список к тому навыку, который хотите приобрести, вы узнаете все необходимое, чтобы занятия стали рациональными и эффективными.

Применение теории на практике

*Бесполезно садиться и писать, когда вы еще не встали, чтобы жить.
Генри Дэвид Торо*

Хватит теории — пора приступать к практике.

Мы уже рассмотрели основы быстрого приобретения навыка, но главное — не знать, а делать. Помните: без практических занятий невозможно овладеть никаким умением. Вместо того чтобы все больше углубляться в теорию приобретения навыка, я продемонстрирую, как это делается. Я применю изложенные принципы к овладению несколькими новыми навыками, а вы будете наблюдать за мной, как говорится, из первого ряда.

Вот навыки, которыми я намерен овладеть.

- Йога: комплекс асан для домашних занятий.
- Программирование: создание работающего веб-приложения.

- Машинопись: переучивание на печать слепым методом.
- Стратегия: игра го, самая древняя и самая сложная из настольных игр.
- Музыка: игра на гавайской гитаре.
- Виндсерфинг: передвижение и маневрирование на поверхности воды.

У меня нет опыта ни в одном из этих занятий. При помощи описанных выше приемов и методов я хочу овладеть каждым из навыков не больше чем за 30 дней. Предполагаемое время занятий для каждого навыка — приблизительно 20 часов, то есть от 60 до 90 минут занятий ежедневно.

О примерах

Приведенные ниже конкретные примеры сугубо индивидуальны. Это умения, которыми мне захотелось овладеть по разным причинам, и об этих причинах я потом подробно расскажу. Навыки, интересующие *вас*, могут быть совсем другими, однако процесс их приобретения в своей основе останется тем же.

Естественно, какие-то из приведенных примеров могут вас заинтересовать больше остальных. В таком случае прочтите эти главы первыми. Если вы начали читать главу и описываемый в ней навык неинтересен или неприменим к вашей ситуации, можете спокойно ее пропустить. Метод, который я использую, одинаков во всех случаях, и вы не упустите ничего важного.

Следующие главы я пишу в виде пособия, предполагая, что эти навыки для вас новые, вы не имеете ни опыта в их использовании, ни знаний о них. Надеюсь, что, помимо иллюстрации метода, эти главы познакомят вас с шестью интересными занятиями, которые могут доставить удовольствие.

Если у вас богатый опыт в одной из этих областей, то вы, наверное, найдете ошибки в моем понимании предмета, терминологии или вообще не согласитесь с моим методом. Это абсолютно нормально. *Помните: я начинаю как полный профан и не знаю, что делаю.* (Пока что.)

То, что вы прочтете, — это описание процесса моего обучения каждому из навыков. Я очень старался, чтобы информация, представленная в книге, была точной и полной, однако ошибки неизбежны. Во всех случаях самое главное — это *метод* исследования и намеренной практики.

Итак, начнем: йога.

4

Йога

Урок: не усложняйте себе жизнь

Я измеряю свои успехи в йоге не глубиной наклона или поворота, а тем, как обращаюсь со своей женой и детьми.

Т. К. В. Дешикачар, знаменитый учитель йоги

Я старею.

Поясню: вообще-то я еще относительно молодой человек, однако в последнее время наблюдаю у себя признаки, которые вызывают беспокойство.

Просыпаясь утром, я часто чувствую тупую боль где-то посередине спины. После дня, проведенного за компьютером, шея и плечи у меня напряжены и немеют. Не очень приятное ощущение.

До этого момента я не уделял особого внимания своему телу. Оно просто действовало. Расставшись со школой, а вместе с ней и с регулярными уроками физкультуры, я вообще не занимался спортом. В колледже я относился к своему телу как к средству передвижения, которое предназначено для того, чтобы перемещать мой мозг из одной аудитории в другую.

По окончании университета это отношение не изменилось, и до сих пор я не чувствовал никаких неудобств. Теперь все эти годы пренебрежения физическими упражнениями дали себя знать.

У меня хрустит все тело, и я это чувствую.

«Ты должен внимательно присмотреться к йоге...»

Моя жена Келси начала заниматься йогой еще в колледже. Почти каждый день она ходит в студию йоги, расположенную на нашей улице, и проводит час или больше в очень теплой комнате, принимая разные позы. Ей это нравится, и она всегда возвращается домой довольная и расслабленная.

Каждую неделю Келси говорит мне: «Ты должен обязательно попробовать йогу. Это здорово. Тебе понравится».

Я колебался. Йога меня не очень привлекала.

И дело не в растяжке. В старших классах школы я занимался бегом, так что растяжка для меня не проблема. Моей любимой дисциплиной был бег на 110 метров с барьерами, который требует хорошей гибкости [1].

Тренировка барьериста предполагает серьезную растяжку, и за прошедшие годы мне удалось сохранить эластичность связок и мышц. Даже

теперь, 10 лет спустя, я легко могу достать ладонями пол, не сгибая коленей.

Мои колебания относительно йоги, откровенно говоря, были связаны с тем, что она казалась мне *странной*.

Чакры, ауры и кундалини... О боже!

Видеоролик «Что говорят йоги» (Sh!t Yogis Say) [2], снятый компанией Lululemon [3], самым прибыльным из розничных торговцев спортивными аксессуарами, представляет собой превосходную пародию на то, как йога может выглядеть в глазах непосвященных. Вот несколько самых ярких фраз:

- Меня беспокоит ваша аура...
- Как сказать это на санскрите?
- Мои чакры *так* сбалансированы.
- Хотите посмотреть, куда я могу закинуть ногу?

Со времени появления ролика на YouTube в декабре 2011 года его просмотрели более 2 миллионов раз — он явно задел зрителей за живое.

Я очень рациональный и приземленный человек. И готов поверить, что регулярная растяжка приносит пользу. Выучить несколько упражнений на растяжку — это мне кажется разумным.

Однако все эти разговоры про ауры, чакры и эзотерические духовные практики только отвлекают от дела. Я не понимаю, какое отношение все это имеет к физическим упражнениям, и, откровенно говоря, считаю, что мистические теории в стиле хиппи делают йогу похожей на какой-то странный культ.

Но это еще не все. Занятие йогой в Америке превратилось в образ жизни, индустрию фитнеса и спортивной экипировки с оборотом в 8 миллиардов долларов. Демонстрация своей принадлежности к сообществу любителей йоги стала модным сигналом о социальном статусе, своего рода эквивалентом последней модели сумочки Louis Vuitton.

Нет, спасибо. Мои «чакры» в полном порядке, премного благодарен.

«Расслабьте лицо»

В конечном итоге Келси убедила меня попробовать заняться йогой — в качестве новогоднего обещания «начать новую жизнь». Я работал дома, и она сказала, что это прекрасная возможность выйти из квартиры и пообщаться с людьми. Я был настроен скептически, но решил доставить ей удовольствие.

Мы записались в Pure Yoga, шикарную новую студию йоги в Верхнем

Ист-Сайде Нью-Йорка. Занятия там вел Марко Рохас, любимый преподаватель Келси и один из лучших инструкторов в городе.

Представьте более молодую и худую версию Антонио Бандераса, в широких штанах для занятий йогой — это будет Марко. Красивый, харизматичный, с неотразимым голосом и испанским акцентом. На занятиях Марко всегда полно народу, во-первых, потому что он действительно потрясающий преподаватель, а во-вторых, потому что 99 процентов любителей йоги города без ума от него. (И кто может их за это винить?)

Поскольку я ничего не знал о йоге, то пытался следовать указаниям Марко. Я не знал ни поз, ни их названий, а просто смотрел, что делают другие, и пытался им подражать. Это требовало полной концентрации. Я справлялся, но с трудом. Некоторые позы, например стойка на голове, оказались за гранью моих возможностей.

Я старался держаться в глубине зала, чтобы видеть, что делают остальные, а также минимизировать ущерб для окружающих в случае моего падения, что случалось чаще, чем я готов это признать.

Мне нравились занятия Марко, но истинного удовольствия от происходящего я не получал. Марко постоянно останавливался у моего коврика и корректировал мои движения, помогая принять правильную позу.

Один случай мне запомнился особенно. Марко взял меня за руки и потянул. Сильно.

— Используйте четырехглавые мышцы... поверните бедра. Хорошо. А теперь... расслабьте лицо.

Мое лицо было искажено гримасой, и на то имелась веская причина. И я расслабил мышцы лица.

Дилемма домоседа

Мне нравились занятия у Марко, но выделить для них время оказалось не так-то просто. Продолжительность каждого составляла полтора часа, плюс время, чтобы дойти до студии, переодеться, принять душ после занятий и вернуться домой.

Несмотря на то что студия йоги находилась всего в нескольких кварталах от меня, все эти действия в совокупности занимали не менее двух часов. В то время я работал дома, мои обязанности в компании Fortune 500 требовали большой ответственности; кроме того, мне часто приходилось встречаться с людьми, и эти встречи периодически совпадали по времени с занятиями в студии. Чтобы попасть на утренние или дневные уроки, мне приходилось откладывать работу. Но, вообще говоря, проблема была по большей части психологической: я с трудом мог убедить себя, что можно в течение двух часов

завязывать свое тело в узел, когда меня ждет работа.

В конечном итоге за год я посетил около пятнадцати занятий. После каждого занятия я прекрасно себя чувствовал, но считал, что все это не стоит затраченного времени.

Кроме того, я не занимался йогой дома. Просто не знал как. В студии я копировал действия других учеников. Конечно, дома можно расстелить на полу коврик и заняться растяжкой, но, как мне кажется, это совсем не йога.

Когда ученик готов, появляется учитель

После переезда в Колорадо Келси как-то вечером призналась мне, что скучает по Нью-Йорку. Ей не хватает занятий в студии Марко, курсов по анатомии йоги Лесли Каминоффа, на которые она записалась после получения свидетельства преподавателя йоги. Лесли — соавтор книги «Анатомия йоги»^[11] («Yoga Anatomy») (2007, 2011), одной из самых успешных за последнее время книг о йоге, по которой готовят преподавателей йоги во всем мире. К сожалению, для того чтобы посещать курсы Лесли, нужно жить в Нью-Йорке, и после переезда Келси пришлось бросить их.

— Кто-то должен выложить курсы Лесли в интернет, — сказала Келси.

Парадоксальное заявление. Несколько недель назад Келси помогала мне запустить мои первые онлайн-курсы, так что она знала, как это делается. И Келси была права — уроки Лесли Каминоффа доступны только ньюйоркцам, а поскольку потребность в них есть во всем мире, есть смысл вывести эти курсы в онлайн.

Я указал жене на очевидный факт: она вполне способна сделать онлайн-курс обучения, если Лесли даст на это согласие. Вечером Келси отправила ему свое предложение.

Два дня спустя Келси и Лесли договорились о сотрудничестве, и Келси открыла свой первый бизнес, связанный с продвижением онлайн-курсов йоги.

Поскольку теперь мы оба работали дома, я стал *гораздо* больше слышать о йоге. Производственный процесс включает просмотр видеосъемки каждого занятия и составление подробных комментариев. Так я познакомился с самим курсом Лесли.

Момент, когда я решил серьезно заняться йогой

Прежде всего, меня привлекла тональность речи Лесли. Совсем не похожая на тональность беззаботного хиппи, которую я ожидал услышать.

Наоборот — обладая 30-летним практическим опытом в области анатомии, физиологии и спортивной медицины, он не терпел расплывчатых терминов.

С точки зрения Лесли, йога ценна в первую очередь своей практичностью. Занятия йогой — это превосходный способ развития силы, повышения гибкости и поддержания подвижности.

Эффективность йоги значительно усиливается, если сосредоточиться на технике дыхания. Большинство людей рассматривают занятия йогой как причудливую смесь аэробики и гимнастики, но такая картина неточна. Йогу делает *йогой* комбинация дыхания, движения и внутренней сосредоточенности.

Лесли также объясняет некоторые странности в терминологии. Первые йоги экспериментировали со своим телом и психикой, пытались систематизировать свои знания еще до того, как оформились такие современные научные дисциплины, как анатомия, биология и когнитивная психология.

В результате, когда первые йоги обнаруживали нечто интересное, единственным способом описания для них оставались метафоры. Так, например, когда практикующие йоги замечали, что некоторые позы и способы дыхания вызывают определенный эмоциональный отклик, то объясняли эти явления в доступных им терминах.

Именно таким образом возникло понятие *чакры*: с научной точки зрения, не существует тайных центров энергии в животе, крестце, солнечном сплетении, сердце, горле, во лбу и макушке. Однако эти метафоры помогали первым йогам описывать свои ощущения и оказались настолько удобными, что сохранились до наших дней.

Последней каплей стал видеоролик на YouTube, который Лесли представил во время одного из первых онлайн-уроков. В нем Гил Хедли [4], преподаватель анатомии, проводил учебную аутопсию трупа перед группой студентов [5]. Гил обратил внимание на любопытную особенность мышечной фасции, оболочки, которая покрывает наши мышцы.

Подобно связкам и сухожилиям, фасции относятся к соединительным тканям: они скрепляют части нашего тела. Охватывая группы мышц, фасции помогают нам двигаться, облегчая скольжение мышц относительно друг друга.

Но у фасции есть один недостаток: когда мышцы неподвижны, «ворсинки» соединительной ткани, по внешнему виду и структуре похожие на вату, начинают прорастать *сквозь* фасцию.

Обычно это не составляет большой проблемы. Отдельные волокна очень тонкие и легко рвутся при движении мышц. Неприятности возникают при недостатке движения.

Если мышцы сохраняют неподвижность в течение определенного промежутка времени, соединительных «ворсинок» становится все больше, они уплотняются, затвердевают и ограничивают диапазон движений.

Я *знал*, что слишком мало двигаюсь, и этот видеоролик привлек мое

внимание. Может быть «хруст», который я ощущал во всем теле, указывал как раз на то, что мои мышцы зарастают «ворсинками»? (Жуть какая!)

Это одно из возможных объяснений, но не единственное. По мнению Лесли, боль в спине и шее также может объясняться недостаточным поступлением кислорода к этим мышцам — такое явление получило название *ишемии*. Неподвижность может привести к ограничению кровоснабжения мышц. Когда мышца получает недостаточно кислорода, в ней возбуждаются болевые рецепторы, в результате чего возникает тупая, не проходящая боль. Чем дольше сохраняется такое состояние, тем сильнее боль.

Ишемии также способствуют хронический стресс и неправильное дыхание. Эти факторы часто сопутствуют друг другу — при стрессе люди склонны задерживать дыхание. Если вы поможете своим мышцам получать больше кислорода — движением, правильным дыханием, уменьшением уровня стресса, — то боль в них исчезнет [6].

В любом случае, нагрузка на мышцы и снабжение их кислородом играют важную роль в ослаблении боли и в поддержании диапазона движений. Йога сочетает движения, дыхание и медитацию, и поэтому она особенно эффективна в ослаблении и устранении хронической мышечной боли.

Этих доказательств достаточно. Я буду учиться йоге.

Но... с чего начать?

Что же такое «йога»?

Поскольку я собираюсь заняться йогой, то мне нужно знать основы того, что я делаю. Если йога представляет собой только растяжку, то ее так и называли бы «растяжкой».

Как выяснилось, ключом к определению сути йоги является понимание того, где возникла эта практика и почему мы сегодня называем ее «йогой» [7].

Йога в той или иной форме существует уже не одну тысячу лет. Первые известные свидетельства существования йоги дают нам артефакты, найденные в местах археологических раскопок, связанных с цивилизацией долины Инда, которая существовала на территории современного Пакистана и северо-запада Индии приблизительно с 3300 по 1300 гг. до н. э.

Йога как философия и практика представляла собой свод приемов и техник, которые не имели прямого отношения к религии, но были тесно связаны с религиозными и философскими традициями. В поисках Брахмы, «основы всего сущего», жрецы древней ведической культуры проводили сложные ритуалы, чтобы соединить материальный мир с духовным. Термин «йога» происходит от древнего санскритского слова *упряжка*. Подобно тому как бык привязан к плугу, вспахивающему поле, жрецы пытались привязать духовный

мир к миру материальному.

Со временем ведические традиции угасли. Жрецы стали опираться на понятия *атмана*, высшей природы человеческой души. Внешний, связанный с движением ритуал превратился в интроспективную медитацию, и приверженцы духовной практики в поисках абсолютной истины начали отвергать материальный мир и сложный ритуал, заменяя его скитаниями по лесам, обетами бедности и аскетизма.

Мудрец Яджнавалкья, живший во времена, которые мы называем эпохой Упанишад, предположил, что властное эго не позволяет нам познать свою истинную суть. Разрушив эго, мы можем стать единым целым с атманом, «душой», или истинным «я». Яджнавалкья также ввел понятие *кармы*, которую он определял как действия, предпринимаемые для освобождения от эго. Отделив тело от атмана, можно достичь состояния, которое Будда в более поздние времена называл просветлением.

Появление асан

Философские идеи Яджнавалкьи привели к развитию того, что большинство практикующих йогу считают ядром современной йоги: физические позы в сочетании с контролем дыхания и медитацией. Специалист по истории религии Карен Армстронг в своей книге «Великое превращение» («The Great Transformation») [8] объясняет:

Йога — одно из величайших индийских достижений и в своей самой развитой форме почти наверняка было задумано [практикующими йогами] как средство освобождения *пуруши* [истинного «я»] от пут природы.

Эта классическая йога существенно отличается от той версии йоги, которой сегодня часто обучают на Западе. Это были не аэробные упражнения, и занятия не помогали людям примириться с жизнью — как раз наоборот. Йога была систематическим наступлением на эго, строгим режимом, который за большой промежуток времени учил кандидата отказываться от нормального сознания с его ошибками и заблуждениями и заменять его экстатическим познанием своего пуруши.

Йога была настоящей работой. Такое занятие не для дилетантов или малодушных — это требовательная и суровая духовная дисциплина, исключительная сфера монахов и гуру. Классический йог занимается не для того, чтобы стать более сильным или гибким. Он стремится разорвать связь между своим телом и атманом. Нелегкий труд.

Кодификация йоги

Во II веке до н. э. ученый по имени Патанджали начал объединять философские учения и практику йоги той эпохи. Изложением лучших (*раджа*, или «царских») практик стали его «Йога-сутры», собрание изречений, основополагающий текст для тех, кто практикует йогу.

В «Йога-сутрах» Патанджали определил восемь главных аспектов, или ветвей, *раджа*-йоги:

- **яма** — нравственность;
- **нияма** — духовное самоочищение;
- **асана** — позы;
- **пранаяма** — управление дыханием;
- **пратьяхара** — управление эмоциями;
- **дхарна** — концентрация;
- **дхьяна** — медитация;
- **самадхи** — погружение/размышление.

Патанджали называл эту систему «аштанга», или «восьмиступенчатой» йогой. Усердные занятия всеми восемью ступенями позволяют практикующему йогу достичь состояния *кайвалья* — отделения души от материального мира, которое приводит к вечному блаженству.

То, что представляет большинство людей, услышав слово «йога», — складывание своего тела в необычные позы — это только одна из ступеней, *асана*. Как правило, асана сочетается с определенной техникой дыхания, которая называется *пранаяма*. Цель таких занятий — подготовиться к медитации (*дхьяна*).

Система Патанджали рассматривает йогу как философию, а не как систему физических упражнений. Степень асана по большей части была ограничена статическими позами в положении сидя — совсем не похожими на гимнастические трюки тех, кто практикует йогу сегодня.

Но это не значит, что сложных поз тогда не было. Позы *хатха-йоги* (усиленной йоги) были известны еще со времен Яджнавалкьи, но атлетизм тогда считался второстепенной задачей.

Самая известная систематизация практик хатха-йоги содержится в трактате «Хатха-йога Прадипика» жившего в XV веке мудреца Махариши Сватмарамы. Сватмарама считал хатха-йогу идеальным путем к *раджа*-йоге Патанджали. Сватмарама верил, что, очищая тело с помощью физических упражнений, практикуя пранаяму и вдыхая дым благовоний, стоя на голове, можно достичь высших уровней сознания.

Ранние формы хатха-йоги не сохранились. Интенсивные физические упражнения оставались малоизвестным ответвлением раджа-йоги, пока четыре века спустя брамин из индийского города Майсор не революционизировал это направление йоги.

Человек, который «изобрел» современную йогу

Тирумалай Кришнамачарья родился в 1888 году и был старшим сыном известного брамина Шри Тачарайи [9]. С хатха-йогой его в пятилетнем возрасте познакомил отец, и Кришнамачарья продолжал заниматься ею до завершения формального образования. В 1919 году он предпринял путешествие на гору Кайлас в Тибете, чтобы обучаться под руководством одного из последних мастеров хатха-йоги, Шри Рамамохана Брахмачари.

Кришнамачарья провел у Брахмачари семь с половиной лет, запоминая «Йога-сутры» Патанджали и практикуя асану. Рассказывают также, что он научился некоторым эффектным трюкам, таким как замедление дыхания и остановка пульса на длительное время. Впоследствии Кришнамачарья использовал эти умения для популяризации хатха-йоги, демонстрируя их на публике.

Завещание гуру

Отношения учителя (гуру) и ученика имеют в Индии долгую историю. При расставании с гуру ученики обычно делали ему пожертвование, так называемое *гуру-дакшина*, в знак благодарности за науку и глубокого уважения. Нередко пожертвование вносилось деньгами или материальными ценностями, но иногда гуру давал ученику особое задание. Когда обучение Кришнамачарьи закончилось, Брахмачари обратился к ученику с просьбой: он должен жениться, завести детей и распространять знания, приобретенные на горе Кайлас.

Просьба гуру потрясла молодого человека. Дело в том, что Кришнамачарья получил блестящее образование, уже имел несколько религиозных и научных степеней и ему прочили место, которое занимал его дед, — религиозного лидера Южной Индии. Чтобы получить представление о масштабах потери такого статуса, вообразите, что вас назначили в руководство престижной компании, а затем вдруг объявили, что вы должны стать консьержем. *Дакшина* Брахмачари означала суровый приговор — жизненные трудности и низкий социальный статус.

Кришнамачарья выполнил просьбу своего наставника; он вернулся в Майсор, жил в бедности и искал возможность распространять хатха-йогу.

В 1931 году Кришнамачарья был приглашен преподавать хатха-йогу в Санскритском колледже в Майсоре. Там его представили Кришне Радже Вадийяру IV, правящему махарадже Майсора. Впечатленный возможностями и знаниями йога, а также в благодарность за то, что Кришнамачарья помог ему излечиться от диабета, махараджа предложил открыть у себя во дворце школу йоги, которая будет находиться под покровительством правителя.

Поскольку большинство учеников *йога шала* в Майсоре составляли непоседливые мальчишки, Кришнамачарья разработал новую разновидность хатха-йоги, направленную на развитие силы и гибкости. Он объединил асаны традиционной хатха-йоги с движениями британской гимнастики, приспособив и изменив последовательность поз в соответствии с возможностями каждого ученика [10].

Новая йога

Таково происхождение системы, которая сегодня известна под названием аштанга-виньяса-йога. Если вы пришли на занятия в студию йоги или взяли любую книгу, в которой есть слова *аштанга*, *виньяса*, «сила», «поток энергии», Айенгар, то знайте — вы принадлежите к традиции Кришнамачарьи.

Кришнамачарья приспособивал свое учение к потребностям учеников. Аштанга-йога была популяризирована одним из первых последователей Кришнамачарьи, Паттабхи Джойсом, который учился в Майсоре. Аштанга сфокусирована в первую очередь на шести последовательностях поз, начиная с «первой последовательности», которой Кришнамачарья обучал в своей *йога шала*. Современная аштанга сохранила упор на последовательность поз и атлетизм.

Обучая своего шурина Б. К. С. Айенгара, Кришнамачарья основное внимание уделял положению тела в асанах — с учетом слабого здоровья Айенгара. Со временем Айенгар развил этот подход и разработал тот современный, основанный на асанах стиль йоги, который практикуют сегодня.

В преклонном возрасте Кришнамачарья большую часть времени посвящал врачеванию и обучил своего сына Т. К. В. Дешикачара терапевтическим аспектам практики асан. Сегодня ученики Дешикачара подчеркивают значение йоги как средства улучшения самочувствия и указывают на широкий спектр ее потенциального применения в медицине, от физиотерапии до снятия тревожного состояния. В результате практически все крупные школы современной йоги испытали сильное влияние учения Кришнамачарьи.

Неизменным осталось одно — обращение к обычным людям с повседневными обязанностями. Йога Кришнамачарьи предназначена не для монаха или аскета: упор на индивидуализированную практику обычных людей

сделал асану доступной миллионам занятых людей во всем мире. Как говорил Т. К. В. Дешикачар, «вы занимаетесь йогой, чтобы жить, а не наоборот».

Йога = дыхание + движение + медитация

Вернемся к настоящему. Какое отношение имеют все эти истории к современной йоге?

Вот что меня удивило: с исторической точки зрения, современная практика асан является *недавним* изобретением. Конечно, философским аспектам йоги уже несколько тысяч лет, но реальная практика, когда вы принимаете различные позы, сочетая их с определенной техникой дыхания и медитацией, исторически очень молода.

Внимание, уделяемое в практике йоги силе и гибкости, — тоже новая тенденция. Лишь очень немногие из практикующих йогу пытаются силой отделить душу от тела в поисках своего атмана. Большинство не причисляет себя к аскетам и отшельникам. Современные поклонники йоги в основном хотят поддержать физическую форму, развить гибкость, избавиться от стресса.

В целом понятно, что современная практика йоги сочетает движения, дыхание и медитацию. В самих позах нет ничего сверхъестественного. Они менялись на протяжении столетий, и Кришнамачарья без колебаний добавлял новые элементы или видоизменял старые в соответствии с возможностями ученика.

Понятно также, что в йоге нет стремительных движений. Подход к упражнениям здесь философский: осознать, что ты тратишь время на совершенствование себя. Но, несмотря на то что современная асана относительно нова, философия йоги насчитывает несколько тысяч лет и интересна сама по себе. Йога обладает глубиной, которой лишены новейшие направления фитнеса.

Кроме того, нет никакой нужды тренировать самые невероятные позы. Стать «настоящим» йогом — это не значит гнуться, как пластилиновый человечек Гамби, а делать то, что поможет вам улучшить свою жизнь. Не можете принять какую-то позу из-за недостаточной гибкости или перенесенной травмы? Не проблема — скорректируйте позу или выберите другую. Если во время движений вы концентрируетесь на дыхании и прислушиваетесь к своему телу, значит, вы все делаете правильно.

Как я уже сказал, в своей основе современная практика йоги объединяет дыхание, движение и сосредоточенную медитацию. Это не гимнастика, не акробатика и не развитие гибкости. Необыкновенная гибкость или умение выполнять акробатические трюки не делает вас хорошим йогом.

Избавляемся от ложных представлений

Вот несколько опровержений самых распространенных ложных представлений о йоге.

- Для занятий йогой вам не нужно знать тысячи поз. Запоминание большого количества поз не сделает вас хорошим йогом. Сам Кришнамачарья начинал с 24.

- Вам не нужно знать санскритские названия поз — и даже названия на родном языке. Запоминание большого количества поз и их названий не сделает вас хорошим йогом.

- Некоторые приверженцы йоги исповедуют такую религию, как индуизм, однако религиозные убеждения не обязательно укрепляются практикой асаны.

- Некоторые приверженцы йоги придерживаются разного рода вегетарианских диет или практикуют аюрведу, однако диеты не являются необходимым условием для практики асаны.

- Некоторые приверженцы йоги также интересуются альтернативной медициной, астрологией и другими подобными вопросами, но этот интерес не является необходимым условием для практики асаны.

Я разделял все эти заблуждения, когда впервые обратил внимание на йогу. Потратив немного времени на изучение того, что *в действительности* представляет собой практика йоги, я избавился от первоначальных сомнений, а также не стал тратить время на ненужные дополнительные занятия.

Опасна ли йога?

У меня оставалось одно опасение: насколько велика вероятность травмы при занятиях йогой?

Приблизительно в то время, когда я изучал асану, в газете New York Times появилась провокационная статья «Как йога может сломать ваше тело» («How Yoga Can Wreck Your Body») [11]. Ее написал штатный сотрудник отдела науки редакции Уильям Дж. Броуд. В сущности, статья представляла собой выдержки из книги Броуда «Наука йоги: риски и вознаграждение» («The Science of Yoga: The Risks and the Rewards») (2012).

Статья вызвала оживленные споры в мире йоги. Броуд приводил данные нескольких научных исследований о пользе асаны, в большинстве своем связанной с диапазоном движений и эмоциональным равновесием. А затем

переходил в наступление, утверждая, что йога может стать причиной широкого спектра неприятностей, от растяжений до инсульта. Бруд даже предположил, что в нескольких случаях йога стала причиной комы.

Данная статья представляет собой характерный пример полезной *инверсии*. Как будут выглядеть неправильные занятия йогой и каковы главные опасности практики асаны?

Реакция преподавателей йоги на статью и книгу Бруда была быстрой и яростной: судя по всему, Бруд гонялся за сенсацией. Подобно любым физическим упражнениям, йога не свободна от риска, однако при соблюдении мер безопасности вы не нанесете себе серьезного вреда.

Действительно, травмы при занятии йогой не исключены, однако серьезные проблемы со здоровьем, такие как инсульт или кома, встречаются крайне редко, и нет никаких доказательств, что практика асаны повышает риск оказаться в таком состоянии. Статистика свидетельствует, что от других видов физической активности, даже бесконтактных, таких как гольф, йога отличается низкой степенью риска.

Что касается растяжения и даже разрыва мышц или связок, такие типы травм вполне вероятны, если вы заставляете свое тело сложиться в ту или иную позу, не обращая внимания на боль и дискомфорт. Если вы убеждены, что «настоящий» йог может коснуться носом коленей, и принуждаете свое тело сложиться пополам, то рискуете заработать растяжение связок или даже разрыв подколенных сухожилий. Аналогичным образом, если преподаватель йоги силой заставляет вас принять позу, которая не соответствует возможностям вашего тела, это может привести к серьезной травме шеи или спины.

Зная об этих опасностях, я могу принять меры предосторожности — прислушиваться к своему телу, не прикладывать силу, заставляя себя принимать те или иные позы, избегать чрезмерной нагрузки на шейные позвонки — и извлечь из занятий пользу, не подвергая себя риску травм. Это чрезвычайно ценная информация.

Минимальная эффективная асана

Итак, я готов к занятиям... Но с чего начать?

Совершенно очевидно, что существует большое количество видов, форм и философий «настоящей» йоги. Все это очень хорошо, однако я должен вернуться назад и понять, зачем йога нужна именно *мне*.

Почему я выбрал именно ее?

Я хочу вести более активный образ жизни, и мое тело подает ясные сигналы о том, что мне нужно больше двигаться. Врач рекомендовал мне упражнения, которые расширяют диапазон движений, назвав йогой как один из

вариантов. Я читал о пользе физических упражнений и медитации, и мне нравится, что йога сочетает и то и другое. У меня нет времени для посещения студии, поэтому мне хотелось бы заниматься дома, но так, чтобы занятия были эффективными и безопасными.

Всего этого достаточно, чтобы сформулировать целевой уровень мастерства. Вот что у меня получилось.

1. Выучить комплекс поз, которые позволят мне увеличить ежедневную физическую нагрузку, развить силу и общую гибкость.

2. Соединить физические упражнения с техникой дыхания и медитацией, чтобы улучшить не только свое физическое, но и психологическое состояние.

3. Заниматься самостоятельно от 20 до 30 минут, учитывая необходимую силу, гибкость и общий уровень энергии.

Принадлежности для йоги

До начала занятий мне нужно приобрести принадлежности. Первое и самое главное — коврик, который служит своего рода мягким матом и не дает рукам и ногам скользить по полу. Во время занятий йогой можно вспотеть, и, если вы поскользнетесь, это может привести к травме. Потому хороший коврик очень важен.

Когда я занимался в студии Pure Yoga, там были свои коврики, и меня это устраивало. Однако вскоре я заметил, что коврик для меня коротковат. Довольно часто я соскальзывал, и мне приходилось прерывать упражнение, чтобы переместиться к центру. Вероятно, мне лучше найти коврик побольше.

После недолгих поисков я остановился на коврике Manduka PRO [12]. Он удлиненный, совсем не скользкий, прочный и, что тоже немаловажно, приятного зеленого цвета. То, что мне нужно.

Кроме того, я приготовил чехол для коврика и несколько полотенец для рук — на тот случай, если вспотеют ладони. Футболки и несколько пар трико для йоги у меня уже были — так что по части одежды я экипирован. Асаной занимаются босиком, и поэтому специальная обувь не требуется.

По совету Келси я купил длинную хлопковую ленту и бамбуковый опорный блок, которые могут понадобиться для некоторых поз.

Вот и все принадлежности, которые мне необходимы. И это хорошо. Для занятия йогой нужно минимум вещей.

Изучение поз

Моя начальная стратегия сводилась к следующему: Келси — опытный преподаватель йоги, и поэтому есть смысл начать с ее инструкций. В один из дней мы потратили полтора часа на изучение основ.

Не будь у меня в семье преподавателя йоги, я бы взял частные уроки у местного инструктора. Индивидуальные занятия для такого типа обучения очень полезны, поскольку разучить движения по книге трудно.

Но это не значит, что я отказываюсь от книг. Несколько источников информации я уже подобрал.

- «Анатомия йоги» Лесли Каминоффа и Эми Мэтьюз (2011).
- «Йога для здоровья» («Yoga for Wellness») Гари Крафтсоу (1999).
- «Аштанга-йога: Практика и философия» («Ashtanga Yoga: Practice and Philosophy») Грегора Мехле (2007).
- «Дыхательная йога с Лесли Каминоффом» («Breath-Centered Yoga with Leslie Kaminoff») (DVD) Лесли Каминоффа (2010).

Эти книги я использовал главным образом как справочники. Они содержат фотографии и рисунки поз, а также подробные инструкции по каждой позе, так что при необходимости я буду в них заглядывать. После начала занятий я пойму, какая еще информация мне нужна.

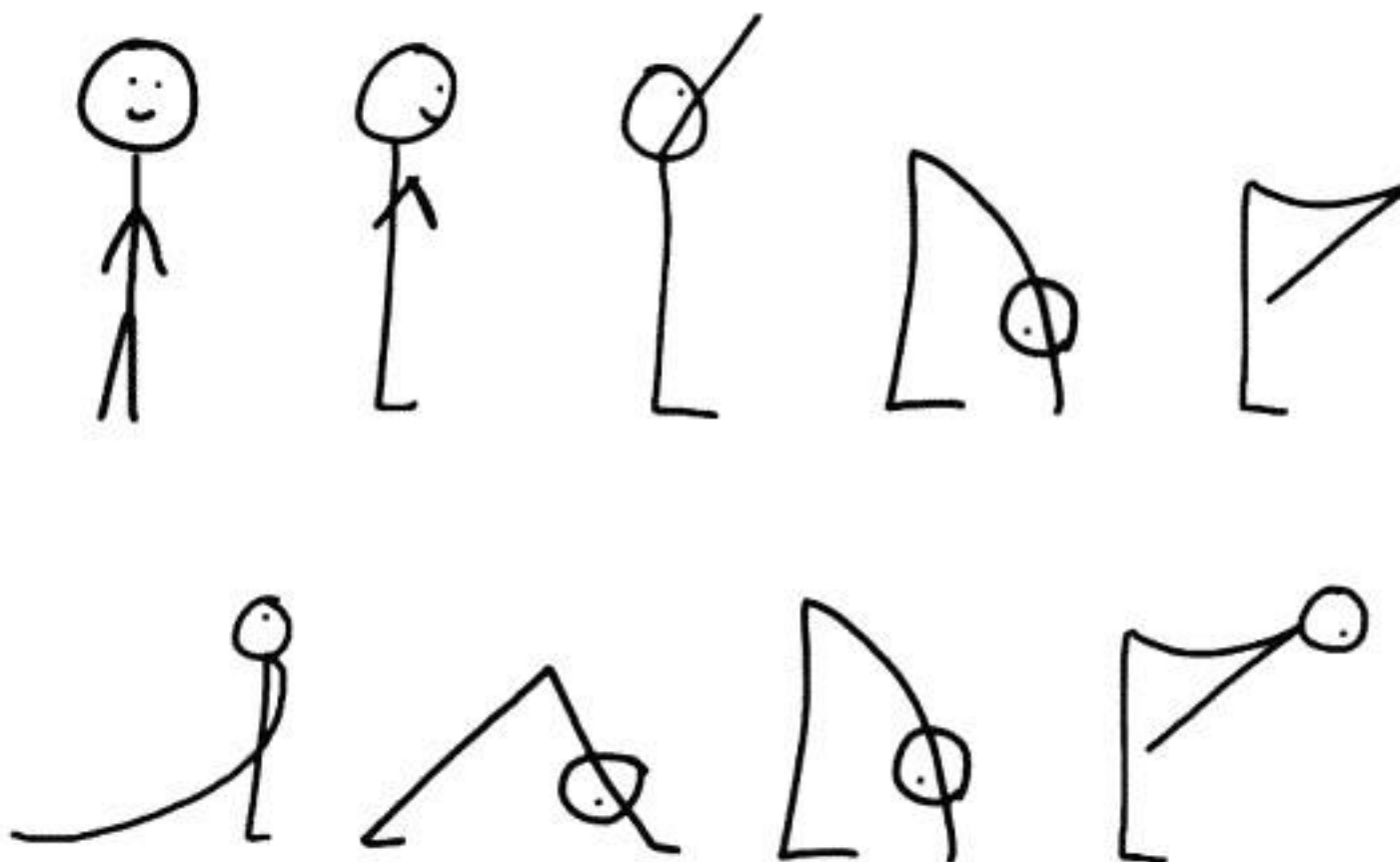
Мы с Келси расстилаем свои коврики и приступаем. Первым делом — «Приветствие солнцу».

Комплекс «Приветствие солнцу»

«Приветствие солнцу» — это последовательность поз, предназначенная для того, чтобы разогреть тело, и большинство практик асаны начинается с нескольких повторений этого комплекса.

Небольшое примечание, если вы решили попробовать это самостоятельно: все люди разные, и в каждом случае «правильная» поза зависит от индивидуальных возможностей. Если вы прислушиваетесь к своему телу, не тянетесь слишком сильно и адаптируете каждую позу к своим возможностям, то практика асаны будет безопасной.

Вот оно, «Приветствие солнцу».



1. Вдохните. Встаньте на переднюю часть коврика, руки расслаблены и опущены вдоль туловища, колени не напряжены.

2. Выдохните. Сложите ладони перед грудью в «молитвенный» жест.

3. На вдохе поднимите руки через стороны над головой, соедините ладони. Посмотрите на ладони (если это трудно, просто посмотрите перед собой).

4. На выдохе наклонитесь вперед и коснитесь ладонями пальцев ног. Наклонитесь как можно ниже, но не прилагайте чрезмерных усилий; колени можно согнуть. Расслабьте плечи и шею.

5. На вдохе посмотрите на стену перед собой, не отрывая ладоней от пола. (Или постарайтесь выпрямить ноги, расслабив мышцы головы и шеи.)

6. На выдохе вернитесь в предыдущее положение.

7. На вдохе сделайте шаг назад и примите положение для «отжиманий», так чтобы тело было полностью выпрямлено и его вес приходился на руки и подушечки пальцев ног. Ноги можно отставлять назад постепенно, а можно прыжком. (Эта поза называется «планка».)

8. На выдохе опуститесь, как при отжимании от пола. Не ложитесь на коврик, а остановитесь в дюйме или двух от пола и замрите. Следите за тем, чтобы предплечья располагались под углом 90 градусов к полу — это позволит

избежать травм. Если поначалу у вас не получается, упритесь коленями в пол. (Эта поза называется «чатуранга дандасана», или просто «чатуранга».)

9. На вдохе поднимите грудную клетку, отталкиваясь от пола и прогибая спину. Одновременно «перекатите» стопы так, чтобы их подъемы уперлись в пол. (Эту позу называют «позой собаки мордой вверх», поскольку она напоминает потягивающуюся собаку.)

10. На выдохе подтяните стопы, так чтобы подушечки пальцев упирались в коврик. Одновременно перемещайте таз вверх и назад, пока ваше тело не примет форму перевернутой буквы V. Расслабьте шею и плечи. Если возникает тянущее ощущение в подколенных сухожилиях, представьте, что вы «дышите ими». Оставайтесь в таком положении в течение трех-пяти дыхательных циклов. (Эту позу часто называют «позой собаки мордой вниз».)

11. Вдохните, затем на выдохе приставьте стопы к ладоням (прыжком или по очереди), так чтобы они оказались под плечами.

12. Повторите шаги 5 и 6.

13. На вдохе поднимите руки через стороны вверх над головой, полностью выпрямив их, как в шаге 3.

14. Опустите руки перед собой, так чтобы они оказались перед грудью в «молитвенном» жесте, как в шаге 2.

Наверное, вы заметили, что цикл начинается и заканчивается одной и той же позой. Завершив последовательность, вы можете повторить ее столько раз, сколько захотите, или перейти к любой другой позе в положении стоя.

Вы обратили внимание, что каждая поза начинается вдохом или выдохом? Это подчеркнуто намеренно: любое движение должно быть согласовано с дыханием. Именно фокусировка на дыхании отличает йогу от аэробики.

Вы можете рассматривать комплекс «Приветствие солнцу» как «минимальную эффективную асану»: при желании все занятия могут состоять из многочисленных повторений «Приветствия солнцу». Эта последовательность поз включает в работу все главные группы мышц тела, требует силы и гибкости и при всей кажущейся несложности является серьезной нагрузкой. Кришнамачарья знал, что делал.

В традиционной практике аштанги «Приветствие солнцу» (комплекс также называют *виньяса*) выполняют между отдельными позами. Это означает, что бо2льшую часть времени двухчасовой «первой последовательности» вы посвящаете именно этому простому комплексу. Вот вам хороший пример степенного закона в действии: бо2льшая часть времени на занятиях тратится на небольшое число поз.

Тем не менее человеческое тело имеет склонность к травмам, возникающим при постоянной нагрузке, и если вы переусердствуете с «Приветствием солнцу», то есть вероятность получить травму [13]. Вот почему

полезно было бы выполнить несколько серий «Приветствия солнцу» в качестве разминки, а затем переходить к разнообразным позам.

Не забывайте дышать как Дарт Вейдер

Во время занятий асаной обычно используют специальную технику дыхания, которую называют *уджайи*. Это простейшая из дыхательных техник (пранаяма), используемых в практике йоги, предназначенная для того, чтобы облегчить фокусировку на дыхании.

В задней части гортани, там же, где расположены миндалины и голосовые связки, есть так называемая *голосовая щель*. Сосредоточившись на этой области, вы можете ее сознательно сузить, ограничив поток воздуха, поступающий в легкие или выходящий из них.

Эффект сужения голосовой щели заметен: вы услышите (и почувствуете) поток холодного воздуха, устремляющегося в легкие при вдохе, и теплого воздуха, выходящего из легких при выдохе.

В «Йога-сутрах» о звуке дыхания уджайи говорится как о «подобном океану», но я предпочитаю современную аналогию — звук уджайи похож на голос Дарта Вейдера из «Звездных войн». Сымитируйте знаменитое дыхание Темного лорда ситхов, а затем закройте рот — получится уджайи.

Древние тексты приписывают уджайи, как и другим техникам пранаямы, мистические свойства. Неизвестно, действительно ли уджайи «создает жар в теле» или «способствует направлению праны (универсальной жизненной силы)», но явно облегчает управление дыханием.

Главная цель уджайи — поддерживать спокойное, расслабленное и равномерное дыхание. Во время занятий уджайи помогает сосредоточиться на дыхании, что не дает вниманию рассеиваться. Если вы следите за дыханием, не обращая внимания на все остальное, значит, вы медитируете — незаметно для себя.

Я освоил самую популярную последовательность поз и основы дыхания приблизительно за 20 минут. Пока все хорошо.

А теперь давайте выучим несколько популярных поз в положении стоя.

Позы в положении стоя

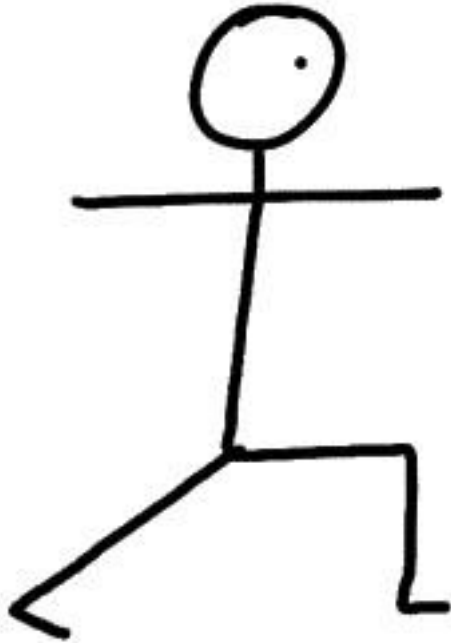
Поза воина 1



Позы воина — это основа аштанги и виньясы. Эти позы часто появляются после (или в середине) «Приветствия солнцу», но их можно выполнять и самостоятельно, объединяя в короткую последовательность.

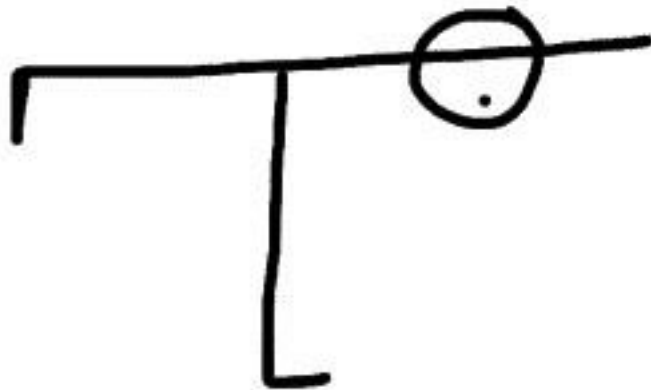
Вот как это делается. Встаньте прямо, смотрите перед собой. На вдохе сделайте шаг назад одной ногой, повернув стопу наружу приблизительно на 45 градусов. Поднимите руки над головой, параллельно друг другу. Передняя нога должна быть согнута в колене, стопа прижата к полу, голень перпендикулярна полу. Стопа задней ноги должна быть прижата к полу. Оставайтесь в этой позе в течение трех-пяти дыхательных циклов.

Поза воина 2



Из позы воина 1 разведите руки в стороны и поверните туловище (если сзади правая нога, то вправо, если левая, то влево). Руки должны быть параллельны полу и как бы «раскрывать» грудную клетку. Оставайтесь в этой позе в течение трех-пяти дыхательных циклов.

Поза воина 3

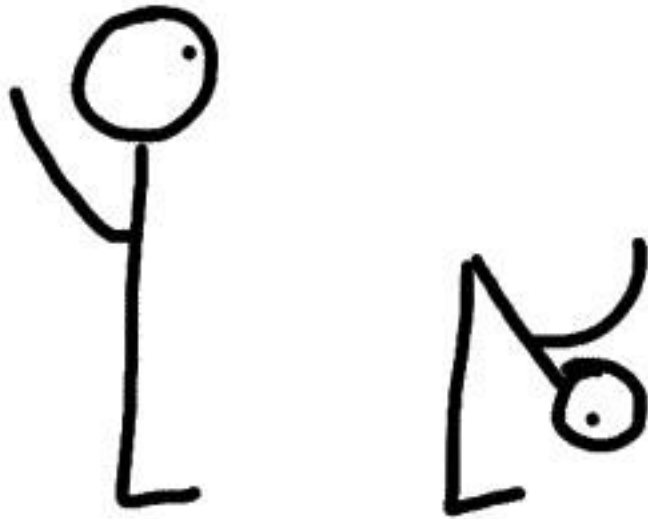


Позу воина 3 можно назвать «позой Супермена». На этот раз ваша цель — наклониться вперед, балансируя на одной ноге, и удерживать тело параллельно полу.

Из нейтрального положения (начальной стойки позы воина 1) вытяните руки над головой, затем слегка приподнимите одну ногу, оторвав ее с пола. Эту ногу отведите назад, одновременно наклоняя туловище вперед, пока не примете позу Супермена. Сосредоточьтесь на удержании равновесия мышцами стопы; колено поднятой ноги должно быть выпрямленным, но не до конца.

Оставайтесь в этой позе в течение трех-пяти дыхательных циклов, затем вернитесь в нейтральное положение.

Спираль Каминоффа



В нейтральном положении стоя сделайте вдох и поднимите руки по спирали в стороны и вверх, как в начале «Приветствия солнцу». Затем на выдохе опустите руки по спирали назад и вниз и наклонитесь вперед, согнувшись в пояснице и приблизив грудь к коленям. Повторите несколько раз, вдыхая на спирали вверх и выдыхая при наклоне.

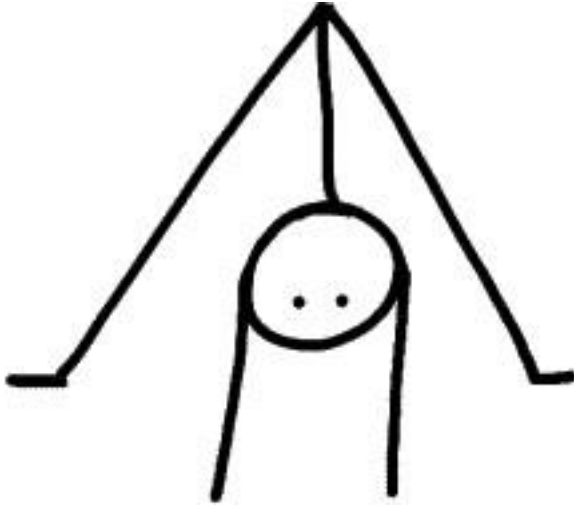
Это моя любимая поза в положении стоя: она очень помогает по утрам, когда встаешь с постели, и после нескольких часов работы за компьютером. Преимущество этого варианта заключается в отсутствии нагрузки на руки, что очень важно, если вы страдаете от кистевого туннельного синдрома, восстанавливаетесь после травм или стремитесь развить силу и гибкость для перехода к более трудным позам.

Наклон вперед



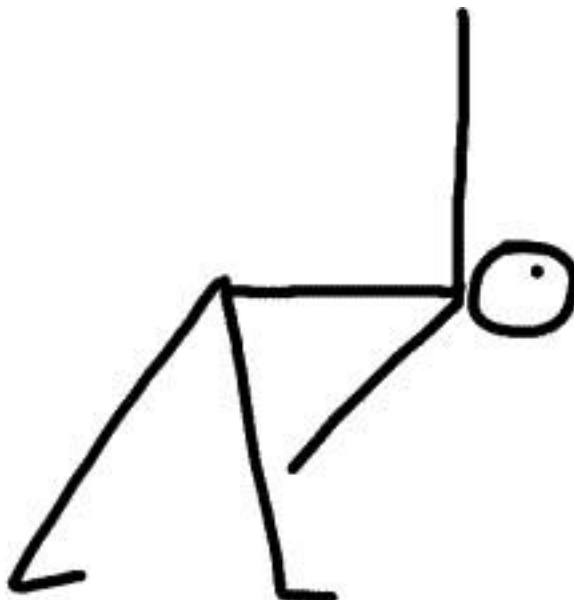
Из нейтрального положения стоя наклонитесь вперед, согнувшись в пояснице; при необходимости колени могут быть слегка согнуты. Расслабьте шею и плечи, опустите голову и постарайтесь коснуться пальцами пола. Попробуйте согнуть и выпрямить колени, отметив положение, где возникает тянущее ощущение. (Если ваши подколенные связки достаточно эластичные, можно коснуться пола ладонями, но усердствовать не стоит.) После трех-пяти дыхательных циклов выпрямитесь в исходное положение.

Наклон вперед с широко расставленными ногами



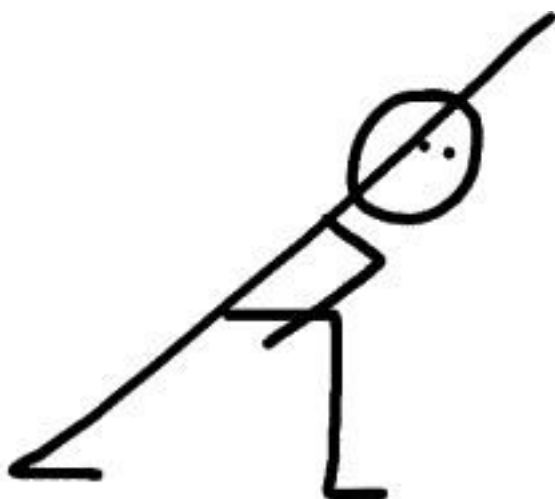
В нейтральном положении стоя широко расставьте ноги. Затем наклонитесь вперед, согнувшись в пояснице. Расслабьте шею и плечи, опустите голову и постарайтесь коснуться ладонями пола. После трех-пяти дыхательных циклов выпрямитесь в исходное положение.

Треугольник



В нейтральном положении стоя широко расставьте ноги; стопы смотрят вперед. Затем повернитесь, так чтобы одна стопа смотрела в сторону. Согните колено этой ноги и упритесь в него локтем. Свободную руку поднимите вверх, к потолку. Наконец, выпрямите колено и нижнюю руку, чтобы она свисала к ноге или к полу. После трех-пяти дыхательных циклов выпрямитесь в исходное положение, затем повторите в другую сторону.

Вытянутый треугольник



В нейтральном положении стоя широко расставьте ноги; стопы смотрят вперед. Затем повернитесь, так чтобы одна стопа была направлена в сторону. Согните колено этой ноги и упритесь в него локтем. Свободную руку поднимите вверх, к потолку, а затем (если сможете) опустите ладонь нижней руки на пол. Руки должны быть вытянуты в одну линию перпендикулярно полу.

Мне нравится другая разновидность вытянутого треугольника: не разводите руки, направляя их в пол и потолок, а сложить их перед грудью в универсальном «молитвенном» жесте. После трех-пяти дыхательных циклов выпрямитесь в исходное положение, затем повторите в другую сторону.

Стул



Встаньте прямо; ноги вместе, руки опущены вдоль туловища. На вдохе поднимите руки над головой. Затем выдохните и согните колени, не разводя ноги, как будто сидите на воображаемом стуле. (Это трудно, причем чем дольше вы остаетесь в этой позе, тем труднее.)

После трех-пяти дыхательных циклов выпрямитесь в исходное положение.

Дерево



Встаньте прямо; ноги вместе, руки опущены вдоль туловища. Оторвите одну ногу от пола, согните в колене. Балансируя на другой ноге, возьмитесь за лодыжку поднятой ноги и прижмите ее стопу к верхней внутренней части бедра; колено должно быть направлено в сторону. Вероятно, вы заметите, что мышцы ноги, на которой вы стоите, сильно напрягаются, чтобы удержать равновесие.

Отпустите лодыжку поднятой ноги и прижмите ладони друг к другу перед грудью в универсальном «молитвенном» жесте. После трех-пяти дыхательных циклов выпрямитесь в исходное положение, затем повторите то же с другой ногой.

Позы на полу

Кошка-корова



Опуститесь на четвереньки; руки точно под плечами, а колени — под бедрами. Расслабьте шею и плечи, направьте взгляд вниз.

На вдохе выгните спину, одновременно втягивая живот. Вы обнаружите, что смотрите на свои ноги. Это поза кошки.

Вдохните, сначала выпрямляя спину в нейтральное положение, а затем выгибая в противоположную сторону, как в начале «Приветствия к солнцу». Теперь вы смотрите вперед, а спина оказывается вогнутой. Это поза коровы.

Следя за дыханием, переходите из одной позы в другую не менее пяти раз.

Наклон вперед из положения сидя



Сядьте на коврик, вытянув ноги перед собой. Стопы напряжены и находятся перпендикулярно полу. Не сгибая коленей, наклонитесь вперед и постарайтесь коснуться пальцами стоп; сгибаться нужно в пояснице, усилием четырехглавых мышц (они располагаются на передней части бедер).

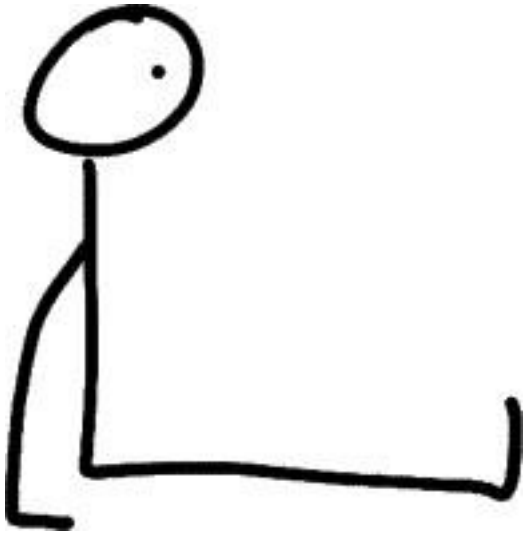
Наклонитесь как можно ниже, но так, чтобы не чувствовать боли или сильного дискомфорта. Оставайтесь в этом положении в течение трех-пяти дыхательных циклов.

Наклон вперед к одной ноге



Рассматривайте эту позу как комбинацию позы дерева и наклона вперед. Поднимите одну ногу, согните в колене и поверните в сторону. Прижмите стопу к верхней внутренней части другой ноги и наклонитесь вперед, согнувшись в пояснице, как при наклоне вперед из положения сидя. Наклонитесь как можно ниже, но без боли или сильного дискомфорта, и оставайтесь в этом положении в течение трех-пяти дыхательных циклов.

Посох



Эта поза труднее, чем кажется. Сядьте на коврик, вытянув ноги прямо перед собой. Стопы напряжены, перпендикулярны полу. Упритесь ладонями в пол и слегка приподнимитесь, стараясь, чтобы стопы были направлены вертикально вверх. Оставайтесь в этом положении в течение трех-пяти дыхательных циклов.

Связанный угол



Сядьте на пол, подтяните колени к груди. Затем разведите колени в стороны, чтобы они опустились к полу, и соедините подошвы стоп. Обхватите стопы ладонями и оставайтесь в этом положении в течение трех-пяти дыхательных циклов.

Скручивание позвоночника в положении сидя



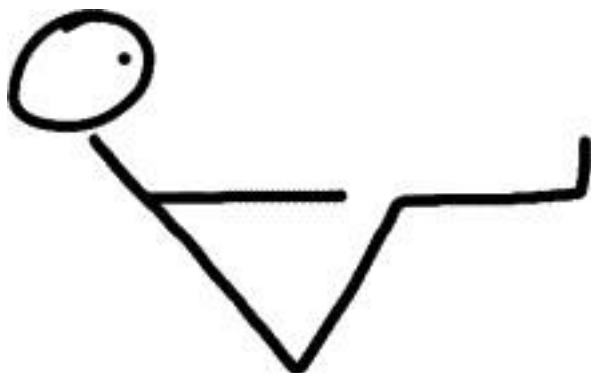
Сядьте на коврик, вытянув ноги прямо перед собой. Подтяните одно колено к груди, так чтобы стопа приблизилась к колену другой ноги. Затем обхватите ладонями стопу и поставьте ее по другую сторону вытянутой ноги. Ладони положите на пол возле бедер, как в позе посоха.

Из этого положения подведите локоть противоположной руки к колену согнутой ноги. (Если согнута правая нога, то локоть левой руки.) Согните эту руку в локте, чтобы пальцы были направлены в потолок. Продолжайте движение, подталкивая ногу рукой. Завершите скручивание поворотом головы в ту же сторону.

Особенность упражнения состоит в том, что вы заметите разницу в скручивании разных отделов позвоночника. По утверждению Лесли Каминоффа, поясничный отдел способен поворачиваться только на 5 градусов. Угол поворота грудного отдела позвоночника составляет 35 градусов, а шейного — от 80 до 90. Таким образом, при полном скручивании вы можете посмотреть назад.

Оставайтесь в этом положении в течение трех-пяти дыхательных циклов, вернитесь в исходное положение, затем повторите в другую сторону.

Лодка



Это трудная поза. Сядьте на коврик, вытянув ноги прямо перед собой. Согните колени, балансируя на ягодицах; пальцы ног должны касаться пола. Вытяните руки прямо перед собой и, прижав ноги друг к другу, приподнимите их над полом. Сохраняйте равновесие. Лодыжки соприкасаются, спина прямая. Оставайтесь в этом положении в течение трех-пяти дыхательных циклов, затем вернитесь в исходное положение.

Ворона



Сядьте на коврик, упритесь ладонями в пол. Упритесь локтями с внутренней стороны коленей, наклонитесь, чтобы центр тяжести сместился вперед, и оторвите стопы от пола. Задержитесь в этом положении, сколько сможете; дышать старайтесь равномерно.

Колесо



Лягте на коврик на спину. Поднимите колени, чтобы стопы уперлись в пол. Одновременно упритесь ладонями в пол рядом с ушами. На вдохе оттолкнитесь руками и ногами, чтобы туловище приподнялось над полом, образовав дугу. Оставайтесь в этом положении в течение трех-пяти дыхательных циклов, затем медленно опуститесь на пол.

Корректировка трудности

Вы заметили, что все позы обычно заканчиваются тем, что вы остаетесь в них в течение трех-пяти дыхательных циклов? Именно дыхание делает йогу уникальной, и это важная составляющая каждой позы. От трех до пяти дыхательных циклов уджайи — этого вполне достаточно для начинающих.

Опытные любители йоги не обязательно осваивают более сложные позы или большее количество поз: они выполняют упражнения медленно, тщательно контролируя себя, и дольше остаются в каждой из поз. Когда я приобрету опыт, то продлю время пребывания в каждой позе на несколько дыхательных циклов.

Шавасана



В традиции йоги *шавасана* считается конечной целью всех движений и дыхательной техники. Не забывайте, что первоначально цель асаны состояла в подготовке к медитации. Эта традиция продолжилась в современной практике, и на то есть веская причина.

Шавасана — это полное снятие мышечного напряжения в сочетании с глубокой медитацией. Ее часто называют «позой трупа», но мрачное название не должно вас отпугивать. Сама по себе поза очень проста: лягте на спину на коврик, ноги выпрямлены, руки вдоль туловища. Закройте глаза и расслабьте все тело. Мышцы вообще не должны работать.

Проверьте свое тело, от пальцев ног до макушки, и, если заметите напряженные мышцы, расслабьте их. Это касается и тех групп мышц, которые мы обычно не замечаем, например шеи или даже языка. Теперь вы отказываетесь от дыхания уджайи и начинаете дышать нормально.

Ощущение такое, словно вы таете и растекаетесь по полу. После длительного периода, когда ваши мышцы пребывали в постоянном движении, шавасана приносит облегчение. Нагрузка на тело, похоже, помогает очистить разум, поэтому в позе шавасана медитировать легче.

Оставайтесь в позе шавасана около пяти минут, затем встаньте. Ваши занятия асаной закончены.

Применение метода

На данном этапе я достиг целевого уровня мастерства. После трех часов инструктажа я могу заниматься йогой дома, эффективно и безопасно.

Давайте кратко опишем суть метода, который я использовал для обучения йоге.

- Я приобрел необходимые принадлежности: коврик, ленту и опорный блок.
- Я потратил несколько часов на занятия с опытным инструктором, чтобы овладеть основами и избавиться от серьезных ошибок.
- Я понял, что практика асаны представляет собой последовательность поз в сочетании с дыханием и внутренней сосредоточенностью.
- Я выучил основную последовательность поз виньясы («Приветствие солнцу»), 10 наиболее распространенных поз в положении стоя и 10 наиболее распространенных поз в положении сидя.
- Я научился предохраняться от травм, прислушиваясь к своему телу, при необходимости корректируя занятия, избегая чрезмерного растяжения, а также чрезмерной нагрузки на позвоночник.
- Я упражнялся до тех пор, пока не выучил наизусть последовательность поз, выполнение которой занимает около 25 минут.

Планы на будущее

В данный момент я удовлетворен своими занятиями йогой. После трех часов исследований и инструктажа я способен заниматься йогой, когда хочу или когда необходимо, не посещая занятий в студии. 25 минут утром или вечером — и дело сделано.

Мне нравится, что в йоге физические упражнения сочетаются с медитацией. После занятий улучшается не только мое физическое самочувствие, но и психологическое состояние. Продолжая заниматься, я извлеку пользу как из медитации, так и из широкого диапазона движений.

Но тут есть одна трудность: польза возможна только при регулярных занятиях. Главное — выделить время для утренних или вечерних занятий. Я изменю свой распорядок дня и теперь по утрам не буду первым делом проверять электронную почту или приступать к работе, а займусь йогой, прежде чем приниматься за дела.

Забота о своем теле перейдет в разряд наивысших приоритетов. Йога будет поддерживать мою физическую форму на протяжении следующих

десятилетий.

Мне не нужно становиться монахом или акробатом, чтобы насладиться пользой от йоги. Мне не нужно знать все позы, не обязательно уметь закидывать ногу за голову. Мне нужно лишь выполнять основные упражнения, выучить которые совсем не трудно.

Более того — я освоил все, что мне было нужно, всего за три часа, что гораздо меньше, чем я ожидал. Я приступил к этому проекту в убеждении, что придется изучать длинные комплексы сложных поз, но выяснилось, что в этом нет никакой необходимости. Я мог достичь желаемого при помощи упрощения, что позволило получить нужный результат за более короткое время.

Когда речь идет об усвоении чего-то нового, нет никакого смысла без необходимости усложнять себе задачу.

5

Программирование

Урок: сложные вещи становятся простыми, если их разбить на части

Если отладка — процесс удаления ошибок, то программирование должно быть процессом их внесения.

Эдсгер Дейкстра, известный ученый, специалист по компьютерам

Я зарабатываю в интернете с 2007 года, с тех пор как оставил работу на полную ставку в качестве менеджера по маркетингу и основал собственную компанию, специализирующуюся на издательских и консультационных услугах.

Мой первый сайт, PersonalMBA.com, дает мне средства к существованию: в сущности, я обучаю науке бизнеса, хоть и не преподаю в университете. Каждый год я пополняю список написанных мною книг, которые посвящены законам бизнеса и предназначены для читателей, желающих овладеть его основами [1].

С момента первой публикации в 2005 году список рекомендуемой литературы из книги «Сам себе MBA» оставался одним из самых популярных, а дополнения к этому списку делают посетители сайта со всего мира. С 2005 года сайт PersonalMBA.com посетили более 2 миллионов читателей.

Моя первая книга «Сам себе MBA» была естественным развитием сайта PersonalMBA.com и стала международным бестселлером. Издание книги предполагает также и ее распространение, и поэтому я на протяжении нескольких лет изо всех сил старался привлечь новых читателей.

О книге «Сам себе MBA» упоминалось в New York Times, Wall Street Journal, Fortune, Forbes, FastCompany, на многих популярных сайтах, в блогах. И после каждого такого упоминания на PersonalMBA.com приходили тысячи новых посетителей.

Цена прогресса

Тысячи посетителей вашего сайта — это здорово, но при условии, если все они имеют возможность одновременно на него попасть. Моя проблема состоит именно в этом: каждый раз, когда трафик сайта значительно увеличивается и достигает определенного порога, сайт рушится в зените своей славы и популярности, оставляя посетителям лишь сообщения с кодом ошибки.

Вот типичный пример: популярный блог Liferhacker.com, посвященный программному обеспечению и компьютерам, за последние семь лет три раза помещал список литературы «Сам себе MBA». И каждый раз тысячи людей одновременно пытались зайти на сайт PersonalMBA.com, перегружая мой веб-сервер (компьютер, предоставляющий веб-страницу по запросу пользователя). Вместо запрашиваемой информации сервер выдавал сообщение «Ошибка установки связи с базой данных» или «Ошибка 503», что эквивалентно просьбе о пощаде в цифровой форме.

И каждый раз, когда сервер падал из-за перегрузки, вместе с ним умирала частичка моей души. Время, потраченное на маркетинг моего сайта, оказывалось израсходованным впустую. В конечном итоге мне удалось заинтересовать своим предложением тысячи любознательных людей, но из-за рухнувшего сервера они уходили разочарованными, с пустыми руками. Маркетинг оказывался слишком хорош, и система не справлялась с внезапно возросшими требованиями.

Изучение проблемы

Моя первая реакция — усилить сервер, добавив вычислительной мощности и памяти. Это помогло, но лишь до некоторой степени. При превышении определенной нагрузки мой сайт снова рушился, что совпадало с каждым серьезным успехом в маркетинге.

В то время PersonalMBA.com использовал популярную систему управления сайтами под названием WordPress [2]. Эта система оптимизирована под легкость установки и использования, а не под работу с большими нагрузками. В стандартной конфигурации WordPress каждый запрос веб-страницы запускает цепочку действий сервера, генерируя сотни внутренних запросов, чтобы предоставить каждую страницу пользователю.

Такой процесс делает каждый отдельный запрос веб-страницы «затратным». То есть выполнение каждого запроса требует выделения существенного объема памяти и вычислительной мощности. Если один посетитель сайта просматривает пять страниц, то он генерирует пять затратных запросов. Если тысяча посетителей сайта одновременно обращаются к одной и той же странице, то сервер попытается запустить тысячу идентичных затратных процессов одновременно.

Система зависла

В такой ситуации бедный осаждаемый запросами сервер попытается ответить на все запросы, но поскольку каждый из них требует значительных ресурсов, память сервера будет исчерпана раньше, чем он справится со всеми ими. В этот момент сервер выбрасывает белый флаг — и посетители сайта остаются ни с чем.

В попытке решить эту проблему я пять раз менял компании, предоставляющие услуги веб-хостинга, и потратил сотни часов, обучаясь конфигурировать серверы WordPress таким образом, чтобы они не отключались при большой нагрузке. Каждая новая конфигурация усложняла сервер, и каждая новая инсталляция требовала все больше времени на ее поддержку.

В конечном итоге я разрабатывал пользовательские конфигурации сервера, требовавшие длинных цепочек загадочных системных команд для установки, настройки и изменения сложных серверных приложений, которые я с трудом понимал. Устранение каждой ошибки или поиск ответов на вопросы, встречающиеся в процессе установки и поддержки сервера, требовали многочасовых поисков.

И в довершение всего популярность и относительная уязвимость WordPress сделали систему любимой мишенью для хакеров и спамеров. Приблизительно раз в неделю какой-нибудь хакер находит новую уязвимость, а затем использует ее для того, чтобы заполнить спамом миллионы блогов WordPress. Поддержание безопасности системы и своевременное обновление программного обеспечения — это уже работа в полную смену, особенно если вы поддерживаете несколько сайтов. (Я поддерживал 12 сайтов одновременно и катастрофически не справлялся.)

В какой-то момент мне стало ясно, что я трачу больше времени на обеспечение работоспособности сайтов, чем на материалы для своих читателей. Это было бессмысленно. Не только из-за того, что уменьшалась полезная отдача, но еще и потому, что я до этого не обучался программированию. Я узнавал лишь набор ситуативных приемов и «заплаток», применимых конкретно к системе WordPress. Не самый эффективный вариант [3].

Я решил поискать другой способ поддержки своих сайтов, и находка многообещающей альтернативы не заняла много времени.

Потенциальное решение

Однажды я наткнулся на Jekyll [4], программу управления сайтами, созданную Томом Престоном-Вернером, который известен как основатель социальной сети для программистов GitHub [5]. Система Jekyll предназначена для того, чтобы заменить WordPress, обеспечив простое управление сайтами без затратных запросов.

Представьте, что у вас есть сотни текстовых документов, содержащих важную информацию, и вам нужно их унифицировать — одинаковый шрифт, одинаковый стиль заголовков и так далее. Если вы напишете программу, которая автоматически отформатирует любой файл нужным образом (вместо того чтобы вносить изменения вручную), это сэкономит вам уйму времени.

Именно так и поступает Jekyll с веб-страницами. Запустите одну команду — и программа создаст законченный сайт, используя те файлы на вашем компьютере, которые содержат нужную информацию и шаблоны. Если необходимо изменить дизайн или содержание страницы, вы просто снова запускаете Jekyll, и изменения автоматически вносятся в весь сайт, экономя ваше время и силы.

Jekyll предлагал многообещающие возможности. Теоретически я мог заменить WordPress просто папкой текстовых файлов на своем компьютере. Мой сайт будет быстрым, как молния, необыкновенно стабильным, и я сэкономлю больше ста часов в год, которые трачу на поддержание сервера.

Однако тут меня поджидала одна ловушка: система Jekyll написана на языке программирования Ruby, которого я не знаю. Я понятия не имею, как писать программы на Ruby или запускать приложения Ruby, предназначенные для пользователей. Кое-что из того, что мне необходимо для управления сайтом PersonalMBA.com, требует большего, чем простое форматирование.

Чтобы управлять сайтами с помощью системы Jekyll, мне нужно научиться программировать, а также запускать программы веб-приложений, написанные на Ruby.

Похоже, я нашел привлекательную задачу.

Научиться программированию

Я уже довольно давно хотел научиться писать программы, однако на первый план всегда выходили другие проекты. Если я научусь программировать, то мои возможности в бизнесе существенно расширятся, поскольку весь мой бизнес связан с интернетом.

Важно отметить, что все, чем я занимался до сих пор, — это не программирование. Языки HTML и CSS, которыми я пользовался для

разработки веб-страниц, называются «языками разметки». Коды HTML и CSS просто указывают компьютеру, в каком виде отобразить текстовый файл (то есть «этот текст отобразить жирным шрифтом» или «данный раздел — основной заголовок, шрифт 24 пункта»).

То же самое можно сказать о моих упорных попытках усовершенствования сервера. Конечно, я составлял какие-то программы, но настоящим программированием это назвать нельзя. Я просто устанавливал готовое программное обеспечение, а затем изменял его настройки. Приложения, которые я использовал, писали профессиональные программисты, и для того чтобы их применять, мне не нужно было уметь программировать. Конфигурирование сервера и его поддержка — полезные навыки, но это не программирование.

Что же такое программирование?

Первое, что мне пришло в голову: программирование — это указание, какие вещи должен делать компьютер. Однако такое определение было слишком расплывчатым и бесполезным. Что делать? Какие «вещи»?

10 лет назад в колледже я прослушал два курса по основам программирования и поэтому немного смутился, когда понял, что не могу дать четкого определения этой дисциплине. Я помню некоторые из основных терминов, например *переменная*, *цикл*, *ввод*, *вывод*, *функция*, *объектная ориентация*, а также нечто под названием *пузырьковая сортировка*. Вот, пожалуй, и все.

Я должен был научиться писать программы на языке C++ и помню, как расстраивался, когда много часов искал пропущенную точку с запятой, из-за которой не работала моя программа. Я также помню слова преподавателя: «Вы никогда не будете использовать пузырьковую сортировку в реальных приложениях, но мы все равно будем ее изучать».

В процессе учебы я написал несколько простых программ, по необходимости: цель состояла в том, чтобы получить хорошую оценку, и я ее получил. К сожалению, программы, которые мы писали, были неприменимы в реальной жизни, а с тех пор я не имел дела с программированием. Несмотря на то что в моей памяти застряли несколько терминов, начинать придется с самого начала, с понимания основ.

Поскольку я не могу точно описать свои намерения, когда выражаю желание «научиться программировать», то предвижу трудности с определением целевого уровня мастерства. «Написать компьютерную программу» — это уже чуть конкретнее, но все равно бесполезно.

Пришла пора суммировать все, что я в данный момент знаю о

программировании.

- Я знаю, что программисты «пишут» программы, что предполагает творческую задачу, которую можно решить разными способами.

- Программы часто называют «приложениями», и эти термины взаимозаменяемы.

- Когда компьютерные программы «запускаются» или «выполняются», то делают то, для чего они были написаны.

- Термины «вход» и «выход» запомнить легко, поскольку это обычные слова. Под входом понимают информацию или данные, которые использует программа, а под выходом — результат, получаемый по завершении ее работы.

- «Переменная» — это место для заполнения тем, что меняется. Вы можете создавать сколько угодно переменных, по своему желанию, и обозначать ими все что угодно.

- Сама «программа» представляет собой подробный набор инструкций и правил, которые указывают компьютеру, что нужно делать с входными данными. Когда программа заканчивает свое выполнение, вы получаете выходные данные.

- Программа «виснет» или выдает сообщение об ошибке, когда что-то идет не так и компьютер не может понять, что делать дальше.

Ну вот, уже кое-что. Получилось довольно грубое разбиение на элементы: вместо «программирования» мы теперь имеем три подчиненных понятия.

- **Вход** — информация, которая используется для выполнения процесса.

- **Процесс** — последовательность шагов, предпринимаемых программой в соответствии с входными данными.

- **Выход** — конечный результат выполнения программы.

Такое разбиение очень полезно. «Написать компьютерную программу» означает следующее: определить информацию, с которой вы хотите начать, установить последовательность шагов, точно описывающую действия компьютера с входными данными, а также определить выходные данные, предоставляемые вам компьютером «на выходе».

Представьте себе блок-схему, которая может служить полезной мысленной зацепкой, помогающей понять работу программы. Вы начинаете процесс с входных данных. Затем производите определенные действия в зависимости от выполнения тех или иных условий. Процесс заканчивается по достижении конца блок-схемы. Вы остаетесь с выходными данными: конечным результатом процесса, описанного блок-схемой.

Создание компьютерной программы аналогично разработке блок-схемы.

Вы задаете те же вопросы:

- С чего я начинаю?
- С чего начинается процесс?
- Что происходит после этого? А дальше?
- Когда процесс заканчивается?
- Что я получу, когда процесс закончится?

Блок-схемы описывают ответы на эти вопросы в визуальной форме.

Программы описывают то же в виде текста, однако суть от этого не меняется.

Аналогия с блок-схемой полезна также потому, что помогает объяснить и другие важные понятия.

Условные операторы — это выражения такого вида.

- Если X истинно/ложно, то делать Y .
- Если X не Y , то делать Z .
- Когда X истинно/ложно, то делать Y .
- Когда X не Y , то делать Z .
- Пока X истинно/ложно, делать Y .
- Пока X не Y , делать Z .

В данном случае X , Y и Z — это *переменные*, которые могут обозначать все что угодно. Переменные могут обозначать числа, как в элементарной алгебре, или слова. Иногда переменные — это отдельные буквы или символы, а иногда целые слова. В любом случае они обозначают объекты, с которыми мы работаем.

Условные операторы (ЕСЛИ, ТО, КОГДА И ПОКА — IF, THEN, WHEN и WHILE) подобны стрелкам с вопросами на блок-схеме. Представьте себе, что вы за рулем автомобиля: ЕСЛИ (IF) сигнал светофора красный, ТО (THEN) вы останавливаетесь. КОГДА (WHEN) сигнал светофора сменяется на зеленый, ТО (THEN) вы едете. КОГДА (WHEN) сигнал светофора желтый, ТО (THEN) вы тормозите, готовясь остановиться.

Имеет смысл остановиться на этих условных операторах немного подробнее, поскольку здесь есть несколько общих закономерностей. Условие истинно/ложно встречается довольно часто, а оператор ПОКА (WHILE) указывает не на одно конкретное действие, а на продолжение тех или иных действий.

Во всех случаях условный оператор содержит высказывание, которое определяет, нужно ли предпринимать какие-либо действия. Это высказывание называется *условием* и может принимать разные формы. Иногда условие представляет собой простое сравнение типа истинно/ложно (Сигнал светофора красный?), иногда математическое сравнение (X больше 100?), а иногда

содержит логическую конструкцию (Сигнал светофора НЕ красный?).

Назначение условия — определить, будет ли выполняться соответствующий процесс. Если условие истинно или достоверно, программа выполняет определенную команду. Если нет, то эта команда пропускается, и программа переходит к следующей.

Переменные типа истинно/ложно называются *булевыми переменными* — это красивое название обозначает то, что может принимать всего два значения. Да/нет и включено/выключено тоже относятся к булевым переменным. Булевы переменные играют большую роль в программировании, поскольку являются основой как простейших процессов (подобно истинно/ложно в блок-схемах), так и для включения/выключения крошечных электронных переключателей, из которых состоит компьютер.

В данном случае ПОКА (WHILE) — это особый тип условного оператора, который называется *циклическим*. Циклический оператор заставляет процесс повторяться до тех пор, пока не будет выполнено условие. Вернемся к аналогии управления автомобилем: ПОКА (WHILE) сигнал светофора красный, ехать нельзя.

Все очень просто, правда? Если провести аналогию между написанием компьютерной программы и составлением блок-схемы, этот процесс будет довольно легко представить.

И еще одно, последнее замечание: что будет, если компьютер зайдет в тупик, не зная, что ему делать, или команды, которые компьютер пытается выполнить в какой-то момент, не работают или не имеют смысла? Что произойдет?

В таком случае программа «вылетает» — останавливается и нередко вместо ожидаемого результата выдает сообщение об ошибке. Мы все знакомы с внушающим страх «синим экраном смерти» Windows или с сообщением «Ошибка 404: веб-страница не найдена» при поиске в интернете. Иногда происходит непредвиденное, компьютер «сходит с ума» и зависает.

Ваша задача как программиста предотвратить зависание и ошибки программы. Лучший способ — убедиться, что программа всегда имеет информацию, чтобы завершить процесс так, как планировалось, но это не всегда возможно. В тех случаях, когда неопределенности не избежать, полезно предусмотреть способ восстановления, если программа не может завершить процесс.

Эти операторы восстановления после ошибки называются *исключениями*, и они очень полезны. Их можно считать условными операторами, связанными с ошибкой: ЕСЛИ (IF) программа вылетает по ветке X, ТО (THEN) делать Y вместо X.

Исключения немного похожи на резервный генератор для больничного оборудования. Большая часть времени генератор просто присутствует, но не

работает. Но если электроснабжение прервется, генератор включится, и больница будет использовать электроэнергию, вырабатываемую генератором, а не погрузится в полную темноту. Это спасет жизнь пациентам, подключенным к аппаратуре поддержания жизнедеятельности, которой требуется бесперебойное электропитание. Полный отказ опасен, и потому резервные планы очень важны.

Это и есть основы программирования. Определить входные данные. Ввести переменные. Создать процессы, которые ведут к желаемому результату. Рассмотреть эти процессы как блок-схему, добавив при необходимости условные операторы и исключения. Если все идет хорошо, ввести входные данные, запустить программу и получить желаемый результат.

Это сильное упрощение чрезвычайно сложного занятия, но достаточно подробное для того, кто впервые сталкивается с программированием. Разбив процесс создания программы на составляющие, легче понять, с чего нужно начинать.

Языки программирования

Тут возникает еще одна трудность: компьютер не разговаривает так, как люди. В сущности, работа компьютера основана на переключении крошечных электронных переключателей. Если у компьютера нет какого-либо способа перевести наш человеческий язык в команды переключения электронных элементов, он не сможет выполнить нашу команду.

Именно для этого и предназначены языки программирования: они предлагают программисту конкретный способ объяснить компьютеру, когда нужно начинать работу, что делать и когда остановиться. Они также позволяют программисту определить входные данные, процессы, выходные данные и действия по завершению программы.

В каждом языке программирования определен свой способ написания команд, который называется *синтаксисом*. Синтаксис языка включает правила, которые компьютер использует для перевода строк программы в команды крошечным электронным переключателям.

Практически в каждом языке предусмотрен способ описания переменных, условных операторов и исключений. Детали разнятся, но суть одна и та же.

Думать как программист

Программисты часто обдумывают проблемы с помощью так называемого *псевдокода*: языка, напоминающего язык программирования, но недостаточно конкретного, чтобы его команды исполнял компьютер. Его можно

рассматривать как эскиз. Псевдокод помогает обдумать процесс решения задачи.

В примере, описывающем поведение водителя за рулем, я использовал псевдокод. Если сесть в машину и вслух произнести: «КОГДА я поверну ключ зажигания, ТО двигатель запустится», — ничего не произойдет. Но это не значит, что мое утверждение бессмысленно. Оно представляет собой способ обдумать шаги, необходимые для разрешения проблемы или получения желаемого результата.

Вы можете использовать основные понятия программирования, которые мы только что рассмотрели, для создания набросков программ, позволяющих решить повседневные задачи.

Вот забавный способ проверить это на себе. Попросите приятеля помочь вам решить простейшую задачу под названием «сделать сэндвич». Единственное правило — приятель должен лишь точно выполнять ваши инструкции, не больше и не меньше. Он ничего не знает и все ваши указания воспринимает буквально.

Вы: Возьми хлеб.

Приятель: Я не понимаю, что такое «возьми».

Вы: Протяни руку туда, куда я скажу, и схвати его.

Приятель: Я не понимаю, что такое «рука».

Вы (вдыхая): Вот эта штука. (Вы указываете на руку приятеля.)

Приятель: Понял.

Вы: Протяни руку туда, куда я скажу, и схвати его.

Приятель: Я не понимаю, что такое «схвати».

Вы: Согни пальцы вот так. (Вы демонстрируете, сгибая и разгибая пальцы.)

Приятель: Понял.

Вы: Протяни руку к хлебу и схвати его.

Приятель: Я не понимаю, что такое «хлеб».

Вы: Вот эта штука, вот здесь! (Вы указываете на хлеб.)

Приятель: Понял.

Вы: Протяни руку к хлебу и схвати его.

Приятель: (Протягивает руку к хлебу, сжимает пальцы и разжимает. Хлеб остается на месте.)

Вы: ИДИОТСКАЯ ИГРА!!!

Это глупый пример, но именно так выглядит программирование, особенно в самом начале.

Компьютер, подобно вашему приятелю, не может понять то, что вы в точности не определили. Любой сложный процесс, который вы пытаетесь

описать, нуждается в полном и однозначном определении.

Именно в этом и заключается сложность программирования: единственная неточная или ошибочная команда делает всю программу нерабочей. В этом отношении программирование не терпит компромиссов — ваша программа либо корректная, либо некорректная. Точность нужна в каждой детали.

Компьютеру безразлично ваше очарование и чувство юмора. Если программа неполная или плохо написана, то ваше приложение зависнет, и вы потеряете данные. Или вы напишете *баг*: фрагмент неверного кода, который вызовет неожиданные и непредсказуемые последствия. Тут как в математике — либо работает, либо нет. Никто не поставит вам высокую оценку за усердие.

При всем при том в программировании не существует единственного, универсального решения поставленной задачи — так же как и в математике. Есть тысячи способов получить из ваших входных данных желаемый результат. Программист должен выработать конкретный подход на основании имеющихся в его распоряжении средств.

Привыкнув к тому факту, что компьютер не способен читать ваши мысли, вы постепенно учитесь давать инструкции, которые система может понять — точно так же, как вы учились точно определять термины и описывать простейшие действия приятелю в игре «Сэндвич», прежде чем отдавать сложные команды.

В чем особенность программирования интернет-приложений?

Теперь у нас есть рабочее определение программирования. Оно предельно упрощено, однако позволяет понять, что мы пытаемся сделать.

Как бы то ни было, меня не интересуют все разновидности программирования — я хочу писать программы работы с сайтами. Если вы когда-либо пользовались программами электронной почты, например Gmail, Hotmail, Yahoo Mail или любой другой, то понимаете, о чем идет речь. Эти программы выполняются в вашем интернет-браузере. Чтобы использовать их, не нужно загружать программное обеспечение в компьютер. Вы просто указываете нужный сайт в браузере и регистрируетесь. Можно приступать.

Это очень важное отличие программного обеспечения, которое выполняется *локально*, от того, что выполняется *удаленно* на интернет-сервере. Чтобы разработать интернет-приложение, необходимо сначала написать его, а затем протестировать, убедившись в его работоспособности. Весь процесс разработки и тестирования выполняется на вашем компьютере.

После того как программа заработала, вы посылаете ее на «рабочий» интернет-сервер, где к ней получают доступ другие люди. Пользователи

интернета не могут зайти на ваш компьютер из сети, поэтому загрузка программы на интернет-сервер открытого доступа является необходимым условием, чтобы вашим приложением могли пользоваться другие.

Это значит, что процесс разработки делится на два основных этапа: локальное программирование плюс тестирование — и загрузку программы на удаленный рабочий сервер для реального применения. Я должен выяснить, что представляют собой оба этих этапа.

Из предыдущего опыта работы с HTML и CSS я сделал один вывод: эти языки «немые» в том смысле, что вы не можете заставить главную веб-страницу хранить какую-либо информацию.

Допустим, у вас есть файл веб-страницы, содержащий приветствие «Привет всем!» [6], и вы хотите заменить слово «всем» именем пользователя, который посещает страницу. Отличная идея, но главные веб-страницы не способны хранить информацию, которую можно впоследствии воспроизвести. Они просто воспроизводят текст из файла, а этот файл не позволяет себя изменять.

Это свойство описывается таким техническим термином, как *состояние*. Главные веб-страницы, созданные с помощью HTML и CSS, не запоминают состояние и поэтому называются ресурсами «без запоминания состояния». На веб-странице вы можете создать окно для ввода имени пользователя с кнопкой «Сохранить», но для сохранения этой информации необходимо предусмотреть для нее специальное место.

Вот почему интернет-приложения используют для сохранения информации два подхода: *базы данных* и *куки-файлы*.

Понятие базы данных лучше всего иллюстрирует стопка учетных карточек. Предположим, вы хотите создать адресную книгу, в которой указаны имя, номер телефона, адрес электронной почты, пол и возраст всех ваших друзей.

Каждый друг получает свою учетную карточку, в которой записывается информация о нем. Если кто-то из друзей, к примеру, меняет адрес электронной почты, вы можете стереть старую информацию с карточки и заменить ее новой. Взглянув на любую учетную карточку, вы видите всю информацию сразу.

Стопка учетных карточек аналогична базе данных. Каждая карточка в стопке называется *записью*. В базе данных у вас может быть столько записей, сколько вам нужно, однако в какой-то момент стопка может вырасти до таких размеров, что пользоваться ею будет неудобно. Нередко имеет смысл разделить стопку на несколько частей: например, друзья и родственники в одной части, коллеги — в другой.

Пока понятно? А вот теперь самое интересное. Представьте, что ваша стопка учетных карточек стала волшебной. Вы можете разговаривать с ней, приказывая показать вам карточки, отвечающие определенным условиям,

например:

- покажи мне карточку Джона Смита;
- покажи мне все карточки женщин;
- покажи мне все карточки людей, которым больше пятидесяти лет.

Очень удобно, правда? В сущности, именно для этого и служат базы данных — предоставляют вам способ хранения структурированной информации, а также способ ее извлечения в любой удобной форме.

Каждая разновидность данных, которую мы заносим в карточку, называется *полем*. Чем больше полей в базе данных, тем больше потенциальных способов извлечения информации, когда она вам потребуется.

Базы данных — самый распространенный способ хранения информации в интернет-приложениях. Если вам нужно хранить такие данные, как имя пользователя, адрес электронной почты, и другие подобные сведения, для этого прекрасно подойдет база данных. После того как Джон Смит войдет в ваше приложение, вы можете извлечь его имя из записи в базе данных, а затем вывести на экран: «Привет, Джон Смит!»

Другой распространенный способ хранения информации в интернет-приложениях — куки-файл. Это очень маленький текстовый файл, сохраняемый на компьютере пользователя. Куки-файлы удобны для хранения небольшого объема данных непродолжительное время.

В случае нашей программы для адресной книги можно хранить куки-файл, созданный в тот момент, когда Джон Смит входит в систему. Куки-файл Джона содержал бы такую информацию: `username = johnsmith` и `loggedin = true`. Если Смит покинет приложение, а затем снова вернется, программа распознает куки-файл и откроет доступ без повторной регистрации. Можно сделать так, чтобы куки-файлы хранились определенное время, что удобно для этого типа программирования. (Если вы увидите на сайте сообщение «Запомнить этот пароль», то происходит именно запись куки-файла.)

В данном случае мы производим разбиение на основные элементы. Это ни в коем случае не исчерпывающий список уникальных характеристик веб-программирования, однако его вполне достаточно, чтобы получить представление, чему мне нужно научиться. Это переменные, условные операторы, исключения, локальная/рабочая среда, базы данных и куки-файлы.

Видите, насколько полезным оказалось подобное разбиение? Я начал с туманного представления о том, чем я хочу заняться, а теперь у меня есть конкретный список навыков, которые нужно освоить.

Тем не менее я еще не готов приступать к работе. Помните, я говорил о том, что компьютеры не понимают человеческого языка? Мне нужно выбрать язык программирования, чтобы писать команды программы, а для этого

потребуется дополнительные исследования.

Выбор языка для программирования интернет-приложений

В нашем распоряжении есть тысячи самых разных языков программирования, и каждый день появляются новые. У этих языков свой синтаксис, который по большей части определяется тем, для каких задач создавался язык. Каждый язык оптимизирован для решения определенного класса задач.

Прежде чем приступить к изучению программирования, я решил потратить час на просмотр самых популярных сайтов для программистов и выяснить, какие языки рекомендуются разработчикам интернет-приложений. Эти предварительные исследования помогут мне решить, какой язык мне нужно освоить и какие начальные навыки мне потребуются для практических занятий.

Два самых популярных сайта для программистов — это Stack Overflow [7] и Hacker News [8], и поэтому за советом в выборе языка я обратился именно к ним.

Stack Overflow представляет собой диалоговый сайт, место, на котором задают вопросы типа: «Как мне сделать X?» Опытные и знающие программисты обычно отвечают на вопросы — дают конкретные советы, излагают подходы, указывают, как исправить ошибку, — что делает сайт лучшим помощником в решении сложных задач программирования.

Hacker News — это социальная сеть, собрание ссылок на соответствующие обсуждения. Темы на Hacker News меняются ежеминутно, но все они обычно связаны с новыми достижениями в программировании, технологии и бизнесе, что делает этот сайт идеальным местом для поиска хотя бы первичной информации о новостях программирования.

Программисты всего мира ежедневно создают новые языки, библиотеки и методы. Одни сочетания технологий и подходов годятся для конкретной задачи, другие нет. Зачастую убедиться в этом можно только методом проб и ошибок.

С точки зрения программирования термин «лучший» применим лишь по отношению к задаче, которую вы пытаетесь решить, и к вашим приоритетам. В общем случае рекомендуется 1) выбирать инструменты, позволяющие эффективно решить задачу, а 2) при наличии нескольких вариантов выбирать те, которые вам больше нравятся. Вполне справедливо.

Просмотр архивов Stack Overflow и Hacker News привел к перегрузке информацией: ее было слишком много, особенно для того, кто не знаком с терминологией. Мне требовалось избавиться от ненужной информации, чтобы получить конкретный совет.

Тут помогла одна тактическая уловка, о которой многие не знают:

популярные поисковые машины позволяют ограничить поиск конкретным сайтом, чтобы не просеивать всю сеть. Соответствующая команда в Google выглядит следующим образом:

```
"search phrase" site: example.com
```

Замените "search phrase" термином, который нужно найти, а "example.com" — адресом сайта, где вы ищете этот термин. Кавычки означают поиск точного соответствия указанной фразе. Если опустить кавычки, Google покажет страницы, содержащие все слова из фразы, но не обязательно в указанном порядке.

Используя этот прием, я задал поиск нескольких фраз: «программирование интернет-приложений», «обучение программированию», «программирование для начинающих», — и потратил еще около часа на просмотр результатов.

И вот что мне удалось выяснить: опытные создатели интернет-приложений советуют начинать с одного из двух распространенных языков программирования, Ruby или Python. Судя по всему, Ruby и Python относительно легко освоить, они достаточно мощные и служат хорошей основой для важных концепций программирования. Ruby более популярен у программистов, специализирующихся на веб-приложениях, а Python предпочитают ученые, в частности математики, из-за большого количества научных библиотек, позволяющих производить вычисления и строить графики.

У обоих языков, Ruby и Python, есть свои активные сообщества разработчиков, огромное количество бесплатных ресурсов, хорошо написанные книги, а также программы и инструменты, облегчающие применение основных функций. Выбор одного из двух языков, похоже, определяется в основном личными вкусами.

Познакомившись с несколькими примерами программ, написанных на обоих языках, я решил изучать Ruby. Для меня как для новичка программные строки на Ruby выглядели аккуратными, удобочитаемыми и казались относительно понятными. Поскольку основные понятия и приемы, которые я освою при помощи Ruby, будут применимы к другим языкам, которые возможно, заинтересуют меня впоследствии, то почему бы не сделать процесс начального обучения приятным?

Кроме того, я собираюсь использовать программы и инструменты, которые либо поддерживают, либо требуют Ruby. В частности, Jekyll написан на Ruby, и поэтому изучение Ruby поможет мне решить мою насущную проблему. Аналогичные инструменты существуют и для Python, но мне кажется, что пользоваться ими сложнее.

Выбор среды разработки

Помимо советов относительно языков программирования, разработчики интернет-приложений делятся сведениями о *среде разработки*: библиотеках программ, которые облегчают решение задач, встречающихся в каждом приложении.

Библиотеки очень важны потому, что компьютер делает в точности то, что вы ему говорите. Не больше и не меньше.

Это сложно, потому что с точки зрения компьютерной программы существует только код, который вы написали. Как выразился знаменитый физик Карл Саган: «Если вы хотите приготовить яблочный пирог с нуля, вы должны сначала создать Вселенную».

«Вселенная» для вашей программы определяется 1) инструкциями и командами кодовой базы; 2) библиотеками, которые импортирует ваша программа, и 3) системой, в которой выполняется программа. Если код, необходимый для выполнения данной операции, отсутствует в системе, то ваша программа зависнет или выдаст сообщение об ошибке.

Большинство языков программирования включают библиотеки, необходимые большинству программистов, но совсем немного специальных инструментов. Здесь на помощь приходит среда разработки. Вместо того чтобы программировать все с нуля, что займет много времени, вы задействуете среду разработки, которая позволит импортировать и использовать проверенные и надежные библиотеки для решения специальных задач, и сосредоточиваетесь на своем приложении, а не создаете Вселенную.

Среда разработки может иметь разный объем. Некоторые среды включают множество функций и команд в попытке сэкономить программистам время, другие же обходятся минимумом, охватывая лишь несколько основных функций.

В настоящее время у языка Ruby имеется несколько основных сред разработки интернет-приложений. Самые популярные из них — это Ruby on Rails [9] и Sinatra [10].

Ruby on Rails (ее часто называют просто Ruby) была одной из первых больших сред разработки интернет-приложений, предназначенных для Ruby. Созданная в 2004 году Давидом Хейнемейером Ханссоном, Rails использовалась для создания нескольких популярных приложений компании 37signals [11], одним из владельцев которой был Ханссон. В настоящее время тысячи компаний разработали серьезные интернет-приложения для бизнеса на основе Rails.

Программный каркас Rails основан на «генераторах», встроенных программах, которые в ответ на одну команду генерируют большой объем

стандартного кода. Затем стандартный код модифицируется в соответствии с конкретными требованиями программиста. Rails помогает разработчику сэкономить время и создать работоспособное приложение без необходимости писать программу с нуля — при условии, если он знает, что ему нужно.

Среда разработки Sinatra, наоборот, является минимальной. Sinatra не использует генераторы, а предоставляет в распоряжение разработчика несколько простых функций, необходимых практически в каждом интернет-приложении, а затем самоустраняется.

Приложения Sinatra выглядят простыми по сравнению с приложениями Rails. Одна-единственная команда в Rails может сгенерировать больше десяти папок и больше 20 файлов. Приложение Sinatra во многих случаях состоит всего лишь из одного файла. Sinatra не генерирует большой массив кода, который затем может быть удален, а поддерживает простоту проекта, добавляя ровно столько кода, сколько нужно для работы.

Подобно выбору языка программирования, выбор среды разработки определяется в основном личными предпочтениями и характером задачи. Rails больше подходит для крупных проектов, над которыми работают несколько программистов, а Sinatra — для небольших. У этих двух сред разработки много общего, и недавнее исследование на RubySource.com показало, что выбор между ними по большей части определяется субъективными причинами [12].

GitHub — это источник программ с открытым кодом, который используют многие программисты для разработки и поддержки своих проектов. Там легко найти примеры приложений, написанных в Rails и Sinatra, чтобы получить представление о каждой среде разработки, и я потратил еще один час на изучение открытых проектов.

Тут существует нетривиальный риск: чтобы изучить программирование, требуется сделать выбор. После того как вы отдали предпочтение определенному языку программирования и среде разработки, гораздо легче приступить к изучению всего, что необходимо знать для написания программы. Однако, не сделав этого выбора, можно потратить годы на поиски «идеальных» средств программирования.

Лучше выбрать язык и среду программирования, которые кажутся вам привлекательными, потратить какое-то время на знакомство с ними и согласиться на неизбежные компромиссы, чем годами заниматься «исследованиями», не продвинувшись ни на шаг. Целый день просидеть в Stack Overflow и Hacker News — это *не* программирование.

В конечном итоге я решил начать со среды Sinatra. Генераторы Rails могут сэкономить уйму времени, если знать, что ты делаешь, — но я этого *не* знаю.

Интуиция подсказывает мне, что в такой ситуации лучше выбрать Sinatra. В этой среде программирования ясный, простой и понятный синтаксис.

Единственный код, который существует в приложении Sinatra, — это код, созданный разработчиком. Среда разработки хорошо задокументирована, примеры рабочих приложений можно без труда найти на GitHub, а за помощью обратиться к Stack Overflow.

Когда-нибудь я, возможно, поэкспериментирую с Rails. Но теперь лучше начать со среды разработки Sinatra.

Анализ конечного результата

Потратив около пяти часов на предварительные исследования, я сделал все, что требуется для начала работы: разбил интернет-программирование на элементы, выбрал язык, среду разработки и конкретный проект. Пора приступать к делу.

«Разработать приложение в среде Sinatra, которое обслуживает сайт Jekyll» — так выглядит целевой уровень мастерства, однако мне нужно проанализировать это заявление, чтобы спланировать следующий шаг. Чему я должен научиться?

Еще час исследований, и я получаю вот какой результат.

1. Jekyll создает рабочий сайт из локальных статических файлов. Мне нужно сделать шаблон HTML для сайта при помощи специальных форматирующих тегов и экспортировать архив публикаций с сайта PersonalMBA.com, как описывается в инструкции, написанной Полом Стаматиу[13].

2. Приложение, созданное в среде Sinatra, обрабатывает запросы от посетителей сайта, предоставляя им запрошенные файлы. Это приложение я должен написать с нуля.

3. Готовый сайт Jekyll и серверное приложение Sinatra требуется загрузить на сетевой хост.

4. Для решения всех этих задач мне нужно понять, как установить на мой компьютер последнюю версию языка Ruby, а также среду Sinatra и другие необходимые программы.

Последнее требование может служить превосходным примером подготовки необходимых условий для занятий. Если я не смогу понять, как установить Ruby на свой компьютер, то буду не в состоянии перейти к следующим этапам. Так что начинать нужно с этого.

Важно отметить, что интернет-технологии меняются ежедневно. Вполне вероятно, что какие-то последовательности команд, приведенные в данном разделе, уже устареют к тому времени, когда вы будете читать книгу. Не

переживайте: важен метод, а не команды.

Аналогичным образом, у вас возникнет искушение лишь бегло просмотреть приведенные фрагменты программы. Это естественное желание: строки кода выглядят слишком сложными, и вы, скорее всего, не сразу поймете, что они означают.

Я бы советовал вам не поддаваться искушению. Эти имена, команды и символы казались мне такими же чужими, как сейчас кажутся вам. Эта глава посвящена процессу, с помощью которого вы разберетесь в том, что они значат и как их использовать. Если вы попытаетесь прочесть строки программы, то извлечете гораздо больше пользы из этой главы.

Вперед!

Обновление Ruby

Компьютер у меня уже есть, и это хорошее начало: без него писать программу невозможно.

В настоящее время я пользуюсь Apple MacBook Air с операционной системой Mac OS X 10.6. Недолгий поиск в Google позволяет выяснить, что операционная система поставляется с уже установленной программой Ruby версии 1.8.7. И это тоже хорошо: я могу запускать написанные на языке Ruby программы без дополнительных усилий.

Но проблема в том, что Ruby 1.8.7 не самая последняя версия программы. Когда я пытаюсь установить Jekyll, система сообщает, что программа требует версию Ruby не ниже 1.9.1, и это значит, что мне нужно выяснить, как обновить программу. Придется вновь прибегнуть к помощи Google.

Поиск выявил две программы, предназначенные для облегчения установки Ruby: `rbenv` и `ruby-build`. Обе программы поддерживаются Сэмом Стефенсоном, разработчиком Ruby из компании 37signals. Вместе они помогают устанавливать новые версии Ruby и указывать компьютеру, какую версию использовать.

Инструкция на странице документации [14] `rbenv` показывает, как установить программы на компьютер. Вот что представляют собой команды установки:

```
$ cd ~
$ git clone git://github.com/sstephenson/ruby-build.git
$ cd ruby-build
$ sudo./install.sh
$ cd..
$ git clone git://github.com/sstephenson/rbenv.git.rbenv
$ mkdir — p ~/.rbenv/plugins
```

```
$ cd ~/.rbenv/plugins
$ git clone git://github.com/sstephenson/ruby-build.git
$ echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bashprofile
$ echo 'eval "$(rbenv init —)"' >> ~/.bashprofile
$ exec $SHELL
$ rbenv install 1.9.3-p125
$ rbenv rehash
$ rbenv global 1.9.3-p125
```

Выглядит устрашающе, но это всего лишь список команд. Давайте проанализируем его.

Эти команды вводятся в программе Terminal, которая предустановлена на компьютерах Apple. В кинофильмах часто показывают хакеров, которые лихорадочно набирают длинные строчки команд, отображающиеся на экране. Так вот — их компьютеры работают в программе Terminal [15].

Я открываю программу и ввожу первую команду:

```
$ cd ~
```

Эту команду понять нетрудно. Символ \$ программа Terminal выводит на экран, когда готова к приему новой команды. cd — это сокращение от «сменить каталог» (change directory). Не составляет труда выяснить, что символ ~ обозначает «корневой каталог пользователя», то есть папку на моем компьютере, где хранится мой профиль пользователя.

Я набираю команду и нажимаю клавишу «Ввод». Теперь программа Terminal выводит на экран следующую строчку:

```
joshkaufman $
```

Это хорошая новость: я перешел в корневой каталог. Пока все в порядке. Я набираю следующую команду:

```
$ git clone git://github.com/sstephenson/ruby-build.git
```

Компьютер отвечает:

```
git: command not found
```

Похоже, программа git не установлена на моем компьютере. Я должен выяснить, как это сделать.

Что такое git?

Занявшись поиском инструкций о том, как установить git в операционную систему Mac OS X, я наткнулся на Heroku Toolbelt [16]. Heroku [17] — это компания, предоставляющая хостинг для интернет-приложений, следовательно, они заинтересованы в том, чтобы облегчить создание этих самых интернет-приложений.

Heroku Toolbelt представляет собой программу, которая устанавливает несколько распространенных инструментов разработки программного обеспечения, необходимых для создания приложений на Heroku. Среди них есть и git [18].

Я скачал загрузочный пакет, запустил его и получил подтверждение, что все установлено правильно. Теперь попробую еще раз ввести команду:

```
$ git clone git://github.com/sstephenson/ruby-build.git
```

Ответом мне стали следующие строки:

```
Cloning into ruby-build...
```

```
remote: Counting objects: 1004, done.
```

```
remote: Compressing objects: 100 % (453/453), done.
```

```
remote: Total 1004 (delta 490), reused 937 (delta 431)
```

```
Receiving objects: 100 % (1004/1004), 108.77 KiB, done.
```

```
Resolving deltas: 100 % (490/490), done.
```

Успех! «Done» (готово) — это хороший признак. Кроме того, сообщения об ошибке не было. Идем дальше.

Я продолжаю вводить остальные команды. Судя по информации в инструкции по загрузке, я загружаю необходимые файлы и при помощи команды echo добавляю фрагменты текста в файлы конфигурации, а затем перезапускаю программу под названием SHELL, чтобы сохранить изменения. После перезагрузки SHELL программы rbenv и rubybuild установлены на моем компьютере. Ура!

Теперь пора установить последнюю версию Ruby:

```
$ rbenv install 1.9.3-p125
```

Программа автоматически загружает исходный код Ruby и делает сборку, выводя впечатляющее количество информации в окне программы Terminal. (Теперь я чувствую себя как настоящий голливудский программист.) [19]

```
$ rbenv rehash
```

Судя по документации, эта команда помогает компьютеру распознать, что установлена новая версия Ruby.

```
$ rbenv global 1.9.3-p125
```

Эта команда устанавливает, что версия 1.9.3-p125 является версией Ruby по умолчанию на данном компьютере. В руководстве сказано, что запуском

следующей команды я указываю компьютеру необходимость использовать новую версию:

```
$ ruby — v
```

Вот что я получаю в ответ:

```
ruby 1.9.3p125 (2012-02-16 revision 34643) [x8664-darwin11.3.0]
```

Ура! Именно такое сообщение и должен вывести на экран компьютер.

В руководстве сказано, что если я захочу установить новую версию Ruby на этот компьютер, то должен снова запустить выполнение команд `rbenv install`, `rbenv rehash` и `rbenv global`. Достаточно просто.

Несмотря на то что поначалу эти команды вызвали у меня испуг, на самом деле они элементарны. То, что кажется невнятицей, — всего лишь сокращения. После расшифровки этих сокращений понять команду легко.

Полезно помнить, что никто не знает этих тонкостей с рождения. В большинстве случаев нужно потратить несколько минут на чтение документации, а затем попробовать ввести то, что там написано [20].

Установка библиотек Ruby (Gems)

Теперь, когда на компьютере есть последняя версия Ruby, настало время выяснить, как установить нужные мне библиотеки, в том числе Sinatra.

Библиотеки языка Ruby называются *gems*, и, как оказалось, установить их очень легко. Вот команда, которая устанавливает библиотеку Sinatra:

```
$ gem install sinatra
```

Для обновления Sinatra до последней версии нужно ввести следующую команду:

```
$ gem update sinatra
```

Что может быть проще?

Однако, прежде чем устанавливать остальные библиотеки, я хочу убедиться, что программа `gem` у меня тоже обновлена. Поскольку мой компьютер поставлялся со старой версией Ruby, то вполне вероятно, что все связанное с этим языком программное обеспечение тоже нуждается в обновлении.

Потратив немного времени, я нахожу команду для обновления `gem` для Ruby.

```
$ gem update — system
```

Довольно просто.

Запуская команду `gem install`, я замечаю, что она устанавливает и другие библиотеки, такие как `rack`, `rack-protection` и `tilt`. Эти дополнительные библиотеки называются *средствами поддержки зависимостей*. Они нужны для работы Sinatra и поэтому автоматически загружаются по команде `gem install`.

Книги

Теперь, будучи готовым приступить к программированию приложений на языке Ruby, я решил обратиться к двум справочникам по Ruby, которые чаще всего рекомендуют на Stack Overflow: «Программирование на языке Ruby» («The Well-Founded Rubyist») (2009) Дэвида А. Блэка и «Красноречивый Ruby» («Eloquent Ruby») (2011) Расса Олсена. Обе книги представляют собой вводные пособия, предназначенные для того, чтобы познакомить читателя с основными концепциями и приемами Ruby, а также справочники.

Я также купил книгу «Sinatra: к работе готов» («Sinatra: Up and Running») (2011) Алана Харриса и Константина Хассе. Эта книга представляет собой вводный курс в среду программирования Sinatra. Подробная документация на Sinatra имеется в Сети, но книга содержит множество примеров, которые поясняют, как использовать Sinatra для решения стандартных задач.

Просматривая эти книги, я нашел несколько ссылок на сайты, которые могут служить справочниками по синтаксису Ruby.

- The Official Beginner’s Guide to Ruby [21].
- The Ruby Refresher [22].
- Ruby Security Reviewer’s Guide [23].

Для начала этого справочного материала вполне достаточно, и я выделил полтора часа для быстрого просмотра всего, что у меня есть.

Я брал каждую книгу, изучал содержание и указатель, отмечая термины и понятия, которые казались мне важными. Я также выделял концепции, упоминаемые многократно, и обращал внимание на порядок их появления. Я читал заголовки и врезки. Закончив с книгами, я проделал то же самое с сайтами.

Вот что мне удалось выяснить. Помимо переменных, условных операторов, исключений и других основных понятий программирования, язык Ruby строится на двух основных концепциях: *объектах* и *методах*.

Объекты — это существительные мира программирования; с ними мы можем что-то делать. Допустим, я хочу создать в Ruby новую переменную под названием `firstname` (имя) и хочу, чтобы она содержала мое имя. В Ruby эта команда имеет вид:

```
firstname = "Josh"
```

Довольно просто. Помещая имя `Josh` в кавычки, я указываю Ruby, что имя — это *строка*, то есть последовательность букв и цифр. Это делает `firstname` объектом класса «строка». (Класс — это особый тип объектов с определенными характеристиками.)

Строки не единственный класс объектов. Вот пример класса целых чисел:
`million = 1000000`

Если объекты — это существительные языка программирования, то методы являются аналогом глаголов: то, что мы можем делать с объектами.

Предположим, у меня есть два строковых объекта, содержащих мои имя и фамилию:

```
firstname = "Josh"  
lastname = "Kaufman"
```

Я могу использовать символ сложения (+), чтобы *объединить* эти строки, то есть соединить вместе:

```
fullname = firstname + lastname
```

Контрольный вопрос: что содержит переменная `fullname`? Если вы скажете «Josh Kaufman», то ошибетесь.

Не забывайте, что компьютер делает только то, что вы ему говорите. Вы не указывали, что нужно добавить пробел между «Josh» и «Kaufman». Поэтому переменная `fullname` будет содержать «JoshKaufman».

Поэтому если мы хотим исправить эту маленькую ошибку, то нужно изменить команду, добавив в нее пробел:

```
fullname = firstname + " " + lastname
```

«+» — это метод, и его действие зависит от того, к каким объектам он применяется. Если мы применим его к целым числам, а не строкам, то получим операцию сложения, а не объединения.

```
sum = million + million
```

Чему будет равна сумма? «2000000».

Встроенные методы Ruby помогут вам сразу же осуществить много интересных операций. Допустим, я хочу написать свои имя и фамилию задом наперед. Вместо того чтобы записывать это сочетание букв вручную или сочинять маленькую программу для расстановки букв в обратном порядке, я могу использовать метод `reverse`, доступный для каждого строчного объекта:

```
fullname.reverse
```

Вот что получится в результате: `namfuaKhsoJ`

Можно использовать одновременно несколько методов. Если я хочу расположить буквы имени и фамилии в обратном порядке и сделать все буквы строчными, то могу написать команду:

```
fullname.reverse.downcase
```

Результат: `namfuakhsoj`. В точку!

Похоже, научиться программированию на Ruby — это научиться использовать встроенные и создавать новые объекты, классы и методы, а также манипулировать ими. Помимо встроенных, вы можете при необходимости сами создавать, изменять и удалять объекты, классы и методы, что делает язык мощным и гибким средством программирования [24].

Документация на Ruby [25] содержит стандартный список объектов и методов, доступных для пользователя. Этот список впечатляет, но пугаться вы не должны — применять их все совсем не обязательно. Наоборот, большинство из них на первых порах безопаснее не трогать. Это опции, которые вы можете использовать при необходимости.

Документация служит еще одной цели: когда вы пытаетесь дать команду, которую Ruby не понимает, сообщение об ошибке подскажет вам, в каком месте произошел сбой.

Допустим, мы хотим выполнить следующую программу:

```
animal = "Zebra"  
number = 7  
puts animal + number
```

Команда `puts` означает «печать». Мы просто хотим, чтобы программа вывела на экран то, что она понимает под `animal + number`.

Вот что я получил в ответ, когда пытался запустить эту программу:

```
TypeError: can't convert Fixnum into String from program.  
r b:3: in '+'
```

В переводе на понятный человеку язык: нельзя складывать числа со словами, и поэтому компьютер указывает на ошибку. Похоже на деление на ноль: это невозможно, и программа прекращает работу.

Для исправления ошибки в программе нужно либо преобразовать цифру в строку, и тогда метод «+» объединит две переменных, либо изменить программу, чтобы она делала что-то другое.

Вот исправленная программа:

```
animal = "Zebra"  
number = 7.to_s  
puts animal + number
```

Запуская эту программу, мы получаем результат «Zebra7». Встроенный метод `.to_s` преобразует цифру 7 в строку, чтобы Ruby мог использовать объединение переменных.

Можно также сделать нечто совсем другое, например:

```
animal = "Zebra"  
number = 7  
number.times { puts "# {animal}" }
```

Результат на экране компьютера:

```
Zebra  
Zebra  
Zebra  
Zebra  
Zebra  
Zebra
```

Zebra

Тут мы просто использовали условный цикл, встроенный в Ruby: `number.times` означает «выполнять это X раз, где X равняется значению числовой переменной». Изменив значение `animal` или `number`, мы изменим результат. (Да, вы можете модифицировать эту программу так, что она выведет на экран слово «вомбат» миллиард раз.)

Комментарии и отладка

Знакомясь с литературой, я обратил внимание на еще одно фундаментальное свойство языка Ruby — *комментарии*. Каждую строку программы, начинающуюся с символа `#`, Ruby интерпретирует как строку комментария и пропускает при выполнении.

Комментарии к программе делают ее понятной, потому что в них вы простым языком объясняете свои действия. Вот как будет выглядеть программа «Животные» с комментариями:

```
# Назначение переменных
animal = "Wombat"
number = 1000000000
# Цикл вывода
number.times { puts "# {animal}" }
```

Комментарии также очень полезны для устранения ошибок: вы можете обозначить как комментарии несколько строк программы, чтобы устранить проблемы или ошибки. В сочетании с командами `print` или `puts` это позволяет следить за выполнением программы и убедиться, что она работает так, как и планировалось.

Потратив восемь часов на исследования и установку программ, я получил следующий результат: у меня на компьютере последняя версия Ruby, я имею возможность загрузить любую нужную мне библиотеку, и я понимаю, как работают программы, написанные на Ruby.

Теперь давайте разберем более сложные программы.

Тестируем с помощью IRB

Изучая *Stack Overflow*, я обнаружил онлайн-пособие по Ruby под названием «Сложный способ выучить Ruby» («*Learn Ruby the Hard Way*») [26], написанное Робом Соберсом и Зедом Шоу. Это пособие иллюстрирует работу Ruby на примерах простых программ, предлагая вам изменить их для получения тех или иных результатов. Если нужный результат не получается, вы должны

продолжить эксперимент, пока не решите задачу.

Этот метод — кодируем, тестируем, запускаем, отлаживаем — представляет собой превосходный пример *быстрой обратной связи*. Когда вы запустите программу, компьютер уже через несколько миллисекунд даст знать, работает она или нет. Если в коде есть ошибка, вы можете исправить ее и запустить программу снова, в течение минуты проверив несколько вариантов.

Первые главы пособия посвящены настройке Ruby, установке текстового редактора и обучению работы с IRB — программой, которая запускает на вашем компьютере программы, написанные на Ruby.

Вот как это делается. Вы набираете программу в текстовом редакторе и сохраняете ее в файле. (Пусть имя файла будет `program.rb`.) Чтобы запустить ее, нужно набрать команду в окне программы Terminal:

```
$ irb program.rb
```

IRB запустит вашу программу и выдаст результат, а также покажет действия, предпринятые компьютером для получения этого результата, что очень удобно для отладки. Если программа некорректна, IRB выведет на экран подробное сообщение об ошибке.

«Сложный способ выучить Ruby» начинается с назначения переменных, выполнения основных арифметических действий, манипулирования строками и составления стандартных условных циклов — аналогично примерам, которые приводились выше.

Первоначально я собирался прочесть «Красноречивый Ruby» и «Программирование на Ruby», выполнить упражнения из «Сложного способа выучить Ruby», а затем попытаться написать первую «настоящую» программу. Однако к десятому уроку я заметил кое-что важное: мне становится скучно, и я теряю интерес.

Причина проста: я копирую чужие программы и решаю задачи, поставленные другими людьми. Иногда эти программы довольно интересны, однако они не решают моих проблем. Программирование из полезного навыка начинает превращаться в научную дисциплину. Мне нужно заканчивать с исследованием и приступать к практике.

Мне не нужно читать все эти книги, инструкции, пособия и другие источники информации, чтобы приступить к программированию. Я должен писать реальные программы, а если возникнут трудности, обращаться к источникам информации.

Пора замарать руки...

Приложение № 1: статический сайт в среде Sinatra

Я уже придумал свою первую интернет-программу: приложение в среде Sinatra, которое служит основным HTML-сайтом. Вот мой целевой уровень мастерства для этого приложения.

1. Создать рабочее приложение в среде Sinatra, способное предоставить простой сайт пользователю (читателю).
2. Проверить приложение на своем компьютере и убедиться в его работоспособности.
3. Разместить это приложение на сервисе Heroku, сделав «живым», чтобы им могли пользоваться реальные читатели.

Вот и все. Никаких изысков. Простое приложение в среде Sinatra, запущенное на сервере общего пользования.

С чего я должен начинать? Обратимся к списку десяти основных принципов приобретения навыка.

1. Выберите привлекательный проект.
2. Сосредоточьтесь на каком-то одном навыке.
3. Определите целевой уровень мастерства.
4. Разбейте навык на элементы.
5. Приготовьте все необходимое для занятий.
6. Устраните препятствия для занятий.
7. Выделите специальное время для занятий.
8. Создайте быстрые петли обратной связи.
9. Занимайтесь по расписанию, короткими интенсивными интервалами.
10. Уделяйте внимание количеству и скорости.

У меня есть один четко описанный проект. Я разбил навык на составляющие и знаю, как будет выглядеть моя программа, когда я закончу работу. Теперь перейдем к необходимым инструментам: нужно ли мне что-то еще для успешного осуществления проекта?

Как выяснилось, нужно. У меня нет учетной записи на Heroku. Но это легко исправить: я захожу на сайт Heroku.com, щелкаю на кнопке Sign Up, ввожу адрес своей электронной почты и придумываю пароль.

Поскольку я уже установил на компьютере Heroku Toolbelt (программу, использовавшуюся для установки git), то библиотека Heroku загружена в мой компьютер, и с этим проблем не будет.

Судя по инструкции, мне нужно сделать еще одну вещь, чтобы связать мой компьютер с Heroku: сгенерировать «SSH key», специальный файл, который, по всей видимости, служит паролем. Получив этот ключ, я должен сообщить его Heroku, чтобы система могла распознать мой компьютер и

открыть ему доступ.

К счастью, в Heroku есть указания на то, как это сделать [27]. Я ввожу команду для генерации ключа:

```
$ ssh-keygen — t rsa
```

...команду для входа в Heroku:

```
$ heroku login
```

...и команду, которая добавляет ключ к моей учетной записи в Heroku:

```
$ heroku keys: add
```

Отлично: я в системе. Как мне теперь приступить к написанию программы?

Создание простого приложения

Пришло время просмотреть документацию по сервису Heroku. Отлично: там есть два руководства, которые представляются мне полезными:

- «Ruby на Heroku для начинающих» («Getting Started with Ruby on Heroku») [28];
- «Развертывание приложений на основе Rack» («Deploying Rack-Based Apps») [29].

Судя по тому, что написано в этой документации, я должен сделать следующее.

1. Создать файлы с программами на своем компьютере.
2. Добавить их в «репозиторий git». (Понятия не имею, что это...)
3. Использовать команду `git push heroku master`, чтобы отправить готовое приложение на Heroku.

К счастью, в руководстве есть пример, и это приложение, написанное в среде Sinatra! Похоже, все проще, чем я думал...

Я создаю на своем компьютере новую папку. Эта папка называется «корневой», и в ней будут храниться все файлы программы.

Далее я открываю текстовый редактор (я пользуюсь редактором TextMate [30]) и, следуя инструкции, создаю три файла:

```
application.rb
```

```
config.ru
```

```
Gemfile
```

Ядро программы будет находиться в `application.rb`. Приложения, написанные на языке Ruby, всегда оканчиваются расширением. `rb`.

В файле `config.ru` будут храниться параметры конфигурации Rack. Не забывайте, что Sinatra построена над интерфейсом Rack, и поэтому вполне логично, что у него есть отдельный файл конфигурации. Эти файлы оканчиваются расширением `.ru`.

Gemfile — это место, где указываются библиотеки, используемые программой. Ваша программа может иметь только один такой файл, и поэтому он всегда называется одинаково — «Gemfile». Все довольно просто.

Документация по Heroku рекомендует после создания этих файлов написать простейшую программу «Hello, World!», чтобы проверить настройки. Вот что должно быть в файле `application.rb`:

```
require 'sinatra'  
get '/' do  
  "Hello, World!"  
end
```

А так выглядит файл `config.ru`:

```
require './application.rb'  
run Sinatra::Application  
И наконец, Gemfile:  
source 'http://rubygems.org'  
gem 'sinatra'
```

Программа совсем короткая, и понять ее нетрудно.

- Gemfile указывает серверу включить библиотеку Sinatra, которая требуется для работы приложения. Библиотека будет загружена с сайта RubyGems.org.

- Файл `config.ru` указывает основное приложение, затем выполняет программу.

- После запуска программа будет отображать на экране «Hello, World!», когда кто-либо нажмет «/», что указывает на возврат на главную страницу сайта.

Неужели все так просто?

В руководстве говорится, что я должен сделать еще одну вещь: сохранить файлы в репозитории git (иногда его называют просто «репо»). Я не могу сказать, что означает этот термин, но знаю, что git уже установлен, а необходимые команды есть в руководстве:

```
$ git init.  
$ git add — A  
$ git commit — m "Initial Commit"
```

Первая команда создает новый репозиторий git в папке корневого каталога. Команда `add` — A добавляет все файлы в папку репозитория. Команда

commit — m фиксирует файлы в репозитории вместе с сообщением программиста, что именно фиксируется. (Я пока не очень понимаю разницу между «добавлять» и «фиксировать» и беру этот вопрос на заметку, решив разобраться с ним позже.)

После ввода этих команд компьютер отвечает:

```
[master (root-commit) 8ed1099] Initial commit
3 files changed, 9 insertions (+), 0 deletions (-)
create mode 100644 Gemfile
create mode 100644 application.rb
create mode 100644 config.ru
```

Похоже, все получилось! Осталось одно действие: создать новый пустой сервер на Heroku, а затем «поместить» мою программу на этот сервер:

```
$ heroku create
```

Я получаю ответ:

```
Creating shielded-springs-2049... done, stack is stack
is bamboo-ree-1.8.7
http://shielded-springs-2049.herokuapp.com/ | git@heroku.
com: shielded-springs-2049.git
Git remote heroku added
```

Получилось! «Stack» указывает программное обеспечение, выполняемое на сервере, и в сообщении прописывается общедоступный URL сервера.

И последняя команда:

```
$ git push heroku master
```

Если все пройдет хорошо, то я официально помещу свое первое приложение для выполнения на Heroku.

Вот что я получаю в ответ:

```
Heroku receiving push
Ruby/Sinatra app detected
Gemfile detected, running Bundler version 1.0.7
Unresolved dependencies detected; Installing...
Using — without development: test
! Gemfile.lock will soon be required
! Check Gemfile.lock into git with `git add Gemfile.lock`
! See http://devcenter.heroku.com/articles/bundler
Fetching source index for http://rubygems.org/
Installing rack (1.4.1)
Installing rack-protection (1.2.0)
Installing tilt (1.3.3)
Installing sinatra (1.3.3)
Using bundler (1.0.7)
Your bundle is complete! It was installed into ./bundle/gems/
```

Compiled slug size: 500K

Launching... done, v4

<http://shielded-springs-2049.herokuapp.com> deployed to Heroku

И теперь. момент истины... Я открываю веб-браузер, перехожу на <http://shielded-springs-2049.herokuapp.com> и вижу вот что:

«Hello, World!»

ПОБЕДА!

Внимание! Внимание!

Программа заработала, но я также получил предупреждение. Что такое Gemfile.lock?

В документации Heroku's Ruby указано, что для установки библиотек на Heroku система использует библиотеку под названием bundler [31]. Это библиотека, и поэтому я могу установить ее при помощи следующей команды:

```
$ gem install bundler
```

Библиотека bundler необходима потому, что Heroku по умолчанию не устанавливает библиотеки из вашего приложения. Из соображений безопасности Heroku не присваивает мне такой уровень доступа к серверу, какой есть у меня на моем компьютере, и я не могу выполнить команду `gem install sinatra` прямо в моей учетной записи.

Вместо того чтобы предоставить мне (и другим пользователям) небезопасный уровень доступа к системе, Heroku использует bundler для установки библиотек, указанных в Gemfile. Отметив, какие библиотеки вы хотите установить в своем приложении, вы набираете команду:

```
$ bundle install
```

Эта команда создает в вашей программе новый файл под названием Gemfile.lock. Когда вы загружаете свои файлы в Heroku, система просматривает Gemfile и Gemfile.lock, проверяет их идентичность, после чего загружает библиотеки, а затем устанавливает их для вас.

Если вы посмотрите на ответ системы, когда я размещал программу на Heroku, то увидите, что система устанавливала bundler автоматически, как средство поддержки зависимостей. Вместо того чтобы предусмотреть сообщение об ошибке, инженеры Heroku добавили в программу исключение — вместо аварийного завершения автоматически загрузить отсутствующую программу и отправить предупреждение.

На этот раз система сработала, но в будущем я должен добавить файл Gemfile.lock в репозиторий git, прежде чем размещать приложение. Полезная информация!

На сцену выходит Sinatra

Теперь, когда мое простейшее приложение создано и запущено, я могу наконец начать разбираться, как работает Sinatra. Документация среды программирования Sinatra [32] очень подробна и включает множество примеров, и поэтому я решил начать именно с нее.

Основа приложений Sinatra называется *маршрут*. Лучше всего объяснить это понятие на примере.

Наше простое приложение Sinatra имеет один маршрут, который содержит «корень» сайта. Пользователи интернета обычно указывают корень сайта как начальную страницу сайта.

Если вы заходите на `google.com` или на `yahoo.com`, то браузер посылает запрос на сервер Google или Yahoo. Этот запрос называется GET-запросом, и он просит сервер показать вам, что находится в корневом каталоге сайта. *Протокол*, или формат, который использует компьютер для отправки запроса, называется HTTP, что расшифровывается как «протокол передачи гипертекста». Именно с этой аббревиатуры начинаются адреса в интернете — «`http://`».

GET — это самый распространенный из запросов HTTP, но не единственный. Существуют три дополнительных «глагола» HTTP:

- POST — отправить ресурс на сервер;
- PUT — обновить ресурс на сервере;
- DELETE — удалить ресурс с сервера.

Если вы оставляете на сайте комментарии, то ваше остроумное замечание отправляется на сервер командой POST. Если вы сделали ошибку и редактируете комментарий, то обновление выполняется с помощью команды PUT. А если вы решили, что комментарий неудачный и его нужно удалить, браузер посылает команду DELETE.

Маршруты GET, POST, PUT и DELETE — это основа работы приложений Sinatra. Каждый создаваемый вами маршрут является условным: «Делать X, если команда GET/POST/PUT/DELETE получена на маршруте Y».

Маршруты Sinatra также содержат переменные, которые называются *параметрами*. Приложения Sinatra обычно используют параметры в качестве входных данных для процесса, содержащегося в каждом маршруте.

Давайте модифицируем наше простое приложение Sinatra «Hello, World!», чтобы оно приветствовало пользователя по имени. Вот маршрут, который это делает:

```
get '/hello/: name' do
  "Hello, # {params [: name]}!"
end
```

end

Вы можете увидеть результат на <http://first20hours.com/hello/name>. Попробуйте заменить «name» своим именем. Работает!

Это приложение позволяет серверу посмотреть, что содержится в параметре маршрута «name», и использовать это значение. Команда маршрута представляет собой простую инструкцию — показать параметр «name» пользователю.

Sinatra позволяет вам дать название параметру (например, name), но также имеет «универсальный» параметр (его также называют «решеткой» (splat)), который может содержать все что угодно. Вот как можно использовать его в нашем модифицированном приложении «Hello»:

```
get '/hello/*' do
  "Hello, # {params [: splat]}!"
end
```

Это очень удобно. С помощью обычных и универсальных параметров можно прописывать самые замысловатые маршруты. Создаваемые вами маршруты определяют работу приложения Sinatra.

Этих сведений достаточно, чтобы понять, как написать программу, которая удовлетворяет моему целевому уровню мастерства. Поскольку Jekyll создает файлы, которые приложение предоставляет по запросам пользователей GET, мне остается лишь написать несколько маршрутов, которые принимают эти запросы, находят в системе нужные файлы и предоставляют их читателю.

Как указывается в документации на Jekyll, программа помещает законченные веб-страницы в папку под названием «site» в корневом каталоге. Маршрут к этой странице автоматически генерируется Jekyll. Если мы хотим, чтобы наша страница About была доступна на <http://example.com/about>, мы ставим маршрут /about в файлах Jekyll, и программа создаст файл в `_site/about/index.html` в корневом каталоге нашего сайта.

Это означает, что я должен создать новый маршрут в Sinatra, чтобы прочесть файл в ответ на запрос пользователя GET. Вот как он будет выглядеть:

```
# Index handler
get '/' do
  ## File.read ("_site/index.html")
end
# Post handler
get '*/?' do
  ## File.read ("_site/# {params [: splat]} /index.html")
end
```

Команда `File.read (“”)` встроена в Ruby. `File` — это объект, `read` — метод. Используется она очень просто: содержимое (“”) указывает на расположение файла, который должна прочитать программа, относительно корневого каталога

приложения. Легко.

А что если такого файла нет? В этом случае необходимо исключение, и в Sinatra встроены два основных маршрута ошибок: `not_found` и `error`. Давайте сделаем так, чтобы оба маршрута вызывали одну и ту же страницу с сообщением об ошибке:

```
not_found do
  ## File.read ("_site/error/index.html")
end
error do
  File.read ("_site/error/index.html")
end
```

Все остальное остается таким же. Я не собираюсь вносить изменения в `config.ru` или `Gemfile`. Я просто добавляю новые маршруты в программный репозиторий `Git`, внося изменения, а затем размещаю исправленные программы на `Heroku`. Готово.

Хотите посмотреть, как работает модифицированная программа? Зайдите на сайт `PersonalMBA.com`: в настоящее время сайт работает на `Jekyll` вместо `WordPress`, используя именно эту программу. При помощи программы нагрузочного тестирования, которая называется `seige`, мой сайт теперь способен без труда обслуживать две тысячи посетителей одновременно. Большинство запросов страниц обрабатываются за 18–20 миллисекунд, так что мой сайт полностью защищен от падения при большом трафике.

Мое первое рабочее интернет-приложение завершено. Мне потребовалось около часа, чтобы разобраться, что я должен делать, и еще час, чтобы понять, как перенести информацию и дизайн моего сайта с `WordPress` на `Jekyll`.

Общее время составило около десяти часов, включая исследования и обзор концепций программирования. Не так плохо!

Приложение № 2: Codex, персональная база данных

Мое первое приложение работает, и работает хорошо. Оно простое, и в данном случае это его главное преимущество. Чем меньше движущихся деталей, тем меньше вероятность поломки — это справедливо и для программы.

Давайте займемся чем-нибудь чуть более сложным.

Помните наш разговор о базе данных? Главные веб-страницы не могут обновляться, а значит, не могут хранить информацию. Мое первое приложение работает только потому, что файлы статичны: они не изменяются. Любые изменения в файлы вносятся посредством `Jekyll`, отдельной программы. Приложение получилось быстрым и стабильным потому, что не использует базу данных.

А как быть с программами, использующими базы данных? Базы данных

обычно являются важной частью большинства приложений, и поэтому я должен понять, как они устроены. А для этого нужно начать работу над проектом, основанным на базе данных.

Одно из приложений, которым я пользуюсь, называется Backpack [33] и разработано компанией 37signals. Основное преимущество Backpack — возможность создавать страницы, содержащие все что угодно: текст, списки, изображения, файлы и так далее. Сохранив информацию о странице в Backpack, впоследствии вы можете получить к ней доступ с любого компьютера, поскольку вся информация хранится в базе данных приложения.

Я спрашиваю себя: могу ли я сам создать нечто подобное? Попробовать стоит... но с чего начать?

Изучая Jekyll, я прочел статью программиста Тома Престона-Вернера, создателя Jekyll, под названием «Начни разработку с Readme» («Readme Driven Development») [34]. Идея статьи заключалась в том, что вопреки волнам увлечения проектным менеджментом, которые каждые несколько лет захлестывают индустрию программного обеспечения, лучший способ создать приложение — начать с написания ознакомительного документа Readme.

Файл Readme программисты включают в корневой каталог приложения вместе с программным кодом. В нем содержится информация по установке, настройке и использованию программы.

Файлы Readme очень важны, поскольку многие программы требуют пояснений. Без этого документа зачастую бывает трудно понять, как пользоваться программой. Самостоятельный просмотр кода не так эффективен, как чтение подробного объяснения, написанного создателем программы.

Том утверждает, что сначала нужно написать файл Readme, а уж затем браться за саму программу. Большинство программистов поступают с точностью до наоборот: сначала пишут программу, а затем (возможно) ознакомительный файл. И упускают благоприятную возможность: разработка документации помогает понять, как именно будет работать программа. Файл Readme может быть не только полезным документом, но и средством разработки.

Такой подход мне нравится. Один из приемов, которые я освоил в процессе работы над «Сам себе MBA», заключался в написании рекламных материалов до, а не после создания коммерческого предложения. Выясняя, что требуется потенциальным покупателям, и включая это в рекламный материал, вы получаете более полное представление о том, каким должен быть продукт, чтобы привлечь внимание клиентов. Разработку самого предложения определяет маркетинговое исследование.

Я взял блокнот и составил список функций, которые я хотел бы иметь в своем приложении, а также характеристик программы.

- Программа представляет собой простое приложение, предназначенное для создания заметок.
- Приложение предназначено для одного пользователя.
- Приложение использует среду Sinatra и базу данных для создания, сохранения, обновления и удаления записей на странице.
- Приложение позволяет пользователю создавать страницы со сложным форматированием, таким как жирный шрифт, курсив, подчеркивание и тому подобное.
- Приложение требует пароль доступа и обеспечивает сохранность базы данных, насколько это возможно.
- У приложения привлекательный внешний вид.
- Приложение может без труда размещаться на Heroku или любом другом ресурсе.

Я собираюсь назвать это приложение «Codex» — в старину так называли книги, — поскольку приложение будет применяться в основном для хранения справочной информации, списков и тому подобное.

В интернет-программировании подобные приложения называются «CRUD» — аббревиатура от английского «создание чтение обновление удаление». Стоит отметить, что эти функции совпадают с командами GET, POST, PUT, DELETE, и поэтому написать такое приложение можно и с использованием маршрутов Sinatra. Разница, причем существенная, заключается в использовании базы данных.

Какие возможности работы с базами данных предоставляет Heroku? Не знаю. Придется вновь заглянуть в документацию.

По умолчанию Heroku использует базу данных под названием Postgres [35]. Каждому новому приложению автоматически назначается маленькая база данных разработки. Мне это подходит, но как ею пользоваться и какими средствами протестировать программу на моем компьютере?

За ответами на эти вопросы я решил обратиться к Stack Overflow. Общее мнение: решение подобных задач значительно облегчает применение базы данных под названием DataMapper [36].

DataMapper относится к классу программ, получивших название «объектно-реляционного отображения», или сокращенно ORM [37]. Метод ORM решает насущную проблему программистов: базы данных зачастую используют собственный язык, отличающийся от языка, на котором пишется приложение. В большинстве баз данных используется язык SQL [38], однако существуют и сотни других языков.

Предположим, что мы пишем программу для сайта Amazon.com и хотим вывести на экран список книг Дж. К. Роулинг, автора серии о Гарри Поттере. Команда на языке SQL будет выглядеть так:

```
SELECT * FROM Book WHERE author = "J.K. Rowling" ORDER BY title;
```

Данная команда извлекает из базы данных все записи, в поле «author» которых содержится «J.K. Rowling», и располагает их в алфавитном порядке по названию книги.

К сожалению, заставить SQL или любой другой язык запросов базы данных корректно работать с такими языками, как Ruby, — сложная задача. Непросто писать программу на одном языке, не говоря уже о нескольких языках одновременно.

Здесь на помощь приходит ORM: она позволяют программисту писать программу на одном языке, который ORM затем переводит на язык базы данных. Так гораздо проще.

Таким образом, DataMapper представляет собой библиотеку, облегчающую связь с базами данных, использующих Ruby. По умолчанию DataMapper содержит большое количество удобных функций для создания, чтения, обновления и удаления записей в базе данных. Поскольку программа DataMapper существует уже довольно давно и тщательно протестирована, в большинстве случаев надежнее пользоваться ею, а не пытаться писать собственную программу для базы данных.

DataMapper доступна в виде стандартной библиотеки, которая устанавливается следующим образом:

```
$ gem install data_mapper
```

Поскольку объем библиотеки DataMapper очень велик, ее можно устанавливать по частям. Этот принцип называется «модульностью», и он может служить признаком профессионального программирования. Вот как выглядит команда загрузки всех отдельных библиотек:

```
$ gem install dm-core dm-aggregates dm-constraints dm-migrations dm-transactions dm-serializer dm-timestamps dm-validations dm-types
```

Вместо установки всей библиотеки вы можете загрузить только те модули, которые будет использовать ваша программа, что гораздо эффективнее.

Использование DataMapper

Теперь, когда библиотека DataMapper установлена, я должен научиться с ее помощью 1) связываться с базой данных и 2) создавать базу данных для хранения и извлечения нужной мне информации.

В документации на Heroku указано, что следующая команда позволит приложению Sinatra связываться с базой данных:

```
DataMapper.setup (: default, ENV ['DATABASE_URL'] || "sqlite3://# {Dir.pwd} /database.db")
```

В данном случае || означает «или». ENV ['DATABASE_URL'] — это

переменная, которую Heroku использует для вашей базы данных. Если эта база данных недоступна, то будет применен второй вариант, база данных под названием Sqlite [39].

Sqlite установлена на компьютерах Mac по умолчанию и уже готова к работе. DataMapper может связываться и с Postgres, и со Sqlite, если я установлю обе библиотеки:

```
$ gem install dm-sqlite-adapter dm-postgres-adapter
```

Это означает, что мое приложение будет использовать Postgres при запуске на сервере Heroku и Sqlite при запуске на моем компьютере. В обоих случаях моя программа остается одной и той же, несмотря на то что базы данных используют разные языки. Это действительно здорово.

Кстати, по поводу запуска такого приложения на моем компьютере... Как это сделать?

Pow!

Информацию о том, как запустить приложения такого типа на моем компьютере, я искал на Stack Overflow и Hacker News. К счастью, существует несколько вариантов. Похоже, я могу установить библиотеки (например, Foreman или Shotgun), которые выполняют приложение, когда я ввожу команду в Terminal, или программу, поддерживающую приложение все время.

Программа, использующая этот подход, называется Pow [40] — «Rack-сервер нулевой конфигурации для Mac OS X». Сайт обещает установить локальный хостинг для разработки приложений на моем компьютере меньше чем за минуту. Звучит привлекательно!

Установка Pow занимает меньше десяти секунд: для загрузки и установки приложения требуется всего одна команда в Terminal. После этого вы вводите команду, которая связывает вашу программу с Pow, и Pow позволит запускать ее на вашем компьютере.

Библиотека Ruby с названием Powder [41] еще больше облегчает этот процесс:

```
$ gem install powder
```

После установки библиотеки вы устанавливаете Pow:

```
$ powder install
```

Затем переходите в корневой каталог своего приложения и вводите следующую команду:

```
$ powder link
```

Вот и все. Мой корневой каталог называется «codex», поэтому теперь приложение выполняется на моем компьютере на <http://codex.dev>, и у меня появляется возможность протестировать свою работу.

Если я вношу изменения, следующая команда заново запускает

программу:

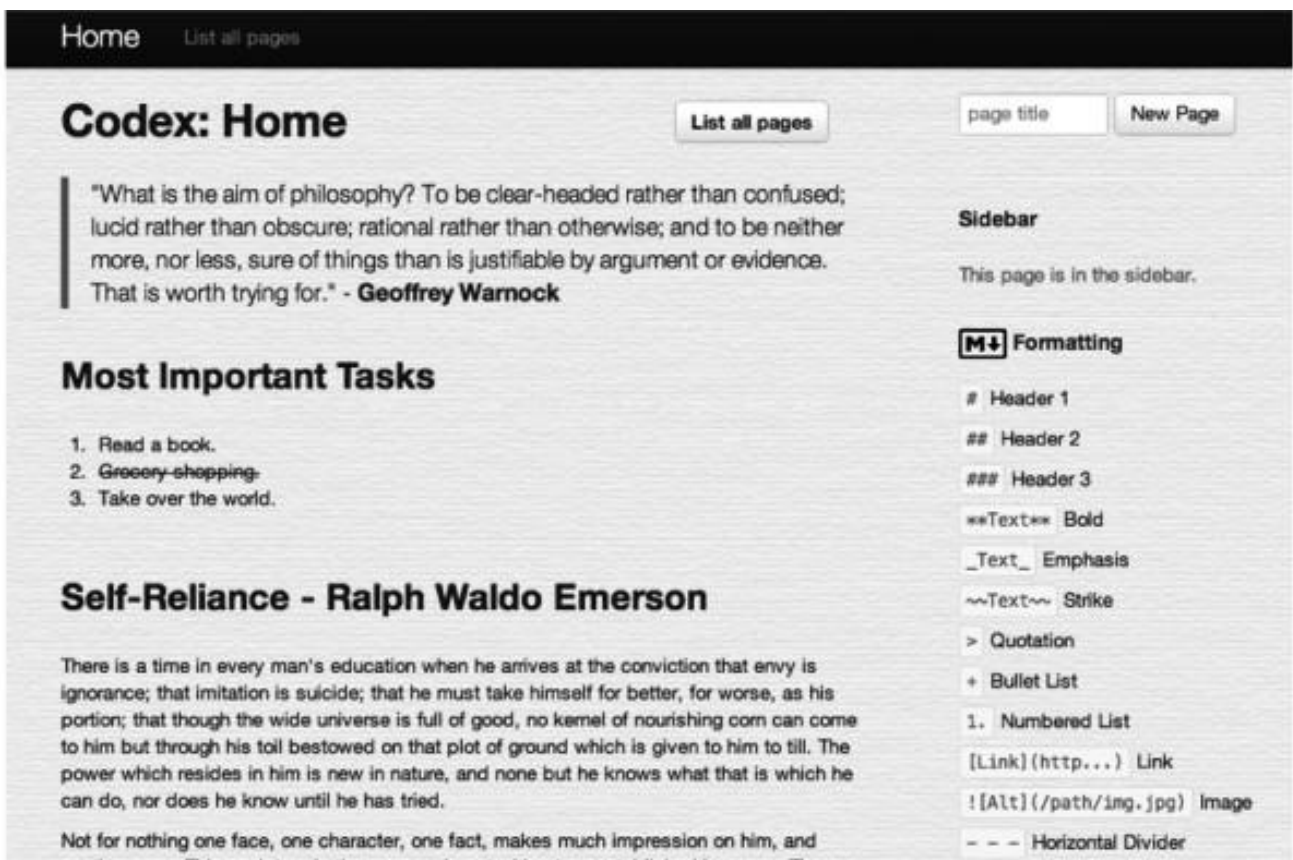
```
$ powder restart
```

Все просто. Теперь я готов писать программу. Каждый вечер я буду выделять полтора часа на составление программы — пока работа не будет закончена.

Кодируем, тестируем, исправляем

Теперь я буду описывать, что я делаю, а не как. Полный текст программы вы, если захотите, сможете найти на <https://github.com/first20hours/codex>.

Вот как должно выглядеть приложение, когда работа будет завершена.



Вы, наверное, заметили, что дизайн страницы состоит из трех частей: верхней навигационной панели, области основного содержания и бокового поля. Я сконструировал эту структуру с помощью инструмента Bootstrap [42], созданного разработчиками Twitter Марком Отто и Джейкобом Торнтоном.

Веб-страницу не нужно разрабатывать с нуля, поскольку Bootstrap представляет собой библиотеку шаблонов HTML и CSS, находящихся в открытом доступе. Использование Bootstrap экономит вам уйму времени: вы сможете собрать прототип своего приложения за несколько минут вместо

нескольких дней.

Основная единица этого приложения называется «Page», и она отображает запись, хранящуюся в базе данных. Специальная кнопка позволяет просматривать все записи базы данных. Внизу расположены еще две кнопки. Первая дает возможность редактировать текущую страницу, вторая — удалить.

В боковом поле реализованы три основные функции. Вверху расположена форма, позволяющая создать новую страницу, для чего нужно ввести ее название. Здесь также есть список страниц, добавленных пользователем, и этот список может служить указателем. И, наконец, справка по форматированию, помогающая пользователю вспомнить, как применять стандартные функции форматирования.

Верхняя навигационная панель очень проста. Она содержит ссылку на главную страницу, а также дополнительную ссылку на экран «Показать все страницы». Позже я могу добавить еще функции, если потребуется, но пока этого достаточно.

Каждое интернет-приложение имеет главную страницу, и я должен решить, что хочу видеть на своей главной странице. В данном случае мне просто нужно отобразить запись Home базы данных.

Что же содержится в Page? Поскольку каждая Page представляет собой запись базы данных с полями, в которых хранится информация, я должен указать `DataMapper`, какие поля требуется создать. Вот как это выглядит:

```
class Page
  include DataMapper::Resource
  property: id, Serial
  property: title, String
  property: content, Text
  property: lastupdated, DateTime
end
DataMapper.finalize
```

Данный фрагмент программы использует `DataMapper` для создания нового объекта под названием `Page`. Теперь Ruby может использовать `Page` подобно любому другому объекту, а я могу создавать и применять методы, которые формируют, изменяют и удаляют объекты `Page`. Изменения, внесенные в `Page`, сохраняются в базе данных при помощи `DataMapper`.

Команда `DataMapper.finalize` указывает приложению, чтобы оно создало эти поля, если они еще не существуют в действующей базе данных.

Теперь, когда база данных есть, пора выяснить, какие маршруты должна будет обслуживать среда разработки Sinatra. Вот какой у меня получился список:

```
# Show home page
get '/'
```

```
# Creates new note from "new page" form
p o s t '/'
# Displays requested note
get '/: url/'
# Edits requested note
get '/: url/edit'
# Saves user edits to note
p o s t '/: u r l / e d i t'
# Deletes specified note
delete '/: url/'
# List all pages in database
get '/all/'
# Error handling
not_found
error
```

Это очень полезный список. Мое приложение будет строиться вокруг команд, которые я создам для каждого маршрута.

Слаги, везде слаги!

Помните, я сравнивал базу данных с магической стопкой учетных карточек, которую вы можете просматривать так, как вам удобно? Нам нужен способ поиска конкретных записей, и именно потому в этих маршрутах встречается `url`. Содержание параметра `url` указывает базе данных, какую запись следует извлекать.

В качестве параметра можно использовать заголовок страницы, но тут есть одна проблема: браузеры не любят в интернет-адресах такие вещи, как пробелы, заглавные буквы и специальные символы (например, `$` и `%`). В заголовках такие символы встречаются достаточно часто, поэтому от них нужно избавиться.

Строка, однозначно определяющая страницу сайта, называется *слагом* [43]. Слаг моей страницы будет основан на заголовке и подчиняться правилам, приемлемым для интернет-адресов.

1. Все символы строчные.
2. Никаких специальных символов — только буквы и цифры.
3. Никаких пробелов — все пробелы должны быть заменены на подчеркивание.

Для этого нужно создать метод, который берет заголовок страницы и форматирует его в слаг. Сложные замены выполняются при помощи метода

программирования, получившего название *регулярных выражений*, который преобразует или находит текст в строках в соответствии с установленными правилами [44].

Регулярные выражения могут быть очень сложными, но применяются они широко, и я смог найти подходящий шаблон. Вот как выглядит мой метод:

```
# Converts page name into post slug
def slugify (content)
  content.downcase.gsub (/ /, '-').gsub (/ [^a-z0-9-] /, "").squeeze ('-')
end
```

Теперь я могу использовать метод `slugify` для преобразования строк типа «Заголовок Страницы» в «заголовок_страницы», которые можно использовать в составе интернет-адреса.

Кроме того, если мы храним слаг вместе с заголовком страницы, то можем использовать его для извлечения записи страницы при помощи параметра `url`.

Я добавил это поле к классу `DataMapper`:

```
property: slug, String
```

Теперь при создании страницы мы можем превратить ее заголовок в слаг, сохранить его в базе данных и использовать для демонстрации страницы, когда потребуется ее извлечь. Именно так приложение узнаёт, какую страницу показывать.

Создание страниц

Я начал разрабатывать список необходимых маршрутов. «Главный» маршрут прост: я перенаправляю его к слагу `/home/`, поскольку хочу, чтобы главная страница отображала содержимое соответствующей записи.

Маршрут «создать страницу» связан с небольшим полем ввода в верхней части бокового поля. Пользователь вводит заголовок страницы в поле ввода и щелкает по кнопке. Система считывает заголовок страницы, преобразует его в слаг, затем сохраняет в базе данных заголовок, слаг и время создания. После этого система посылает запрос `GET`, содержащий слаг, и отображает новую страницу.

В маршруте «создать страницу» есть маленькая, но очень важная деталь: что если страница уже существует? Я не хочу переписывать ее заново, если она содержит данные. Случайная потеря данных недопустима.

К счастью, `DataMapper` разрешает эту проблему при помощи встроенного метода `first_or_create`. Если страница уже существует, `DataMapper` не переписывает ее, а `Sinatra` перенаправляет браузер к существующей странице. Проблема решена.

Маршрут «отобразить страницу» считывает слаг в URL, извлекает запись из базы данных, а затем отображает информацию в области основного содержания. Позже я добавлю красивое форматирование, но теперь мне просто нужно, чтобы программа работала.

Редактирование страниц

Редактирование страниц включает два отдельных маршрута. Первый маршрут запросом GET вызывает страницу, которую хочет отредактировать пользователь, а затем отображает содержимое соответствующей записи базы данных в форме, удобной для редактирования.

Для отображения страниц приложение использует шаблон ERB, который обычно состоит из команд HTML + Ruby. ERB позволяет программисту писать HTML, включающий изменяемые элементы. Поскольку ERB обрабатывает страницу до того, как она появляется перед пользователем, текст на странице можно менять при каждой загрузке, основываясь на командах Ruby шаблона.

Кнопка «Сохранить» на экране редактирования посылает запрос POST приложению, которое обновляет запись Page.

Удаление страниц

Удаление страниц требует осторожности: нельзя забывать, что случайная потеря данных неприемлема. Собираясь удалить страницу, вы должны быть абсолютно уверены, что пользователь хочет именно этого.

Неправильно было бы напрямую связать кнопку «Удалить» с маршрутом DELETE приложения. При таком подходе страница удалится даже при случайном щелчке по кнопке.

Предпочтительнее разбить процесс на два этапа. Щелчок по кнопке «Удалить» на экране отсылает пользователя к окну подтверждения, где отображается заголовок страницы, которую пользователь намерен удалить. Если все правильно, пользователь щелкает по кнопке подтверждения, посылая запрос DELETE. Если кнопка «Удалить» нажата случайно, можно щелкнуть по кнопке «Отмена» или кнопке браузера «Назад».

Список всех страниц

Кнопка «Показать список страниц» направляет пользователя к слаггу /all/, который отличается от слаггов обычных страниц.

Вместо того чтобы извлечь одну запись, DataMapper извлекает все записи Page базы данных. Шаблон ERB для Page содержит условный цикл, который создает список всех извлеченных страниц. Каждый элемент списка содержит заголовок страницы, отображаемый в виде ссылки на соответствующую странице слаг. Щелчок по ссылке направляет пользователя на конкретную

страницу.

Первый запуск приложения

Итак, у меня есть основные функции, однако есть и проблема: во время тестирования приложения попытка перейти к главной странице вызывает сообщение об ошибке. Программа пытается найти запись Home в базе данных, но ее не существует, поскольку я запустил приложение в первый раз!

Выход — создать «однократный процесс управления» при помощи программы под названием Rake. Программы Rake хранятся в папке Rakefile, которая находится в корневом каталоге приложения.

Файлы Rake работают как обычные приложения Ruby, но с одним исключением: они существуют вне вашей основной программы, и вы должны запускать их вручную.

Это делает Rake очень удобным средством для таких действий, как добавление начальной информации в базу данных перед запуском реальной программы. Я копирую необходимые фрагменты application.rb в Rakefile, затем пишу команду, которая создает новую запись для главной страницы в базе данных. Остается лишь один раз запустить следующую команду:

```
$ rake setup
```

Rake создает запись «Главная страница», и мое приложение перестает выдавать ошибку при запуске. Когда я размещу это приложение на Heroku, то запущу команду Rake удаленно, чтобы настроить базу данных перед первой попыткой использовать приложение.

Теперь все основные функции на месте. Пора добавить что-нибудь интересное.

Добавление поддержки бокового поля

Я только что сообразил, что не включил в боковое поле функцию добавления страниц, как планировал изначально. Это логическая функция, поскольку может иметь только два значения: страницу предполагается отобразить в боковом поле или нет.

```
Я добавил новый класс в DataMapper:  
property: sidebar, Boolean, default => false
```

Я также добавил кнопку-флажок к экрану «Редактировать» рядом с вопросом «Включить в боковое поле?» и связал эту кнопку с соответствующим полем в базе данных. Затем написал простой цикл, чтобы найти записи в базе данных, когда sidebar = true, и вывести их в виде списка, аналогичного странице

«Список всех страниц».

Я снова запускаю приложение, редактирую запись, и приложение вылетает. Проклятье!

Я снова и снова пытаюсь найти ошибку, но безуспешно. Перерыв документацию DataMapper и обратившись к Stack Overflow, я выясняю, что логические переменные плохо сочетаются с формами HTML. Нужен другой подход:

```
property: sidebar, Enum [: yes, no], default =>: no
```

Это другой способ сделать примерно то же самое. Enum, что означает «перечислять», создает список опций, и форма устанавливает, какую опцию сохранить в базе данных.

Добавление поддержки Markdown

Теперь я хочу, чтобы мои страницы могли иметь сложное форматирование, например жирный шрифт, курсив, заголовки.

В качестве форматизирующего синтаксиса я выбрал Markdown, популярный и очень полезный язык разметки, разработанный Джоном Грубером [45]. Я уже знаком с работой Markdown, поскольку его использовали некоторые приложения, которые установлены на моем компьютере. Теперь мне нужно выяснить, как заставить мою программу понимать этот язык.

Довольно быстро мне удастся найти несколько общедоступных библиотек Markdown. Я выбираю библиотеку rdiscount и включаю в application.rb:

```
require 'rdiscount'
```

Rdiscount преобразует текст из формата Markdown в HTML, а браузер пользователя затем отображает должным образом отформатированный текст. Файлы Markdown представляют обычные текстовые файлы, написанные по определенным правилам.

Это значит, что мне не нужно преобразовывать информацию моей страницы в Markdown, прежде чем добавлять в базу данных. Это ведь обычный текст. Если я хочу отобразить отформатированный текст, мне нужно лишь вызвать rdiscount.

Команда, которую я добавил в шаблоны ERB, отвечающие за отображение страниц, выглядит следующим образом:

```
<% markdown (@page.content)%>
```

Метод берет содержимое поля содержания страницы, преобразует в HTML, а затем отображает конечный результат у пользователя. Все просто.

Меры безопасности

А как насчет регистрации пользователей? Если я размещу приложение на Heroku, не запрашивая имени пользователя и пароля, то все смогут увидеть, что я помещаю в базу данных.

Как выяснилось, современные браузеры поддерживают безопасный протокол под названием «Базовая аутентификация HTTP» [46], который представляет собой простой способ потребовать у пользователя имя и пароль, прежде чем продолжить работу. Если пользователь не может предоставить полномочия доступа, его переправляют на страницу с сообщением об ошибке.

Вот фрагмент программы, обеспечивающий базовую идентификацию в Sinatra:

```
use Rack::Auth::Basic, "Restricted Area" do |username, password|
  [username, password] == [ENV ['ADMIN_USER'], ENV ['ADMIN_PASS']]
end
```

В данном случае я храню имя пользователя и пароль в Heroku как *переменные среды*, которые я могу задать командой Terminal. Это позволяет мне использовать один и тот же код для разных приложений, а также показать вам этот код, не сообщая своего пароля!

Это также превосходная иллюстрация того, почему важно знать, что Sinatra построена над интерфейсом Rack. Дело в том, что существует много общедоступных библиотек типа Rack::Auth::Basic, и любую из них я могу использовать с Sinatra. Чем реже мне приходится изобретать колесо, тем лучше.

Кроме того, я хочу добавить еще одну меру безопасности — шифрование. Протокол SSL — его используют банки для обеспечения безопасности онлайн-банковских операций — добавится к моему приложению с помощью библиотеки rack-ssl-enforcer:

```
require 'rack-ssl-enforcer'
use Rack::SslEnforcer
```

Библиотека заставляет браузер использовать безопасное SSL-соединение для доступа к сайту. Heroku позволяет приложениям, размещенным на его доменах, использовать SSL по умолчанию, поэтому никакие дополнительные настройки не нужны [47].

Флэш-сообщения

Есть еще одна функция которую мне хотелось бы добавить. Я видел сайты, где в ответ на ваши действия появляются маленькие сообщения, например «Ваша страница создана/отредактирована/удалена». Как это сделать?

После недолгих поисков я обнаружил библиотеку под названием `sinatra-flash`, которая обслуживает эту функцию [48]. Такие сообщения называются флэш-сообщениями и работают следующим образом: перед переходом на новую страницу в куки-файле браузера пользователя сохраняется небольшой текст. При загрузке следующей страницы приложение читает куки-файл и показывает сообщение пользователю.

Я добавил библиотеку к `Gemfile` и `application.rb`, в соответствующих маршрутах написал сообщения, которые хотел бы видеть, затем добавил несколько строк программы в мои шаблоны ERB, чтобы показывать эти сообщения. Дело сделано. Мое приложение завершено.

Вот как начинается ознакомительный файл `Readme` приложения «Кодекс»:

«Codex» — это простое интернет-приложение, справочник для одного пользователя, написанное на языке Ruby. «Кодекс» использует Sinatra и DataMapper для того, чтобы создавать, сохранять, обновлять и удалять записи страниц в простой базе данных Postgres. Приложение готово для немедленного размещения на сервисе Heroku.

Форматирование Markdown доступно для всех страниц, что облегчает создание сложных страниц с простой разметкой. Сохранность информации обеспечивают аутентификация HTTP и принудительное SSL-шифрование для всего трафика. Дизайн Bootstrap делает страницы аккуратными и привлекательными.

Далее в файле `Readme` приводятся подробные инструкции по установке приложения на Heroku. Метод «Начни разработку с Readme» оказался весьма успешным.

Всего на разработку приложения «Кодекс» я потратил 10 часов. Таким образом, затраты времени на обучение программированию составили 20 часов. (Эту главу я писал дольше, чем программу.)

Когда приложение было готово, я присоединился к локальному форуму программистов, работающих на Ruby, и рассказал о том, как работает программа. Меня похвалили за то, каким ясным, компактным и понятным получился код. Один из участников форума заметил, что качество кода оказалось лучше, чем у многих проектов, выполненных профессиональными программистами.

Миссия завершена.

Ярость против машины

Хотелось бы кое-что прояснить: процесс, который я описал, выглядит линейным и прямым. Причина в том, что до сих пор я рассказывал о своих

успехах, а не о трудностях и ошибках.

Создание полезного, рабочего интернет-приложения немного похоже на сборку пазла, но с дополнительными трудностями: вы не знаете, какие фрагменты существуют, некоторые фрагменты необходимо создавать самому, а в случае ошибки рушится вся картинка.

На самом деле процесс программирования выглядел так. Когда я приходил к выводу, что понял, как должна работать та или иная часть программы, то писал код, тестировал его — и программа вылетала. Я пытался исправить ошибку. Это иногда получалось, иногда нет, а порой изменения приводили к новым проблемам. Если я заходил в тупик, то искал объяснение ошибки или нужную библиотеку на Stack Overflow или в Google.

В процессе обучения созданное вами приложение чаще не работает, чем работает. Кроме того, вы узнаете много полезного, например что такое контроль версий, позволяющий вернуться к предыдущей, рабочей версии программы.

Помните, я говорил, что до конца не понимаю, для чего нужен git? Оказывается, вот для чего: если вы редактируете файлы и программа вылетает, то найти причину бывает не так просто. В такой ситуации спасением становится возможность отката к предыдущей, рабочей версии. В противном случае есть опасность запаниковать.

На этапе добавления бокового поля мое приложение перестало работать. Я пытался выявить свою ошибку, но ничего не получалось. Я был готов рвать на себе волосы — если бы они у меня были.

Именно тогда я осознал ценность git branch и git merge: они позволяют создавать экспериментальные копии своей программы, а затем вносить изменения. Если копия заработала, вы можете сохранить изменения в оригинале. В случае ошибки вы просто удаляете экспериментальную ветвь, не теряя всей предыдущей работы.

Программирование — сложный процесс, и существует миллион возможностей сделать ошибку. Компьютер ничего не прощает и не выполняет неправильно сформулированные команды. Точно так же легко пропустить какие-нибудь мелочи, что приведет к непредсказуемым результатам. Я долго бился над ошибкой во фрагменте, связанном с сохранением пустых записей в базе данных, которые появлялись в списке страниц.

При каждом просмотре возникали фиктивные страницы, и я никак не мог понять, откуда они берутся. В конечном итоге выявилась ошибка в маршруте «показать все»: я использовал некорректную команду для извлечения записей из базы данных.

Каждая моя ошибка меня чему-нибудь учила. В этом заключается одно из скрытых преимуществ программирования. Компьютер — самый быстрый из всех возможных типов обратной связи. Когда вы что-то делаете неправильно, то мгновенно узнаете об этом. А когда правильно, то мгновенно видите результат.

Если вам удастся побороть желание швырнуть компьютер в стену, то мгновенная обратная связь может превратить программирование в весьма увлекательное занятие.

В начале этого эксперимента я совсем не умел писать программы. Теперь умею. Потребовалось потратить определенное время, чтобы сделать коды послушными, а также проявить упорство или выполнить кое-какие исследования, если я заходил в тупик.

Что же я приобрел, потратив столько сил? Очень много. Я узнал, что такое программирование, как оно выглядит и какая от него польза. Я научился с нуля создавать интернет-приложения на языке Ruby, а затем размещать их на рабочих серверах. Я познакомился с Sinatra, Heroku, Jekyll, DataMapper, Rake и другими универсальными инструментами, с помощью которых могу разрабатывать новые приложения. Я научился находить ошибки в программе и устранять их.

Применение метода

Давайте кратко опишем суть метода, который я использовал для обучения программированию.

- Я потратил время на получение общих представлений о программировании и интернет-приложениях, затем разбил эти навыки на более простые элементы, которые проще понять и которым легче научиться.
- Я определил целевой уровень мастерства, выбрав два конкретных проекта, а затем описал, как будет выглядеть конечный результат в каждом из них.
- Я разбил эти проекты на более мелкие этапы, а затем выделил из них самые значимые.
- Я убедился, что у меня есть все необходимые для работы инструменты (например, последняя версия Ruby) и что я смогу найти любые дополнительные инструменты, которые мне потребуются.
- Я нашел несколько надежных источников информации о программировании, но не стал читать заранее приобретенные учебные пособия, а решил сразу приступить к решению реальной задачи.
- Сначала я работал над самыми важными элементами, например выяснил, как протестировать программу на своем компьютере, как разместить готовое приложение на рабочем сервере и тому подобное.
- Я использовал примеры из справочных пособий, чтобы приступить к делу и приобрести некоторую уверенность, затем пробовал разные подходы, выясняя, как запрограммировать нужные мне функции.

- Если я что-то делал неправильно, программа прекращала работу, выдавая мне сообщение об ошибке, что обеспечивало быструю обратную связь.
- Сделав ошибку, я пробовал несколько способов ее исправления. Если я не мог решить проблему самостоятельно, то обращался за помощью.
- Я продолжал придерживаться подхода «кодирование/тестирование/отладка» до тех пор, пока программы не были окончательно готовы.

Общее время составило около 20 часов. 10 часов заняли исследования, а оставшиеся 10 — создание двух приложений, которые в данный момент закончены и используются.

Планы на будущее

Завершив эти два начальных проекта, я продолжил писать интернет-приложения, сосредоточившись на программах, которые могли облегчить мне ведение бизнеса.

Теперь я с гордостью могу заявить: весь мой бизнес использует программное обеспечение, созданное мной самим. Мои приложения способны принимать оплату с кредитных карт, оформлять подписку, отправлять письма по электронной почте и управлять доступом клиентов к интернет-сайту. Научившись программировать, я стал владельцем собственной маленькой армии роботов, выполняющих мои приказы.

Сколько времени у меня ушло на создание всех этих программ? Приблизительно 90 часов, включая те 20, которые подробно описаны в этой главе.

А вот еще дополнительное преимущество: когда я сталкиваюсь с каким-то повторяющимся или скучным аспектом своего бизнеса, то начинаю думать в терминах программирования. Как будет работать программа, решающая эту проблему? В большинстве случаев это помогает систематизировать процесс, что значительно облегчает мне жизнь.

Кроме того, я научился некоторым новым трюкам, например настраивать свой компьютер, чтобы ускорить процесс программирования. Я запомнил комбинации «горячих клавиш» текстового редактора, позволяющие сэкономить время, а также обновил программу Terminal до iTerm2 и Z-Shell, чтобы немного облегчить программирование.

Я все еще учусь, и меня ни в коем случае нельзя назвать специалистом. Мне приходится доходить до всего своим умом, и требуется определенное время, чтобы найти правильный путь и устранить ошибки. Зачастую это вызывает раздражение.

Тем не менее я пишу программы, которые решают реальные задачи, просто и надежно. И только это имеет значение.

Я сражался с компьютером и победил.

6

Печать слепым методом

Урок: старые привычки не обязательно живучи

Хорошее дело стоит того, чтобы сначала помучиться.

Рей Конгдон

До сих пор я приобретал новые навыки в тех областях, где практически не имел опыта. Естественно, отсутствие опыта поначалу было препятствием, но мой мозг, по крайней мере, не *вмешивался* в процесс обучения.

А что происходит, когда вы осваиваете новый навык, а ваш мозг восстает против этого?

Можно привести поистине драматические примеры переучивания. Известно, что знаменитый гольфист Тайгер Вудс трижды менял свой свинг (удар), уже достаточно эффективный [1]. Иногда стоит научиться делать нечто важное по-новому, еще лучше — даже если это приведет к кратковременной неэффективности и раздражению.

Современные технологии меняются так быстро, что переобучение стало привычным делом. Что произойдет, к примеру, если изменится программное обеспечение, которое вы используете для работы, или появится новая программа, позволяющая решать задачу эффективнее? Что если вы устроились на новую работу, требующую нового инструментария? Зачастую быстрое переучивание не менее важно, чем обучение чему-то новому.

Мне интересно: что я буду чувствовать, заново обучаясь делать нечто важное?

Я составил список того, что уже умею делать, а затем выбрал из него навыки, удовлетворяющие двум условиям: 1) я ими достаточно хорошо владею, и 2) они предлагают несколько методов для достижения одного и того же результата. Мне не потребовалось много времени, чтобы найти подходящего «кандидата».

Навык? Печать вслепую.

Жизнь за клавиатурой

По словам Дэвида Аллена, автора бестселлера «Как привести дела в порядок», если ваша работа требует использования компьютера, то обучиться печати вслепую будет самым существенным шагом к повышению собственной

производительности.

Критический порог составляет приблизительно 250 знаков в минуту. Если вы не в состоянии печатать вслепую со скоростью хотя бы 200 знаков в минуту при небольшом количестве ошибок, то серьезно снижаете производительность своей работы. Чем меньше усилий вы затрачиваете на набор текста, тем больше времени и энергии у вас остается на более важные задачи.

Я печатаю вслепую уже лет семнадцать, если не больше. В седьмом классе школы нам преподавали курс машинописи, и я помню, как это было скучно. Я уже умел печатать вслепую, поэтому выполнял задания за несколько минут, а остальное время тратил на то, чтобы заставить текстовый редактор отобразить всякие необычные символы, такие как знаки абзаца (¶), параграфа (§) или различные лигатуры.

Печатью вслепую я овладел в основном потому, что в свободное время много сидел за компьютером. Мое обучение было спонтанным, а не намеренным: я не занимался специально, чтобы повысить скорость или уменьшить количество ошибок. Я просто работал на компьютере и в процессе работы научился печатать вслепую.

Мою технику никак нельзя было назвать совершенной. Мои руки летали над клавиатурой, вместо того чтобы большую часть времени висеть над *основным рядом* клавиш в ее центре. Мой метод не был правильным, как в учебнике, но позволял справляться с задачей — и это меня устраивало.

Да, я печатаю непрофессионально, но зато функционально. По роду деятельности мне приходится много времени проводить за компьютером, а моей скорости и точности печати вполне достаточно для работы.

Тем не менее время от времени мне попадаются статьи об альтернативных раскладках клавиатуры: расположении клавиш, отличающемся от QWERTY, так называемой стандартной раскладке, которой снабжается большая часть англоязычных клавиатур, ежегодно выпускаемых в мире [2].



В этих статьях утверждается, что раскладка QWERTY очень неэффективна — а если точнее, просто ужасна. Существуют другие способы расположения букв на клавиатуре, которые позволяют печатать быстрее, с меньшим количеством ошибок и с меньшими затратами сил.

Тут очень важен кумулятивный эффект: печатание может вызвать или усугубить такие медицинские проблемы предплечий и запястий, как травмы, обусловленные постоянной нагрузкой, и кистевой туннельный синдром. Я еще не замечал у себя никаких симптомов, но у меня есть несколько друзей с подобными проблемами. Я бы хотел по возможности избежать этих болезней.

Я собираюсь еще много лет писать книги и программировать, и если

распознавание речи или чтение мыслей вскоре не станут главными способами общения с компьютером, то в обозримом будущем мне придется продолжать набирать текст на клавиатуре. Возможно, мне пойдет на пользу, если я научусь печатать самым эффективным из возможных методов, даже если придется пережить временное снижение производительности и дискомфорт.

Прощай, QWERTY: я собираюсь переучиваться печати вслепую.

Как раскладка QWERTY стала «универсальным» стандартом

Вопреки распространенному мнению, раскладку QWERTY не придумывали для того, чтобы замедлить скорость печати; это был способ устранения проблем механики пишущих машинок.

В прошлом, до появления текстовых редакторов и компьютеров, текст набирался на механических пишущих машинках, в которых *литерный рычаг* прижимал маленькую металлическую литеру к бумаге, туго обернутой вокруг цилиндра. Между литерой и бумагой была протянута чернильная лента. Литера ударяла по ленте, которая прижималась к бумаге, оставляя отпечаток символа. Переводной рычаг вызывал поворот цилиндра, бумага перемещалась вверх, и печать продолжалась с новой строки.

К. Л. Шоулз, которому приписывают создание раскладки QWERTY, сконструировал свой первый прототип в 1868 году. Шоулз не был первым изобретателем пишущей машинки — до него эту задачу пытались решить не меньше пятидесяти инженеров, и Шоулз внимательно изучил их работы, применив многое из придуманного ими в своей конструкции [3].

На первом прототипе Шоулза клавиши располагались в алфавитном порядке, что было не лишено смысла. В то время никто не мог вообразить, что люди будут (или даже смогут) печатать вслепую всеми десятью пальцами. Расположение клавиш в алфавитном порядке помогало неопытным пользователям, печатавшим двумя пальцами, найти нужную букву.

Но у первой модели машинки имелась одна неприятная особенность: литерные рычаги цеплялись друг за друга, если быстро нажимать соседние буквы. В английском языке чаще всего используются гласные A O E U I и согласные D H T N S. На механической клавиатуре с алфавитной раскладкой буквы S и T располагаются рядом. Если литерный рычаг одной из этих букв идет вверх, а другой вниз, то они цепляются друг за друга, и тогда приходится останавливаться и расцеплять их вручную.

Чтобы преодолеть этот хронический недостаток, Шоулз обратился за помощью к Эймосу Денсмору, учителю. Денсмор произвел приближенную оценку частоты использования букв английского алфавита. Результаты этого исследования Шоулз использовал для того, чтобы разнести буквы в часто

встречавшихся сочетаниях, таких как ТН, на разные стороны клавиатуры, предотвращая таким образом столкновение рычагов.

Эта стратегия полностью не устранила проблему, но до такой степени ослабила ее, что в 1872 году Шоулз получил патент на свою конструкцию [4]. В 1873 году патент приобрела компания Remington & Sons, раньше известная как производитель огнестрельного оружия.

После некоторой доработки, в том числе добавления клавиши Shift, позволявшей переключать регистры прописных и строчных букв, в 1874 году компания Remington начала массовое производство пишущих машинок с клавиатурой QWERTY, рассчитывая продавать их на деловом рынке.

Компания Remington была не единственным производителем пишущих машинок. Другие фирмы, например Hammond и Blickensderfer, предлагали конкурирующие модели, причем каждая имела свою раскладку клавиатуры.

В то время вся документация и деловая переписка велись вручную. Пишущие машинки могли существенно сократить долю ручного труда, но только при условии их должного использования. Чтобы обеспечить продажи, производители должны были избавиться от необходимости обучать пользованию необычным инструментом.

Это привело к интересным изменениям на рынке: производители пишущих машинок сами набирали и обучали машинисток и создавали агентства по найму. Если бизнесмену требовался работник, умеющий печатать на машинке, он обращался в Remington, и компания продавала ему пишущую машинку и услуги машинистки.

По мере распространения пишущих машинок раскладка клавиатуры QWERTY стала превращаться в стандарт. Это произошло постепенно, без кого-либо поворотного момента или бюрократического решения — просто медленное движение рынка к единому удовлетворительному решению.

Бизнесу требовались пишущие машинки и машинистки, и компания Remington справлялась с поставкой и тех и других. А когда предприятию требовалась новая пишущая машинка или новая машинистка, логичнее было покупать машинку QWERTY или нанимать человека, привыкшего к этой раскладке. За 60 лет QWERTY фактически превратилась в стандарт, а другие раскладки постепенно исчезли. По прошествии десятилетий QWERTY незаметно завоевала весь мир.

Появление конкурента: Dvorak

В 1932 году профессор Вашингтонского университета Август Дворак получил от Комиссии Карнеги грант в размере 130 тысяч долларов на разработку раскладки альтернативной клавиатуры. Одной из причин этого

исследования стало понимание того факта, что клавиатура QWERTY была разработана для разрешения проблемы «залипания» рычагов механики, но никак не способствовала снижению усталости рук машинисток. Можно ли придумать более удобную клавиатуру? Этим вопросом и занялся Дворак.

Четыре года спустя, в 1936-м, был выдан патент на «упрощенную клавиатуру Дворака», которая, как утверждалось, превосходила все остальные раскладки. Это утверждение основывалось вот на чем: в раскладке Дворака буквы, которые используются чаще всего, располагались в основном ряду, прямо под пальцами пользователя.



Самые распространенные согласные находились в правой части основного ряда, самые распространенные гласные — в левой. Такое разделение гласных и согласных равномерно распределяло нагрузку на обе руки, теоретически снижая утомляемость и повышая скорость печати.

Дворак также утверждал, что его раскладку легче выучить, и провел исследование в организациях, которые обучали большое количество машинисток, преимущественно для нужд армии. Результаты получились неоднозначными: поскольку Дворак проводил исследования сам, многие факты, подтверждающие его гипотезу, были поставлены под сомнение — из-за явной финансовой заинтересованности исследователя.

Независимое контрольное исследование, проведенное Администрацией общих служб США, которая отвечала за обучение машинисток для правительства и армии, показало, что машинисткам, привыкшим к QWERTY, после переучивания на раскладку Dvorak требуется около ста часов, чтобы восстановить скорость печати. Срок посчитали слишком большим, и поэтому администрация рекомендовала обучать машинисток для государственных учреждений на клавиатуре QWERTY. Производители печатных машинок и бизнес последовали их примеру.

Раскладка Dvorak продолжала существовать как один из экзотических вариантов, но стандартом не стала. Несмотря на многочисленные преимущества по сравнению с QWERTY, она не смогла изменить существующее положение дел. На протяжении нескольких десятилетий раскладка Dvorak оставалась «вторым номером», значительно уступая фавориту.

Новый вызов: Colemak

Кроме QWERTY и Dvorak существуют и другие раскладки английской клавиатуры. Несмотря на то что большинство людей учатся печатать на

QWERTY, изобретатели и энтузиасты не прекращают попыток создания новой клавиатуры. Большинство этих альтернативных раскладок не получили широкого распространения, но в 2006 году внимание пытливых пользователей привлекла одна новая клавиатура. Произошло это благодаря распространению интернета.

Вот как это было. Программист Шай Коулман решил попробовать свои силы в разработке клавиатуры. Цель его была простой: придумать раскладку, по эффективности сравнимую с Dvorak, но более легкую для обучения.

Одна из главных трудностей в освоении раскладки Dvorak состояла в том, что в ней менялось *все*: каждая буква находилась в другом месте по сравнению с QWERTY. Если вы уже умели печатать на QWERTY, то переучивание на Dvorak превращалось в настоящий кошмар.

Более того, в настоящее время главными инструментами для набора текстов стали компьютеры, и для упрощения своих действий — сохранения файла, вырезки и вставки фрагментов и так далее — мы часто прибегаем к сочетаниям «горячих клавиш». Если вы привыкли к этим сочетаниям, то раскладка Dvorak раздражает еще больше: переместились не только все буквы, но и все «горячие клавиши».

Стратегия Коулмана заключалась в следующем: он объединил компьютерный анализ большого количества документов на английском языке с идеей сохранить на своих местах как можно больше букв, чтобы не изменять «горячие клавиши». Левая сторона клавиатуры, а также нижний ряд почти не изменились. Переместились только самые важные буквы. Алгоритм Коулмана вычислил большинство необходимых изменений, и остались лишь несколько клавиш, требовавших субъективного решения.

В конечном итоге Коулман создал новую раскладку клавиатуры, которую назвал Colemak и в которой по сравнению с QWERTY изменилось расположение 17 символов. Статистика подтвердила, что она гораздо эффективнее QWERTY и немного эффективнее Dvorak. Но еще более привлекательным выглядел тот факт, что в Colemak изменялось месторасположение небольшого количества символов, и переучиться на нее, похоже, было гораздо легче.

Коулман создал свой сайт colemak.com, в котором подробно рассказывалось о новой раскладке, а также предлагались инструкции по установке и обучению. По сравнению с дорогостоящими усилиями Дворака, который проводил встречи в крупных организациях, популяризируя свою клавиатуру, интернет сделал распространение сведений о раскладке Colemak чрезвычайно дешевым.

В результате в настоящее время Colemak является третьей по популярности — после QWERTY и Dvorak — раскладкой англоязычной клавиатуры. Опция Colemak теперь по умолчанию встроена в большинство

новых компьютеров и операционных систем, и у нее есть тысячи поклонников во всем мире. Не так плохо для раскладки, разработанной любителем и существующей меньше десяти лет.

Если сомневаешься, проверь

Как выбрать лучшую из нескольких альтернативных раскладок клавиатур?

Очень просто: проверить.

Современные методы программирования значительно облегчают сбор достоверных данных о раскладках клавиатур. Вместо того чтобы тестировать и собирать сведения вручную, можно использовать программы анализа раскладок. Одна из таких программ под названием `carpalx` [5] разработана программистом Мартином Крживинским из канадского Центра генетики в Британской Колумбии.

`Carpalx` разработана для «выполнения стохастической симуляции с целью выявления раскладки, минимизирующей уровень усилий для данного набора параметров модели». Другими словами, `carpalx` может автоматически протестировать эффективность разных раскладок, применяя их к текстам-моделям.

Крживинский использовал программу `carpalx` для разработки собственной раскладки, однако с ее помощью он также сравнил Colemak с QWERTY и Dvorak, получив очень большой и полный объем сравнительных данных. Вот что он выяснил [6]:

QWERTY значительно отстает — по сравнению с Colemak у нее выше затраты энергии (+193 %), большее расстояние удара (+36 %) и существенно выше цена ошибки (+16 %). Dvorak также значительно превосходит QWERTY, и разница между этой раскладкой и Colemak невелика.

Основной ряд в Colemak используется активнее (74 %), чем в Dvorak (71 %), а QWERTY по этому параметру остается далеко позади (34 %). Нижний ряд и в Colemak, и в Dvorak используется редко (9 %).

В Colemak нагрузка на руки распределяется более равномерно — перевес правой руки составляет 6 % (у Dvorak 14 % для правой руки, у QWERTY 15 % для левой).

Colemak хорошо обеспечивает смену рук. И в Colemak, и в Dvorak больше используется мизинец. В Colemak мизинец используется в 16 % времени (в Dvorak 16 %, в QWERTY 10 %). Colemak больше нагружает сильные пальцы (указательный и средний), которые задействованы 67 % времени. Это лучше, чем у Dvorak (60 %), но хуже, чем у QWERTY (69 %).

Судя по данным Крживинского, работа на клавиатуре QWERTY требует

почти в два раза больше физических усилий, чем на Colemak. Кроме того, раскладка Colemak имеет небольшое преимущество перед Dvorak, что очень неожиданно, поскольку в ней изменяется положение лишь 17 клавиш по сравнению со стандартом QWERTY (в Dvorak — 24).

Основываясь на этих данных, я выбрал Colemak. Мне нравится, что я смогу сэкономить силы, не меняя привычных «горячих клавиш», которыми часто пользуюсь.

Что же дальше?

Как выглядит раскладка Colemak?

Вот расположение букв и символов на клавиатуре Colemak [7].



Изменилось положение не только наиболее часто используемых букв, но также клавиши Caps Lock, которая заменена вторым Delete. Это одна из главных инноваций Colemak. Никто (за исключением интернет-троллей) постоянно не пользуется клавишей Caps Lock, которая занимает удобное место на клавиатуре: легко доступное для мизинца левой руки, рядом с клавишей A в основном ряду.

Замена Caps Lock на Delete позволяет пользователям Colemak исправлять ошибки, не убирая правую руку с основного ряда, чтобы нажать Delete в правом верхнем углу, — это экономит много сил. Одно такое изменение приводит к уменьшению движений пальцев на 15–20 % по сравнению с QWERTY. Показатель эффективности Colemak — это количество ошибок, исправленных мизинцем.

Как переключиться на Colemak?

На компьютере изменить раскладку клавиатуры легко: просто указать, чтобы машина использовала другой файл раскладки, как при переключении на другой язык. Обычно эта настройка располагается в разделе вкладки «Systems and Preferences». Colemak включена в операционную систему Mac OS X начиная с версии 10.5, и поэтому ничего загружать не требуется [8].

Кроме того, некоторые клавиатуры (например, TypeMatrix 20309) поддерживают Colemak на уровне аппаратной части. Вместо того чтобы конфигурировать компьютер для распознавания раскладки, клавиатура преобразует нажатие клавиши на Colemak в коды QWERTY, так что на экране

появляются нужные символы. В результате вы можете печатать на раскладке Colemak, вообще ничего не изменяя в настройках компьютера, разве что потратить сто долларов на красивую, эргономичную клавиатуру [10].

Изменить настройки компьютера легко, однако это не единственное препятствие. Поскольку большинство англоязычных клавиатур выпускаются с раскладкой QWERTY, переключение на Colemak означает, что выгравированные на клавишах буквы не будут соответствовать тому, что появляется на экране. Если вам нужно ввести бессмысленный набор знаков, например сложный пароль, это может вызвать затруднения. Где же взять клавиатуру с раскладкой Colemak?

Есть два варианта действий: купить клавиатуру с пустыми клавишами или превратить клавиатуру QWERTY в Colemak.

Некоторые клавиатуры, в частности TypeMatrix 2030, доступны в варианте Colemak — и это наилучший вариант, если вам нужна новая клавиатура. Но мне он не подходит: я работаю на ноутбуке со встроенной клавиатурой.

Модификация компьютера

К счастью, в компьютерах Apple клавиши легко снять, а затем поставить в нужном порядке, не повреждая компьютер. Поскольку в Apple используются плоские клавиши одинакового размера, изменение раскладки на Colemak занимает не больше пяти минут [11].



При помощи маленькой плоской отвертки я аккуратно приподнимал верхнюю правую часть каждой клавиши, вставлял конец отвертки и двигал вниз вдоль правого края клавиши. Когда отвертка доходила до середины, клавиша выскакивала из специального держателя, который заставляет ее возвращаться на место после нажатия.

Освоив эту операцию, я без труда снял те клавиши, которые нуждались в перестановке, а затем, руководствуясь диаграммой новой раскладки, установил их на соответствующие позиции. Для этого требовалось лишь слегка нажать на них, и они со щелчком вставали на место.

Процедура очень простая, но она, по всей видимости, аннулирует гарантию на мой ноутбук. Поэтому из соображений безопасности я сначала потренировался на запасной беспроводной клавиатуре Apple, а затем, освоив

процесс, сменил клавиши на моем MacBook Air.

Теперь у меня ноутбук с клавиатурой Colemak. Круто, правда?

Это пример организации окружения таким образом, чтобы облегчить начало занятий. Теперь, когда у меня клавиатура Colemak, переключиться на нее будет легче. Если я забуду, где находится та или иная клавиша, то всегда могу подсмотреть.

С какой скоростью я печатаю?

Клавиатура готова, и приближается пора больших перемен. Но перед началом занятий мне хотелось бы получить представление о скорости, с которой я набираю текст. Это поможет определить целевой уровень мастерства.

В данном случае сформулировать целевой уровень мастерства довольно просто: я хочу на раскладке Colemak как можно быстрее достигнуть той же скорости, что и на QWERTY. Превышать ее я не собираюсь, поскольку скорость набора текста не является определяющей в моей работе. Мне достаточно печатать с той же скоростью, но прилагая меньше усилий.

Единственная информация, которая мне для этого нужна, — скорость печати, и поэтому я нахожу онлайн-тест на проверку скорости [12].

Этот тест прост: когда вы щелкаете по кнопке Start, программа выводит на экран сто случайных слов из старинной книги [13]. Ваша задача — напечатать образец как можно быстрее, стараясь как можно меньше ошибаться. Закончив, вы щелкаете по кнопке Stop, и программа сообщает вам скорость печати и количество допущенных ошибок.

Я намеревался сначала пройти тест на раскладке QWERTY, а затем перейти на Colemak и повторить — просто для того, чтобы определить начальный уровень.

Закончив тест, я щелкнул по кнопке Stop, и программа сообщила результат: 250 знаков в минуту и ни одной ошибки. Неплохо: я печатаю на том уровне, который рекомендует Дэвид Аллен. До рекорда скорости мне далеко, но этого вполне достаточно, чтобы справляться со своей работой.

Теперь у меня есть все, что нужно: мне известна моя скорость печати, у меня клавиатура Colemak, и я знаю, как переключить раскладку клавиатуры на компьютере. Подготовительный этап завершен.

Ну вот. Готов ли я отказаться от QWERTY?

Щелкаем переключателем

Теперь или никогда. Я переключаю свой компьютер на раскладку

Colemak и закрываю окно настроек. С этого момента я не смогу ничего печатать на QWERTY, пока не закончу эксперимент.

В моем браузере по-прежнему загружен тест скорости печати. Щелчком мыши я запускаю таймер и начинаю набирать текст.

Вот что у меня выходит: #%&@.

Возможно, это звучит несколько театрально, но я *чувствую себя так, словно у меня удалили часть мозга.*

Я привык, что слова безо всяких усилий переносятся из моей головы в компьютер. Теперь же я не знаю, где расположены *все* клавиши. Мне приходится искать их, даже те, которые не изменились относительно раскладки QWERTY.

Я вижу слова на экране, мои пальцы автоматически нажимают на клавиши, а в результате на экране появляется абракадабра, которую я вынужден стирать. На простое слово требуется несколько секунд, и я начинаю волноваться, что клавиша Delete выйдет из строя от частого употребления.

Каждое слово — это новая битва. Я смотрю на часы: два простых предложения съели несколько минут. Я не сделал еще и половины. Меня так и подмывает все бросить, но я заставляю себя продолжать. Минуты идут, и раз десять я с трудом удерживаюсь, чтобы не прервать тест.

Мне потребовалось почти 20 минут, чтобы набрать 100 слов. Вот моя новая скорость: 5 слов в минуту.

Для меня это «смерти подобно».

Я зарабатываю тем, что пишу, и совсем недавно научился программировать. Теперь я лишен возможности заниматься и тем и другим. Как я буду отвечать на письма? Как буду работать? Что я натворил?

Я увидел врага, и этот враг — я сам

Вот он, самый главный барьер, мешающий большинству людей перейти на новую раскладку клавиатуры. Технические аспекты перехода — ничто по сравнению с *эмоциональными*.

Когда вы достигли определенного уровня скорости или легкости выполнения какой-либо задачи, снижение производительности повергает вас в ужас. Особенно это справедливо для машинописи: вы привыкли печатать без всяких усилий, и необходимость напрягаться вызывает возмущение. Еще больше усиливает недовольство понимание того факта, что если вы вернетесь к привычному образу действий, то ситуация мгновенно выправится.

Разум здесь не поможет: у нашего мозга есть неустранимая тенденция считать, что текущее положение дел сохранится и в будущем. В данный момент мой мозг в полной растерянности. Я способен печатать со скоростью лишь пять

слов в минуту, и я больше никогда не смогу работать! Моя карьера рухнет! Моя семья будет голодать!

Разумеется, это неправда, но именно так мне в данный момент *кажется*. Подобные эмоции — главное препятствие для обучения.

Я закрываю ноутбук, беру блокнот и ручку. Мне нужен — причем немедленно! — план, как добиться приемлемой скорости на раскладке Colemak.

Перестраиваем мозг

Первая и самая главная проблема состоит в том, что я не знаю, где располагаются новые клавиши. Разумеется, я распечатал рисунок, и буквы на моей клавиатуре расположены в формате Colemak, но мозг пока не может связать желание напечатать определенную букву с движением пальца.

Поэтому в первую очередь я должен помочь мозгу преобразовать новую раскладку в моторику. Я должен как можно быстрее восстановить способность работать.

К счастью, люди обучаются печати слепым методом уже не одно десятилетие и придумали эффективные средства, которые способны мне помочь. Такие учебные пособия по машинописи, как Mavis Beacon [14] и Typing Trainer [15], выпускаются уже давно и стоят совсем недорого.

К сожалению, эти программы учат слепому методу печати в формате QWERTY, поскольку большинство клиентов хотят иметь дело именно со стандартной раскладкой. Обучение, как правило, начинается с клавиш основного ряда. Однако основные ряды в QWERTY и Colemak отличаются, и поэтому пособия по QWERTY мне ничем не помогут.

Раскладка Dvorak существует довольно давно, и некоторые программы поддерживают ее, а вот формат Colemak относительно новый. Наверное, было бы неплохо использовать обучающую программу, но мне нужно найти ту, которая поддерживает Colemak.

К счастью, у меня есть зацепка. Просматривая сайт Hacker News во время обучения программированию, я наткнулся на пост о Keyzen [16], программе с открытым кодом, которая как раз обучала машинописи; разработал ее Рай Террелл. Это приложение предназначалось для обучения программистов. Программа запускалась в обычном браузере и включала необычные символы — круглые и квадратные скобки, косую черту, — которые часто используют программисты; большинство обучающих программ эти символы, как правило, пропускают. Террелл опубликовал полный текст программы на GitHub и предложил программистам использовать или модифицировать ее.

Сама программа крайне проста: на экране отображается семь символов, начиная с букв основного ряда. Ваша задача — напечатать символы в

указанном порядке.

Во время работы программа издает звуки. Если вы нажали правильную клавишу, то слышите щелчок, как при ударе пишущей машинки; при ошибке раздается резкий звук, и символ становится красным.

Как только вы заканчиваете печать, программа предлагает следующий набор символов. Если вы делаете ошибку в каком-то символе, программа автоматически включает его в следующие наборы. Если вы не сделали ошибок три раза подряд, то раздается мелодичный звон, означающий, что вы перешли на следующий уровень. Программа добавляет новый символ, и тренировка продолжается.

Модификация Keyzen

Как и большинство программ обучения машинописи, по умолчанию Keyzen поддерживает раскладку QWERTY. Но поскольку у программы открытый код, мне, возможно, удастся модифицировать ее под раскладку Colemak.

Вот мне и пригодилось обучение программированию.

Я скопировал исходный код Keyzen, затем открыл программные файлы на своем компьютере. Сама программа оказалась простой, и я без труда нашел ту часть, которая управляет выводом символов на экран; организовать нужный мне вывод тоже оказалось несложно.

Оригинальная программа начинала с основного ряда раскладки QWERTY, с указательных пальцев, затем добавляла по одному символу, пока не заканчивался основной ряд. За ним шел верхний ряд, расходясь с середины к краям, затем нижний, в том же порядке.

Взяв эту схему обучения за образец, я отредактировал программу для формата Colemak. Сами клавиши остались теми же, но обучение теперь шло в раскладке Colemak, а не QWERTY.

Закрыв файлы, я получил собственную программу обучения машинописи в раскладке Colemak. Ура!

Если вам интересно, можете попробовать ее сами — она выложена на интернет-странице <http://first20hours.com/keyzen-colemak>.

Мелкая моторика

Итак, у меня есть обучающая программа, а моя первая цель проста — выучить расположение символов на клавиатуре.

Я запускаю Keyzen и начинаю печатать («_» означает пробел):

nn_nn_n
n_nn_n_
tntt_tn
t_tntt_
ttttnt

Подобные упражнения не очень эффективны, но необходимы. Тренируясь находить отдельные буквы, появляющиеся в случайном порядке, я помогаю мозгу преобразовать символ в мелкую моторику пальцев.

Кроме того, Keyzen помогает эффективному обучению несколькими способами. Во-первых, раскладка Colemak разработана так, что наиболее часто используемые буквы располагаются в основном ряду, под самыми сильными пальцами. Программа выводит эти буквы первыми, и поэтому я сначала осваиваю ARSTDHNEIO, а затем перехожу к QWZXУМ.

Когда я ошибаюсь, Keyzen добавляет неправильно набранный символ в следующую последовательность. В результате бо2льшую часть времени я тренируюсь набирать буквы, в которых делаю ошибки. Это одна из разновидностей интервального повторения, которое сочетается с мгновенной обратной связью, что делает занятия чрезвычайно эффективными.

Я составил график занятий: не меньше двух сеансов по 20 минут с коротким перерывом между ними, в конце дня перед сном.

С ума сойти! Приобретение нового моторного навыка требует *физических изменений в структурах мозга*, и, как выяснилось, сон способствует этому процессу.

Учитесь во сне!

В последние четыре десятилетия приобретение моторных навыков было предметом активных исследований психологов. В любой научной библиотеке вы найдете не один стеллаж с работами о приобретении навыков в целом и моторных навыков в частности.

Именно так я и поступил. Я бродил между стеллажами книг по когнитивной психологии в библиотеке Университета штата Колорадо в поисках полезной информации. Найти книги и отчеты об исследованиях не составляло труда. К сожалению, большая часть этих отчетов очень нудная, изобилующая профессиональным жаргоном.

Тем не менее обнаружили и исключения. Мое внимание привлекла одна работа.

«Стадии формирования моторного навыка» (Stages of Motor Skill Learning) (2005) [17]

Успешное обучение моторному навыку требует повторных тренировок...

Эта статья посвящена растущему числу свидетельств того, что формирование моторных навыков проходит несколько стадий, на каждой из которых задействуются разные механизмы хранения информации. Фаза выработки навыка характеризуется быстрым (во время сеанса) и медленным (между сеансами) обучением. В короткий период непосредственно после первых сеансов обучения навык подвержен интерференции со стороны других навыков и вследствие замедления синтеза белка, и это указывает на то, что процесс консолидации имеет место в периоды отдыха между тренировками. Во время занятий, а также в периоды отдыха изменяется активность разных отделов мозга.

Ключевые термины здесь — «интерференция» и «консолидация». Когда вы тренируете какой-то навык, ваш мозг необыкновенно активен — он стремится найти общие схемы и сохранить их в памяти. В случае моторных навыков эти схемы связывают то, что вы видите, слышите, чувствуете на ощупь или на вкус, с возбуждением нейронов, которые управляют мышцами вашего тела. Чем прочнее нервные связи, тем лучше у вас получается.

Во время занятий мозг занят формированием этих связей и ассоциаций, но это не значит, что они мгновенно сохраняются в структуре нейронов. Для их формирования требуется определенное время, и этот процесс называется *консолидацией памяти*. Консолидация происходит непрерывно, но с наибольшей эффективностью во время сна.

Вот одна из первых обнаруженных мной работ, которая описывает прямую связь между сном и приобретением навыка.

«Практика и сон — залог успеха: Влияние обучения во сне и пластичности на совершенствование навыка» (It's Practice, with Sleep, That Makes Perfect: Implications of Sleep-Dependent Learning and Plasticity for Skill Performance) (2005) [18]

Тренировку часто считают единственным определяющим фактором совершенствования навыка. Несмотря на то что постоянное повторение новой операции часто благотворно влияет на обучение, благодаря чему появился известный афоризм «Дело мастера боится», многочисленные исследования, проведенные за последнее десятилетие, постепенно меняют эту точку зрения. Их результаты дают основание предположить, что после первоначального сеанса обучения мозг человека продолжает обучаться и в отсутствие дальнейшей практики. Это «отложенное совершенствование» происходит во время сна.

Вот какой вывод делается в этой работе:

Несмотря на то что работа спящего мозга остается для нас загадкой, быстро растущее число исследователей поддерживает теорию о важной роли

сна в укреплении памяти. Они приводят множество данных, указывающих на то, что связанные со сном механизмы нейронной пластичности ведут к консолидации навыка и, следовательно, к его отложенному совершенствованию. Разные формы памяти простых и сложных навыков, по всей видимости, требуют разных типов сна для укрепления памяти. Некоторые исследования указывают на важную роль сна в течение первых 24 часов после начальной тренировки.

Похоже, эффективное приобретение навыка, особенно моторного, требует сна, который играет важную роль в консолидации навыка в долговременную память. Новейшие исследования позволяют предположить, что для максимального эффекта лучше лечь спать в течение четырех часов после тренировки моторного навыка: даже короткий сон лучше, чем ничего. Если пройдет больше четырех часов, то способность мозга консолидировать информацию, собранную во время практики, ослабевает.

Вот почему я занимался машинописью непосредственно перед сном. Если я лягу спать примерно через час после занятий, то помогу мозгу эффективнее консолидировать моторные навыки.

Странно, но я *вижу, что это работает*. Мое первое занятие было ужасным: я все делал неправильно, постоянно ошибался и с трудом преодолел буквы основного ряда. После полноценного ночного сна я сел за компьютер и вдруг обнаружил, что делаю меньше ошибок. Сон консолидировал то, чему я научился накануне вечером.

Наш мозг действительно крут.

Когнитивная интерференция

Интерференция противоположна консолидации; это разрушение процесса консолидации. Если вы тренируете другой, схожий навык (или просто пользуетесь им) вскоре после занятий новым, это может ослаблять способность мозга консолидировать полученную информацию.

Критический период интерференции, по всей видимости, тоже составляет около четырех часов. Если подождать, пока не завершится процесс консолидации, а потом практиковать конфликтующий навык, то он не будет в такой степени мешать укреплению нового.

Вот почему я не сажусь за клавиатуру QWERTY сразу же после занятий с раскладкой Colemak. Это ослабит способность мозга консолидировать мои занятия с Colemak, снизит скорость приобретения навыка.

Интересно также отметить, что после семи часов занятий с раскладкой Colemak я вдруг обнаружил, что мне трудно перейти на формат QWERTY, хотя я уже очень давно печатаю на стандартной клавиатуре слепым методом. Мой

мозг переключает моторику на Colemak и, похоже, затрудняет восприятие QWERTY — по крайней мере, на время.

После того как мой мозг привыкнет к раскладке Colemak, я при желании смогу вернуться к QWERTY: судя по тому, что я читал у других пользователей Colemak, можно стать «клавиатурным билингвом» и при необходимости переходить с одной раскладки на другую. Однако теперь я сосредоточиваюсь в основном на Colemak, чтобы минимизировать интерференцию. Восстановление навыка работы с QWERTY подождет.

Избавиться от привычки подглядывать

Теперь у меня за плечами семь часов намеренной практики, в среднем по 45 минут за вечер. Я без особого труда печатал все последовательности программы Keyzen. Ошибки по-прежнему случаются, но уже гораздо реже.

В течение дня у меня есть небольшая вынужденная практика — ответы на не терпящие отлагательства электронные письма. Когда я сажусь за клавиатуру, то выясняется, что все не так страшно. Печатаю я медленно, но способен выразить свою мысль. Уже прогресс!

Тем не менее я замечаю, что часто смотрю на клавиатуру. Правильные обозначения клавиш очень удобны при вводе бессмысленного набора знаков, например паролей, однако в случае сомнений возникает искушение опустить взгляд. Если я хочу научиться печатать вслепую, то должен как можно быстрее избавиться от вредной привычки. Но сделать это трудно — взгляд опускается сам собой.

Das Keyboard

Чтобы избавиться от привычки подглядывать, я решил приобрести новое средство обучения: клавиатуру без обозначений символов на клавишах.

Das Keyboard — самая крутая клавиатура, которую я когда-либо смогу найти. В модели Ultimate Model S [19] клавиши вообще пустые. Не умея печатать вслепую, вы не сможете пользоваться Das Keyboard... и точка. Забавно наблюдать за реакцией людей, которые впервые видят эту клавиатуру, — теряются даже опытные пользователи.

Я подключил Das Keyboard и закрыл клавиатуру ноутбука листом бумаги, чтобы вообще не видеть обозначения клавиш. Ощущение примерно такое же, как при первой попытке печатать в раскладке Colemak: растерянность и раздражение. Но только на секунду. Включились моторные навыки, приобретенные за первые семь часов занятий, и выяснилось, что я могу

печатать вполне сносно. Забывая букву, я охотился за ней в течение нескольких секунд методом проб и ошибок, но все же мог работать.

Поскольку смотреть было больше не на что, я сосредоточился на экране. Клавиатура Das Keyboard сделала свое дело. Сменив клавиатуру, я автоматически изменил свое поведение.

Вместе с клавиатурой изменениям подверглась и методика занятий. Печать отдельных символов уже ничего не давала, и я переключился на программу под названием Type Fu [20]. Помимо отдельных символов и слов, Type Fu содержит базу данных пословиц и цитат, что делает занятия более интересными. Программа также следит, в каких символах вы ошибаетесь чаще, и это очень удобно. Я обнаруживаю, что в данный момент труднее всего мне даются буквы J, U, V и B.

Каждый вечер я занимаюсь по 45 минут. После 14 часов намеренной практики я печатаю со скоростью 170 ударов в минуту.

На этом этапе я прекрасно справляюсь с работой: могу свободно отвечать на письма и искать информацию в интернете. Мне даже удалось без особых усилий напечатать пятистраничное коммерческое предложение. Это заняло больше времени, чем обычно, но не стало самым неприятным занятием в моей жизни.

Намеренная практика и фоновая практика

Теперь, когда я снова могу работать, мне хочется проверить одну гипотезу: насколько важна намеренная практика?

В настоящее время я тренируюсь двумя способами. Сеансы работы с программами Keyzen и Type Fu — это *намеренная* практика, поскольку я сосредоточен на поставленной задаче и активно работаю над совершенствованием навыка. Набор текста в течение дня — это *фоновая* практика. Печатаю электронное письмо или заметку, я работаю в раскладке Colemak, хотя сосредоточен не на технике набора, а на содержании сообщения.

Мне интересно: что произойдет, если на некоторое время прервать намеренную практику и просто продолжить писать письма и бродить по интернету? Две трети пути к целевому уровню мастерства, определенному как 250 ударов в минуту, я прошел всего за 14 часов намеренной практики. Можно ли преодолеть остаток пути с помощью фоновой практики, не прилагая дополнительных усилий?

Я решил провести эксперимент: на 30 дней прервать намеренную практику и посмотреть, что произойдет. Я продолжу работать в раскладке Colemak, не возвращаясь к QWERTY. Может, для достижения скорости в 250 ударов в минуту будет достаточно одной фоновой практики — с учетом того,

сколько времени я провожу за компьютером?

По истечении 30 дней я снова выполнил тест на скорость печати. Угадайте, сколько получилось?

170 ударов в минуту. Никакого прогресса.

Несмотря на то что печатать мне приходилось довольно много, я не прилагал усилий для совершенствования навыка. Фоновой практики недостаточно для прогресса.

Для совершенствования навыка необходима намеренная практика — по крайней мере на первых этапах. Урок усвоен.

Последний рывок

Возвращаемся к намеренной практике. Я хочу провести еще один эксперимент. Человеческие языки, в том числе английский, подчиняются степенной зависимости, получившей название закона Ципфа: чаще всего используется небольшое количество слов. Анализ «Брауновского корпуса» («The Brown Corpus» (1964)) — сборника 500 документов на современном английском языке объемом в 1 миллион слов, показывает, что 50 % использования приходится всего на 135 слов [21]. Артикль «the» занимает долю в 7,5 %, а предлог «of» — 3,5 %.

Эту идею можно развить: во всех словах есть сочетания из двух или трех букв, которые встречаются постоянно, например TH, AN, ING и NCE. Такая группа называется *n-грамма* (иногда *n-граф*); в данном случае *n* — это переменная, обозначающая количество символов в группе.

Мне удалось найти список самых распространенных *n*-грамм в книге под названием «Криптологическая математика» («Cryptological Mathematics») Роберта Эдварда Леванда (2000). *N*-граммы широко применяются в криптографии. Если в зашифрованном сообщении удастся обнаружить некие закономерности, они могут стать ключом к содержанию. Сравнивая *n*-граммы шифрограммы с наиболее распространенными *n*-граммами языка, на котором она написана, криптографы получают возможность разгадать самые сложные шифры.

Попробуем соединить теорию с практикой. Вот составленный Левандом список наиболее часто распространенных англоязычных 2-грамм (*диграмм*) в порядке убывания частоты использования:

th, he, in, en, nt, re, er, an, ti, es, on, at, se, nd, or, ar, al, te, co, de, to, ra, et, ed, it, sa, em, ro

А это самые распространенные 3-граммы (*триграммы*), тоже в порядке убывания частоты использования:

the, and, tha, ent, ing, ion, tio, for, nde, has, nce, edt, tis, oft, sth, men

Эти списки очень полезны. Чем лучше я буду справляться с этими сочетаниями букв, тем выше будет общая скорость печати.

Наш мозг прекрасно приспособлен для решения такого рода задач. В науке о мышлении для описания процесса приобретения навыков, следующих в определенном порядке, используется такой термин, как *процедурная память*. Печатая самые распространенные n-граммы, я буду напрямую тренировать процедурную память, участвующую в процессе набора текста.

Для этого я загрузил бесплатную программу под названием Amphetamine [22], предназначенную именно для такой практики. Программа позволяет самому задавать обучающие последовательности, а также критерии успеха, такие как скорость печати или процент ошибок.

Начиная сеанс работы в Amphetamine, вы можете указать программе, каким образом генерировать обучающую последовательность. Я настроил программу так, что каждая n-грамма повторялась три раза, в каждой последовательности присутствовали три n-граммы, а каждая последовательность повторялась три раза.

В результате первая обучающая последовательность выглядела следующим образом:

th he in th he in th he in en nt re en nt re en nt re er an ti er an ti er an ti

Моя цель — напечатать всю последовательность со скоростью не меньше 250 ударов в минуту и точностью не ниже 95 %. Если результат не будет удовлетворять этим критериям, всю последовательность придется повторить.

Такого рода упражнения не назовешь очень увлекательными, но они чрезвычайно эффективны. Я повторял каждую последовательность снова и снова, пока не добивался желаемого результата, а затем переходил к следующей.

Каждый день, приступая к занятиям, я начинал с самого начала. Прогресс был заметен невооруженным глазом: если вчера какую-то последовательность приходилось повторять пять раз, то сегодня всего один или два раза. Прошло совсем немного времени, и я уже без проблем справлялся со всем набором обучающих последовательностей.

Освоив диграммы, я перешел к триграммам. Затем я нашел список самых распространенных английских слов, составленный доктором Питером Норвигом.

Питер Норвиг в настоящее время занимает должность директора по исследованиям корпорации Google. Несколько лет назад он опубликовал список самых распространенных английских слов, полученный в результате анализа «корпуса миллиарда слов», содержащего все уникальные слова, когда-либо проиндексированные поисковой машиной Google [23].

Публикуя эти данные, Норвиг хотел помочь программистам создать полезные утилиты, такие как проверка орфографии, но в моем случае они

представляли собой превосходную обучающую последовательность. Я взял сотню самых распространенных слов, загрузил их в Amphetamine и продолжил тренировки.

Еще через восемь часов намеренной практики я снова выполнил тест на скорость печати. Результат (среднее из нескольких попыток, чтобы исключить случайность): 60 слов в минуту, с ускорениями до 70 и 80, при точности 98 %. Общее время намеренной практики составило 22 часа.

Миссия завершена!

Впечатления от скорости 250 ударов в минуту

Мне очень нравится работать с раскладкой Colemak. Я больше не раздражаюсь. Наоборот, мне теперь непонятно, как я столько времени терпел QWERTY.

При работе с QWERTY создается впечатление, что руки летают над клавиатурой: множество движений во все стороны. Сама раскладка вообще лишена логики — часто используемые буквы находятся в самых неудобных местах клавиатуры.

В Colemak пальцы почти неподвижны, а текст появляется на экране. Руки перемещаются явно меньше, вы гораздо реже обращаетесь к клавишам верхнего и нижнего ряда, и это экономит время. Я рад, что отказался от QWERTY, и даже представить себе не могу, что вернусь к этой раскладке.

Применение метода

Давайте кратко опишем суть метода, который я использовал для обучения печати вслепую.

- Я узнал, как изменить раскладку клавиатуры на Colemak.
- Я создал петлю быстрой обратной связи, переставив клавиши на клавиатуре компьютера, чтобы без труда найти символ, если я забуду его местоположение.
- Я использовал обучающую программу Keyzen, чтобы вслепую находить нужные символы, начиная с тех, которые задействуются чаще всего. В случае ошибки Keyzen снова включала данный символ в обучающую последовательность, так что бо2льшую часть времени я потратил на трудные символы, уменьшая число ошибок.
- Я занимался каждый вечер по 45 минут, непосредственно перед сном, чтобы мозг получил возможность эффективно консолидировать моторные навыки в долговременную память.
- Достигнув функциональной скорости (20 слов в минуту), я перешел к программе Type Fu, сосредоточившись на печати предложений — как можно быстрее, с точностью 99%.
- Достигнув скорости 40 слов в минуту, я использовал программу Amphetype, чтобы тренировать ввод самых распространенных английских диграмм и триграмм, что позволило повысить скорость и уменьшить число ошибок.
- Справившись с последовательностями n-грамм, я перешел к занятиям с

использованием самых распространенных английских слов (в программе Amphetamine), пока не достиг скорости 60 слов в минуту с 98 % точности, что произошло после 22 часов занятий.

Планы на будущее

Теперь, когда я уверенно печатаю со скоростью 250 ударов в минуту, у меня нет потребности продолжать занятия, чтобы увеличить скорость. На этом уровне я могу набирать текст настолько быстро, насколько это требуется. Скорость печати больше не влияет на конечный результат, следовательно, работа над скоростью печати больше не относится к числу моих приоритетов.

Скоростная машинопись — это отдельный навык. Чемпионы по набору текста могут печатать со скоростью около 400 ударов в минуту, но улучшение результатов теста скорости печати не обязательно приводит к повышению скорости написания программ или статей. В стандартных тестах скорости печати вы обычно набираете то, что видите, и поэтому в первую очередь имеет значение умение читать следующий фрагмент текста и удерживать его в кратковременной памяти достаточно долго, чтобы пальцы успели его напечатать.

Конечно, мне хотелось бы печатать готовую прозу со скоростью, превышающей 450 ударов в минуту, но этого не могут даже чемпионы по скорости машинописи. Набор текста становится более трудным и медленным процессом, когда вам нужно сочинить то, что потом появится на странице.

Тем не менее, я могу извлечь пользу из уменьшения количества ошибок. Поэтому я продолжаю упражняться в печати часто встречающихся слов, диграмм и триграмм с помощью программы Amphetamine, а также использовать мизинец левой руки для исправления ошибок. Снижение числа ошибок приведет к повышению производительности, при этом может также возрасти скорость.

При переходе на раскладку Colemak больше всего меня удивило, с какой легкостью можно забыть почти 20 лет практики на раскладке QWERTY. Мне казалось, что для замены двух десятилетий мышечной памяти потребуются не 20 часов, а гораздо больше. Я ошибался.

Наш мозг изменить легче, чем мы думаем.

7

Го

Урок: исследуйте, а затем решайте

Го использует самые простые материалы и понятия: линию и точку, дерево и камень, черное и белое, — соединяя их простыми правилами для создания изящной стратегии и сложной тактики, поражающих воображение.

Ивамото Каору, профессиональный игрок в го, мастер 9-го дана

Я — меч во тьме, я — Дозорный на Стене[12].

Ночь холодна: идет снег. За моей спиной колеблющийся свет факелов освещает фигуры соотечественников. Мы стоим на зубчатой стене замка и глядяемся во тьму. Мы в ожидании.

У дальнего конца моста через ров, вне досягаемости наших катапулт, собирается вражеское войско, готовясь нанести удар. Солдаты, воры и бродяги точат свое оружие. Некроманты призывают на помощь армии отвратительных зомби. Инженеры готовят мортиры и огненные тараны — мощное оружие, предназначенное для того, чтобы пробить наши ворота.

Эта орда хочет украсть самое ценное сокровище моего королевства — сияющую державу великой власти. Им ее не видать. По крайней мере, пока я на посту.

Мы хорошо подготовились. По обе стороны от меня солдаты окружили баллисты, готовые обрушить смертоносные заряды на атакующих. Заклинатели духов приготовились метать огонь и молнии во вражеские орды. Наши хранители и маги расставили по периметру руны и отражающие щиты, гарантирующие защиту от стрел и заклинаний, по крайней мере на время.

Мы ждем. Снег все еще идет, и свет факелов отражается от полированной стали моих доспехов.

Мы не дрогнем. Мы не отступим. Мы будем сражаться, и мы победим.

Из темноты появляется громадный камень и пролетает над вражеским войском. Булыжник ударяет в ворота замка и разбивается на куски, круша оббитое железом дерево.

Враг с ревом устремляется по мосту.

В ту же секунду у меня в руках появляются меч и щит. Мой боевой клич отражается от каждого камня, донося мою силу и ярость до всех защитников стены.

«ЗА ВЫСШУЮ СПРАВЕДЛИВОСТЬ!»

Сложите оружие — я не причиню вам вреда

Командовать армией во время яростного штурма крепости — превосходное развлечение в субботу вечером.

Я не поклонник телевизора и кино, не увлекаюсь спортом. Скорее, вы застанете меня играющим в видеоигры, такие как World of Warcraft или Guild Wars 2 — я могу потратить на это час или два.

Я очень люблю игры — особенно те, где нужно творить заклинания, сражаться с чудовищами и превосходить врагов в хитрости. Мне с детства нравились эпопеи о волшебниках, воинах и подвигах, а видеоигры предоставляют возможность поучаствовать в подобных приключениях, хотя бы недолго [1].

Мое поколение было первым, которое выросло в окружении компьютерных или видеоигр. Начавшись с приставок Atari и Nintendo, игры постепенно увеличивали свою сложность и детализацию. Теперь, с наступлением эры интернета, можно сражаться с драконами и другими игроками в реальном времени, набирая себе союзников по всему миру.

Привлекательность подобных игр для меня не в последнюю очередь обусловлена тем, что хорошая игра требует умения. Каждый может создать персонажа, но для того, чтобы вступить в сражение и победить, нужно знать, что делаешь.

Мы уже далеко ушли от Марио, который прыгал на врагов и иногда бросал огненный шар. Теперь считается обычным делом, если у персонажа видеоигры около сотни потенциальных умений. Чтобы стать мастером игры, вы должны знать эти умения, понимать, когда их использовать, а также настраивать своего персонажа, чтобы получить максимальный эффект в той или иной ситуации.

Тут нужно многому учиться. Какие умения являются самым грозным оружием, а какие лучше всего защищают? На что способны вражеские чудовища? Какова наилучшая стратегия при сражении с другими игроками?

Чем больше я играю, тем сильнее становлюсь. Вот почему мне это нравится.

С исторической точки зрения, видеоигры очень молоды. Однако игры, в которых человек может рассчитывать на удачу и мастерство, существуют уже не одну тысячу лет.

Самая древняя в мире стратегическая настольная игра

Не так давно я наткнулся на интересную игру. По сравнению с тем, во что

я привык играть, она кажется тихой, даже безмятежной. Однако за этим спокойствием скрывается полная напряжения драма. Доска — это поле великой битвы, а игроки — генералы, сражающиеся друг с другом не на жизнь, а на смерть.

В го, самую древнюю игру в мире, играют точно так же, как и тысячи лет назад. Согласно легендам, игра го возникла в Древнем Китае, и ее правила остаются неизменными уже не менее трех — некоторые историки считают, что и четырех, — тысяч лет. Возраст явно указывает на качество, а значит, го достойна внимания.

В Китае эта игра носит название *вэйци*. *Вэй* (圍) означает «окружающий», а *ци* (棋) — настольная игра. Два этих иероглифа образуют слово *вэйци* (圍棋), или упрощенно 圍棋, что означает «игра в окружение».

Это простое и точное описание цели игры: окружение противника.

Игра го попала в Европу и Америку из Японии, и наше название «го» — это упрощенное японское название игры, *и-го* (囲碁). Как бы вы ее ни называли — го, вэйци, и-го, байдук, — это одна и та же игра.

Искусство войны

На Западе самой популярной стратегической настольной игрой являются шахматы, поэтому мы можем взять их в качестве примера для сравнения. На первый взгляд, эти игры очень похожи.

В каждой игре участвуют два игрока, за черных и за белых. Поле действия — квадратная доска. Белые начинают. Игроки по очереди делают ходы, вплоть до окончания партии. Игру можно рассматривать как военное столкновение, где игроки — противостоящие друг другу генералы.

Но на этом сходство заканчивается.

В шахматы играют на доске размером 8 на 8 клеток — всего 64. Каждая шахматная фигура занимает одну клетку. Доска напоминает поле битвы, а фигуры — солдат.

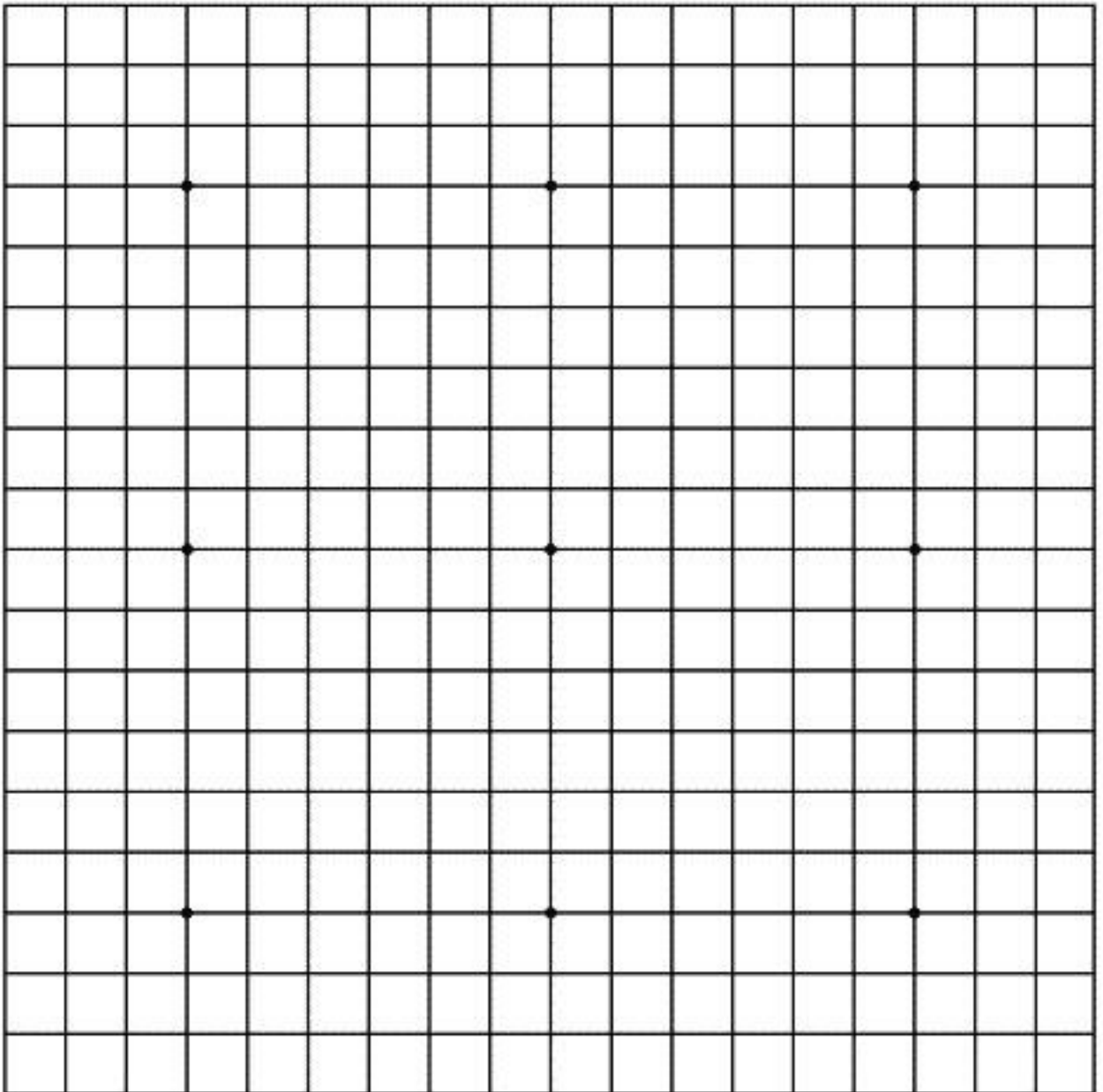
Вот как выглядит шахматная доска перед началом игры.



В го играют на доске, расчерченной 19 вертикальными и 19 горизонтальными линиями. Камни ставятся на пересечения линий, а не на клетки. В результате на доске для игры в го имеются 360 пересечений для камней — в 5,625 раза больше, чем клеток на шахматной доске.

Вот как выглядит доска для игры в го перед началом игры:

GO



Заметили разницу? Доска (гобан) для го значительно больше, а игра начинается с пустого поля. В го камни обычно *добавляются* в процессе игры. В шахматах сбитые фигуры *снимаются* с доски.

Обратите внимание, что маленькие черные точки на доске (их называют «узловыми точками») расположены абсолютно симметрично. Эти точки очень

важны, и мы к ним вернемся чуть позже.

В шахматах насчитывается шесть типов фигур, каждая со своими правилами поведения и возможностями. *Пешки* могут двигаться только вперед, за исключением случаев, когда они берут другую фигуру. *Слоны* могут передвигаться по диагонали, но не по вертикали или горизонтали. *Ладьи*, наоборот, передвигаются по вертикали и горизонтали, но не по диагонали. *Конь* может перепрыгивать через другие фигуры, но сдвигается лишь на две клетки по вертикали и одну по горизонтали или на две по горизонтали и одну по вертикали. *Ферзь* может перемещаться по вертикали, горизонтали и диагонали, но не имеет возможности перепрыгивать через фигуры. *Король* ходит во всех направлениях, но только на одну клетку — за исключением особых случаев, называемых рокировкой, когда он перепрыгивает через ладью. Все это нужно запомнить.

В отличие от шахмат, в го все ходы одинаковы: камень ставится на пересечение линий. Каждый выставленный камень не перемещается, если только впоследствии он не «захвачен» противником.

Шахматные фигуры «берутся» фигурами противника, как в поединке на поле брани. Ладья булавой ударяет слона по голове, и тот выходит из игры.

Камни го захватываются в том случае, если они со всех сторон окружены камнями противника. Когда армия «черных» со всех сторон окружена «белыми», камни «черных» сдаются и попадают в плен.

Профессиональные шахматные партии обычно насчитывают от 30 до 40 ходов. В го 30 ходов считаются началом игры: в завершающую фазу партия переходит после 100 ходов. Обычно для завершения партии в го требуется больше 250 ходов.

Со всех точек зрения масштаб го значительно больше. Если шахматную партию сравнить со сражением, то партия в го — это военная кампания.

Как же играют в го? Попробуем разобраться.

Правила игры

Хотите верьте, хотите нет, но в го всего семь главных правил, и с двумя из них мы уже знакомы.

1. Камни ставятся на пересечении линий.

2. Черные и белые по очереди выкладывают камни на гобан.

Следующие пять правил определяют ход игры и условия выигрыша.

3. Камни «захватываются», когда они со всех сторон окружены камнями противника.

4. Запрещено ставить камень, который немедленно захватывается («самоубийство»).

5. Запрещено бесконечное повторение последовательности ходов (так называемое правило *ко*).

6. Игра заканчивается, когда у игроков заканчиваются камни, один игрок сдается или оба игрока пасуют.

7. Побеждает игрок, к концу игры окружающий бо2льшую часть доски.

На соревнованиях обычно добавляются еще несколько правил, устраняющих неопределенности (например, относительно повторяющихся ходов), уточняющих систему подсчета очков и предотвращающих ничью. Вот и все.

Очень просто, правда?

Выучить правила игры в го легко: это займет несколько минут. Сами правила тоже несложные.

Существует очень старая поговорка: «Чтобы научиться играть в го, требуется пять минут; чтобы научиться побеждать — вся жизнь». Соедините эти простые правила с очень простой доской и простыми камешками, и получится невообразимая сложность.

Размеры Вселенной

В то время как причудливые правила шахмат могли быть созданы только людьми, правила го настолько элегантны, органичны и строго логичны, что если где-то во Вселенной существуют иные разумные формы жизни, они несомненно играют в го.

Эдуард Ласкер, шахматный мастер, автор книги «Го и гомоку» («Go and Go toki»)

Давайте представим, что мы хотим создать компьютерную программу, которая хорошо играет в го, — нечто подобное знаменитому искусственному интеллекту Deep Blue, который в 1996 году обыграл в шахматы действующего чемпиона мира Гарри Каспарова.

Обычно компьютеры превосходят человека благодаря вычислительной мощности: они просчитывают все возможные ходы на доске, затем выбирают ход с наибольшей математической вероятностью успеха, которая основана на анализе базы данных предыдущих партий.

На шахматной доске такие вычисления произвести очень трудно, но возможно. Клеток всего 64, а движение каждой фигуры подчиняется строгим правилам. Поскольку каждая фигура может перемещаться лишь определенным образом, программе нужно учитывать относительно небольшое количество вариантов.

В игре го игрок, чья очередь делать ход, может поместить камень на *любое* свободное пересечение на доске. Игра начинается с выбора одного из 360 вариантов, и поэтому с самого начала нашему неоперившемуся искусственному интеллекту предстоит гораздо больше работы.

Давайте произведем кое-какие подсчеты. Сколько последовательностей из пяти ходов можно разыграть на доске для го, предполагая, что игра только начинается и ни один из игроков не успел захватить камни противника?

Вот что у нас получится:

$$360 \times 359 \times 358 \times 357 \times 356 = 5\,880\,282\,488\,640$$

То есть более 5,8 триллиона возможных последовательностей — и это лишь первые пять ходов!

При попытке продолжить вычисления цифры очень быстро становятся просто безумными. Помните, я говорил, что средняя продолжительность партии го составляет 250 ходов? Исходя из этого предположения, на доске го размером 19 на 19 линий в такой партии возможно $2,08 \times 10^{170}$ последовательностей ходов.

Если эти подсчеты правильны, то допустимых ходов в партии го больше, чем *элементарных частиц в известной людям Вселенной*.

Математически достоверно доказано, что каждая когда-либо сыгранная партия в го никогда не была сыграна раньше за всю историю Вселенной, даже если вы допускаете возможность, что где-то существуют миллионы высокоразвитых цивилизаций, которые тоже играют в го [2].

При современном уровне техники самым мощным компьютерам, работающим по методу перебора, потребуется около 400 лет, чтобы вычислить один оптимальный ход — предполагая, что программа тратит на один вариант несколько миллисекунд.

Невероятно. Эта игра *необъятна*.

Как же люди (и компьютеры) играют в го?

Если бы при игре в го противники опирались на логический анализ, они сошли бы с ума. Совершенно очевидно, что этого не происходит: однако опытные игроки способны определить наилучший ход за несколько секунд. Как им это удастся?

Для определения сильных ходов игроки в го опираются на распознавание образов. Они употребляют разные слова, которые обозначают интуицию, — например, «форма» или *сэнте* (инициатива). Похоже, лучшие игроки в го в своих решениях опираются не только на строгую логику и анализ, но также на

геометрию, красоту и эмоции.

Это не лишено смысла: человеческий мозг не очень приспособлен для «лобовых» вычислений, но не имеет себе равных в распознавании образов. Рассматривая узор из камней на доске, опытные игроки могут оценить позицию и найти наилучший ход гораздо быстрее, чем за четыреста лет.

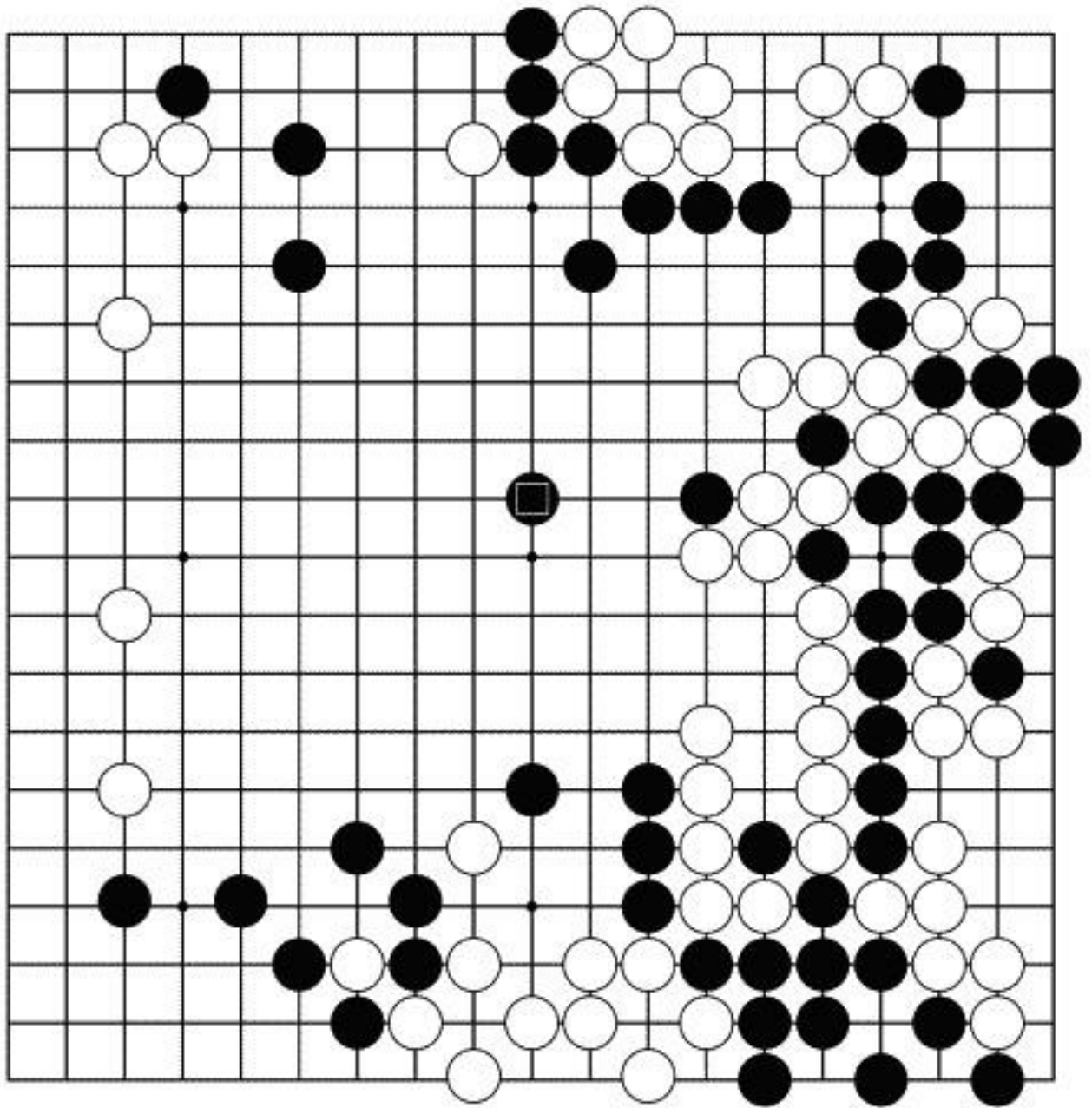
И еще более невероятное: самые лучшие игроки могут предсказать расположение камней в будущем, через 30 или даже 40 ходов. Если вы попробуете сыграть с профессионалом, у вас неизбежно возникнет ощущение, что он читает ваши мысли.

Реальная партия

Для непосвященного разница между камнем, поставленным на пересечение линий, и соседним незначительна. Однако для мастера го это разница между цветком и шлакобетонным блоком.

Дэйв Лоури, автор книги «Вызов го: эзотерический дедушка настольных игр»

Вот как выглядит позиция на доске для игры в го.



Это диаграмма реальной партии, одной из самых знаменитых в истории го [3].

В 1846 году Сюсаку из дома Хонинбо, 17-летний мастер 4-го дана, получил приглашение сыграть с Геннаном Инсэки, имевшим 8-й дан. Инсэки был главой дома Иноуэ, одной из четырех главных профессиональных школ игры в го в Японии в середине XIX века.

Сюсаку принял лестное для себя предложение. Матч собрал большое количество зрителей. Никто не предполагал победы Сюсаку, однако он был

сильным игроком и мог оказать серьезное сопротивление мастеру.

После короткой партии, в которой Инсэки дал Сюсаку фору в два камня, стало ясно, что молодому человеку фора не нужна. Вторую партию Инсэки согласился играть на равных.

Сюсаку, как бросивший вызов мастеру, играл черными. Начало партии было безупречным, за исключением маленькой ошибки Сюсаку во время обмена в правой нижней части доски. На протяжении 126 ходов, как и ожидалось, в счете вел Инсэки. Естественно, кто мог превзойти признанного мастера?

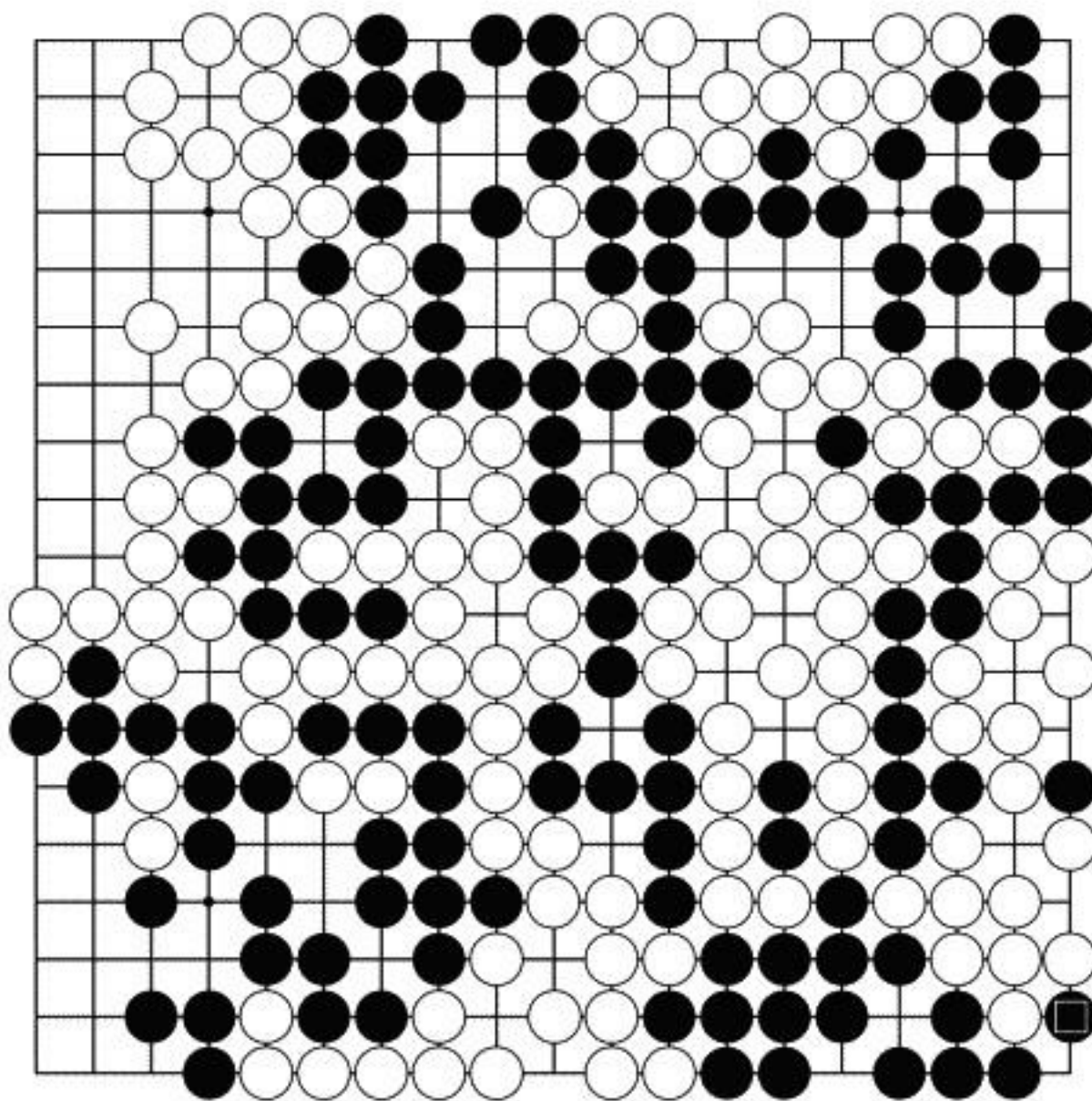
Но следующий ход Сюсаку изменил ход партии. Вы сами можете увидеть этот камень: чуть выше центра доски, отмеченный квадратиком.

После хода Сюсаку один из зрителей заметил нечто странное: уши Инсэки покраснели. Мастер волновался.

127-й ход Сюсаку был хорош — очень, очень хорош. Один камень в центре доски оказывал поддержку камням Сюсаку одновременно в верхней, правой и нижней частях доски, а также создавал новую угрозу слева. Этот ход был идеальным сочетанием защиты и атаки.

Инсэки попал в трудное положение и понимал это. Одним камнем Сюсаку оказывал влияние на всю доску. Игра продолжилась, и теперь уже догонять противника приходилось Инсэки.

Вот как выглядела позиция после 325-го хода.



Внимательно посмотрите на доску. Какие области окружены черными, а какие белыми? Кто захватил больше территории на гобане?

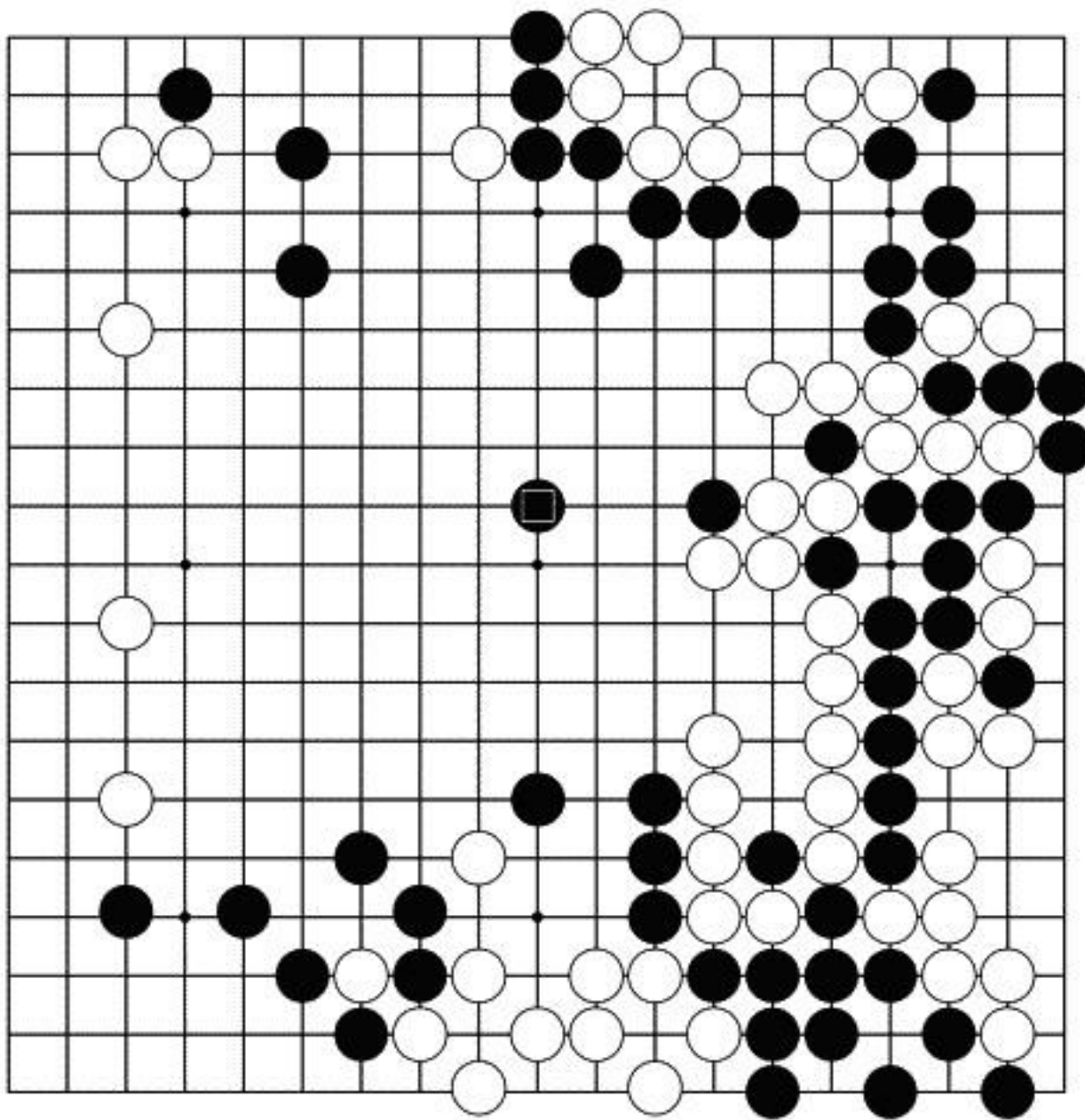
Инсэки признал поражение. Сюсаку выиграл с преимуществом в два очка. Игра вошла в историю как «партия красных ушей», а 127-й ход — как «ход красных ушей».

«Непобедимый Сюсаку» стал одним из самых знаменитых игроков в истории, победителем девятнадцати замковых игр подряд, которые ежегодно устраивались сегуном. Сюсаку умер 7 сентября 1862 года в возрасте 33 лет; он

заразился холерой, когда ухаживал за больными.

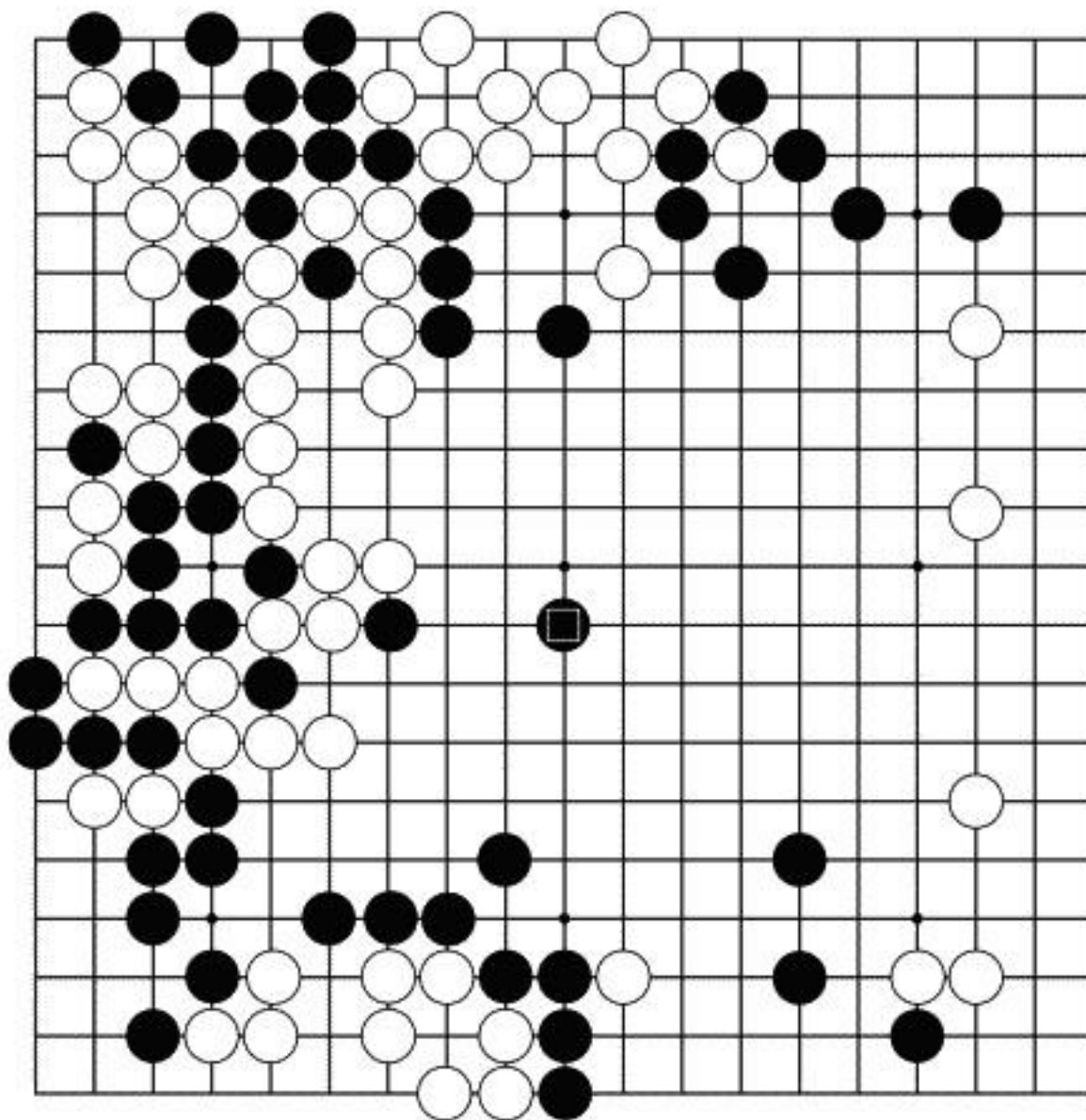
Распознавание позиций

Эти диаграммы могут нас кое-чему научить. Давайте еще раз посмотрим на «ход красных ушей».



На диаграммах позиции обычно представляют со стороны черных. Мы видим доску так, как видел ее Сюсаку во время партии.

А так видел доску Инсэки (белые), повернутую на 180 градусов (на следующей странице вверху).



В шахматах возможны только две перспективы, поскольку соперники начинают игру со всеми фигурами на соответствующих сторонах доски. Ни белые, ни черные фигуры изначально не расставляют на правой и левой стороне

поля. Кроме того, ходы в шахматах обычно имеют определенную направленность — *от* игрока *в направлении* противника.

В результате шахматисты имеют возможность запомнить конкретные позиции. Вот, например, знаменитый «королевский гамбит».



Для того чтобы стать гроссмейстером, нужно запомнить множество позиций, которые встречаются в шахматных партиях. Лучшие шахматисты

мгновенно оценивают развитие событий на доске, поскольку позиции остаются неизменными. Если вы попытаетесь разыграть королевский гамбит, опытный мастер сразу же увидит это и будет знать ответ.

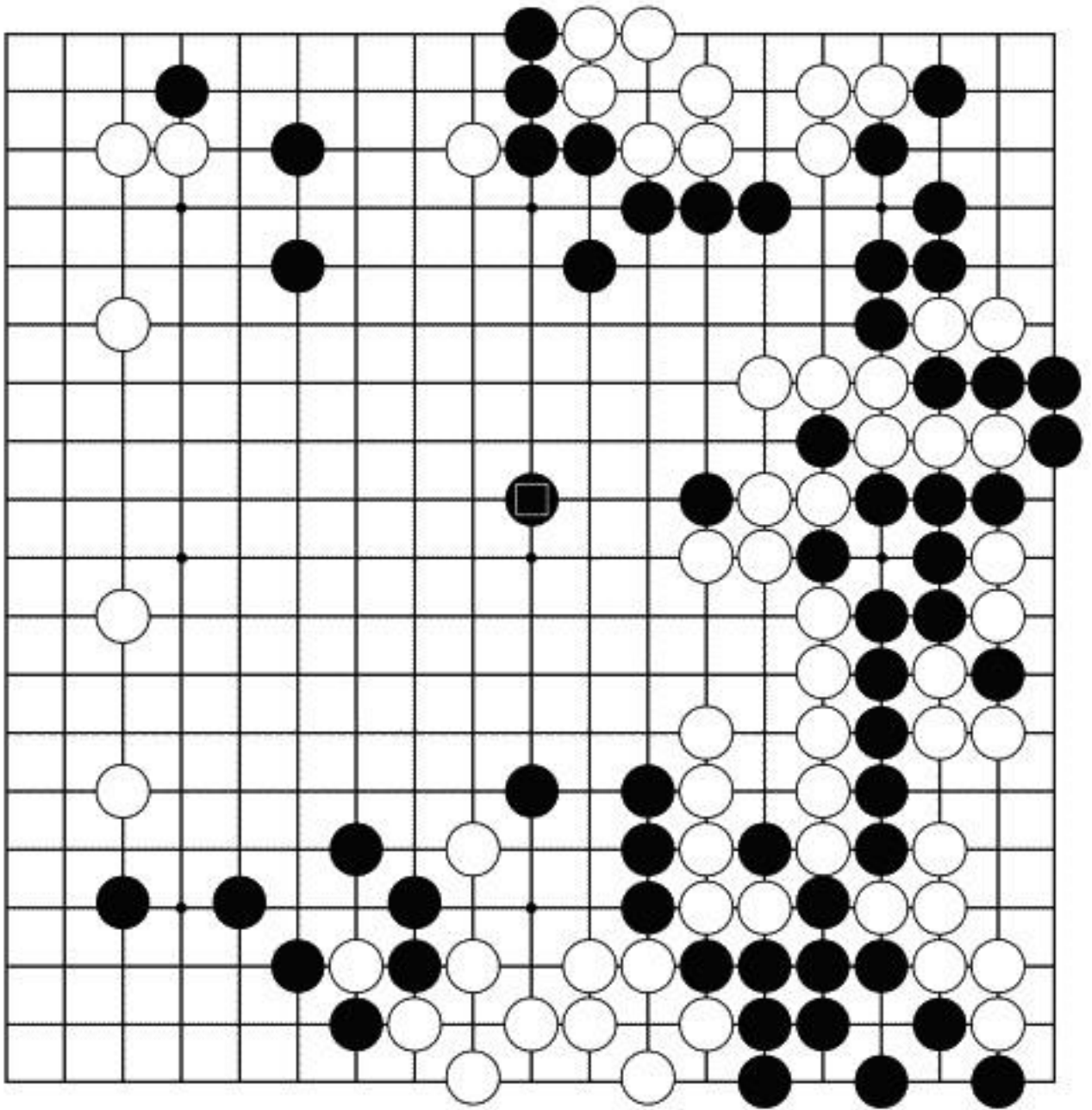
Распознавать позиции в шахматах непросто, но шахматисту помогает тот факт, что ему не приходится смотреть на доску, повернутую на 90 градусов вправо или влево, и позиции всегда выглядят одинаковыми.

В отличие от шахматной доски, гобан симметричен, и все его стороны равноправны. В результате вы можете смотреть на доску с любой из четырех ее сторон.

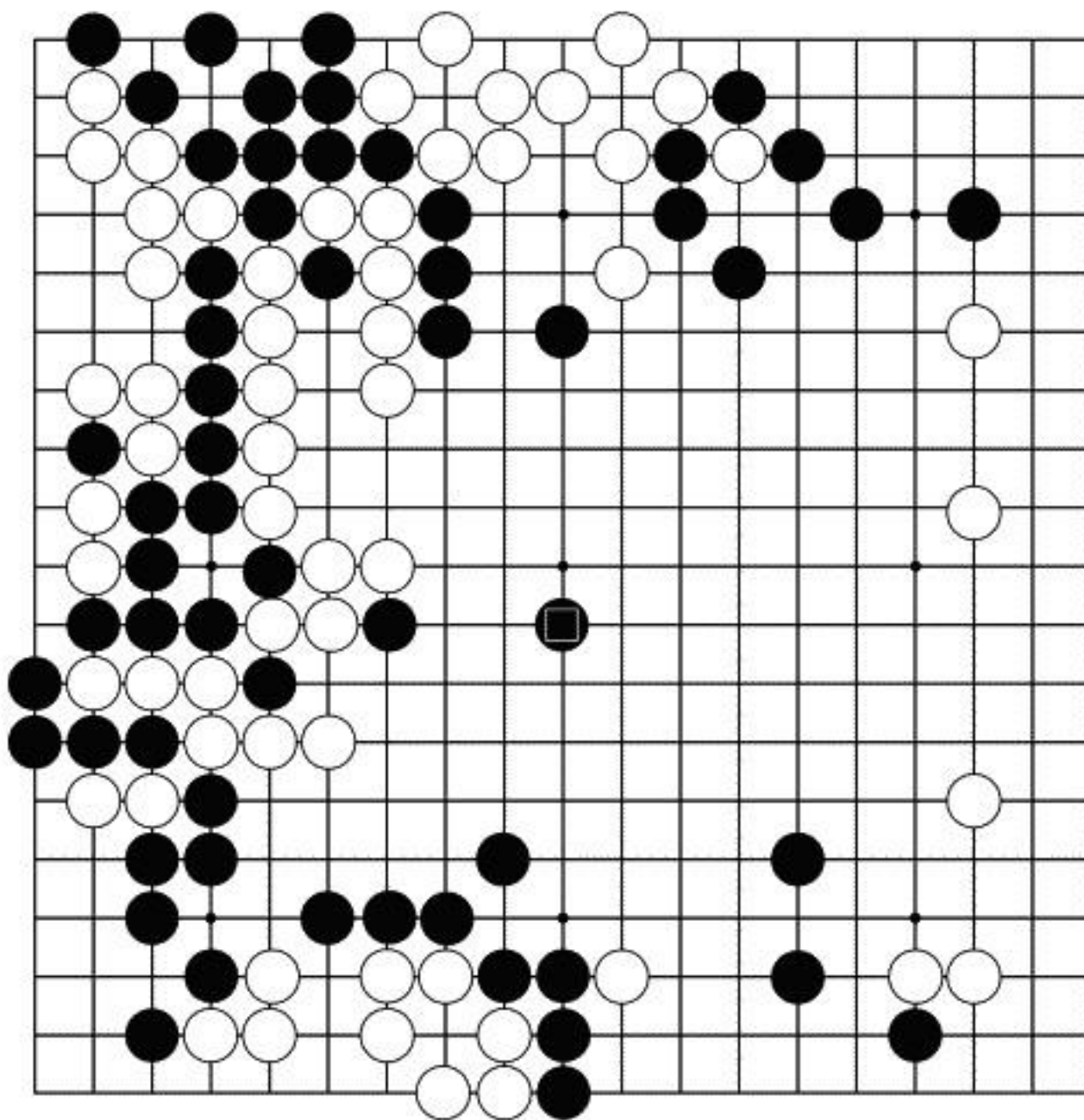
До начала игры форовые камни ставятся в «узловые точки» — симметрично расположенные маленькие черные точки на доске. Если форы нет, гобан остается пустым.

Камни го ставятся на любое свободное пересечение. Делая ход, игрок может смотреть на доску сверху, а может слева или справа. Здесь нет универсального, предсказуемого направления движения, как в шахматах.

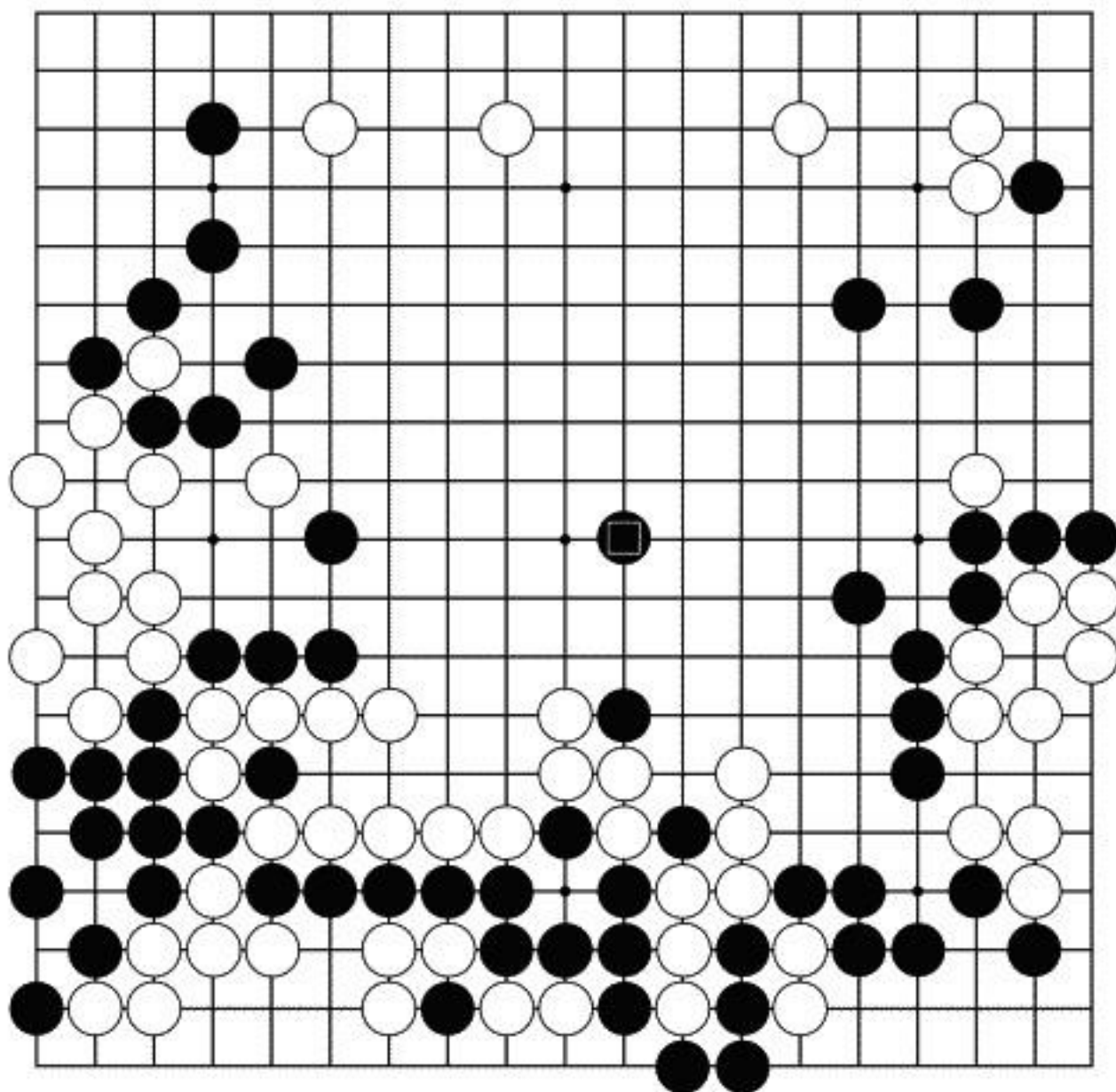
Давайте вернемся к «ходу красных ушей». Вот так выглядела доска со стороны Сюсаку.



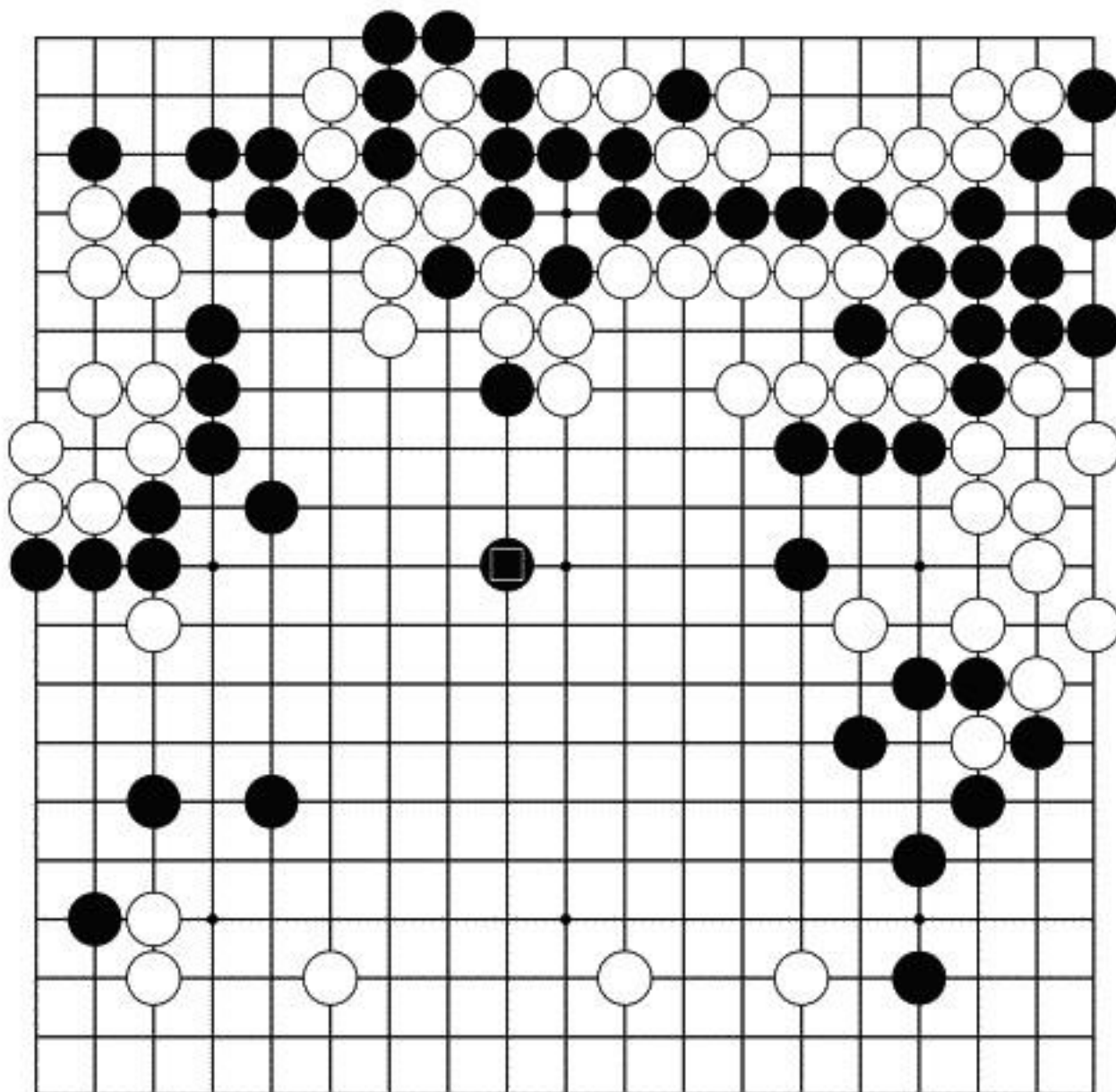
А вот так со стороны Исэки.



А если смотреть справа от Сюсаку и слева от Исэки, то вот так.



А так выглядела позиция на гобане, если смотреть слева от Сюсаку и справа от Исэки.



Симметрия гобана еще больше осложняет распознавание позиций. С точки зрения стратегии эти четыре диаграммы из «партии красных ушей» *абсолютно одинаковы*, хотя на взгляд неопытного игрока выглядят разными. При желании можно после каждого хода поворачивать гобан на 90 градусов, и это не повлияет на ход партии.

Игроки в го используют иные методы запоминания позиций, чем шахматисты, оценивающие развитие партии. Каждая позиция, анализируемая игроком в го, должна рассматриваться с *четырёх* сторон, а не с только одной. Поэтому в го такое значение придается интуиции. Игра слишком масштабна для

прямого запоминания или вычислений, и опытные игроки тренируются видеть развитие партии на более высоком уровне: общие формы, направление движения, ощущение силы или слабости.

Используйте свои чувства...

[Го требует] тактики воина, точности математика, воображения художника, вдохновения поэта, спокойствия философа, а также величайшего ума.

Чжан Юнки, китайский Институт вэйци, 1991

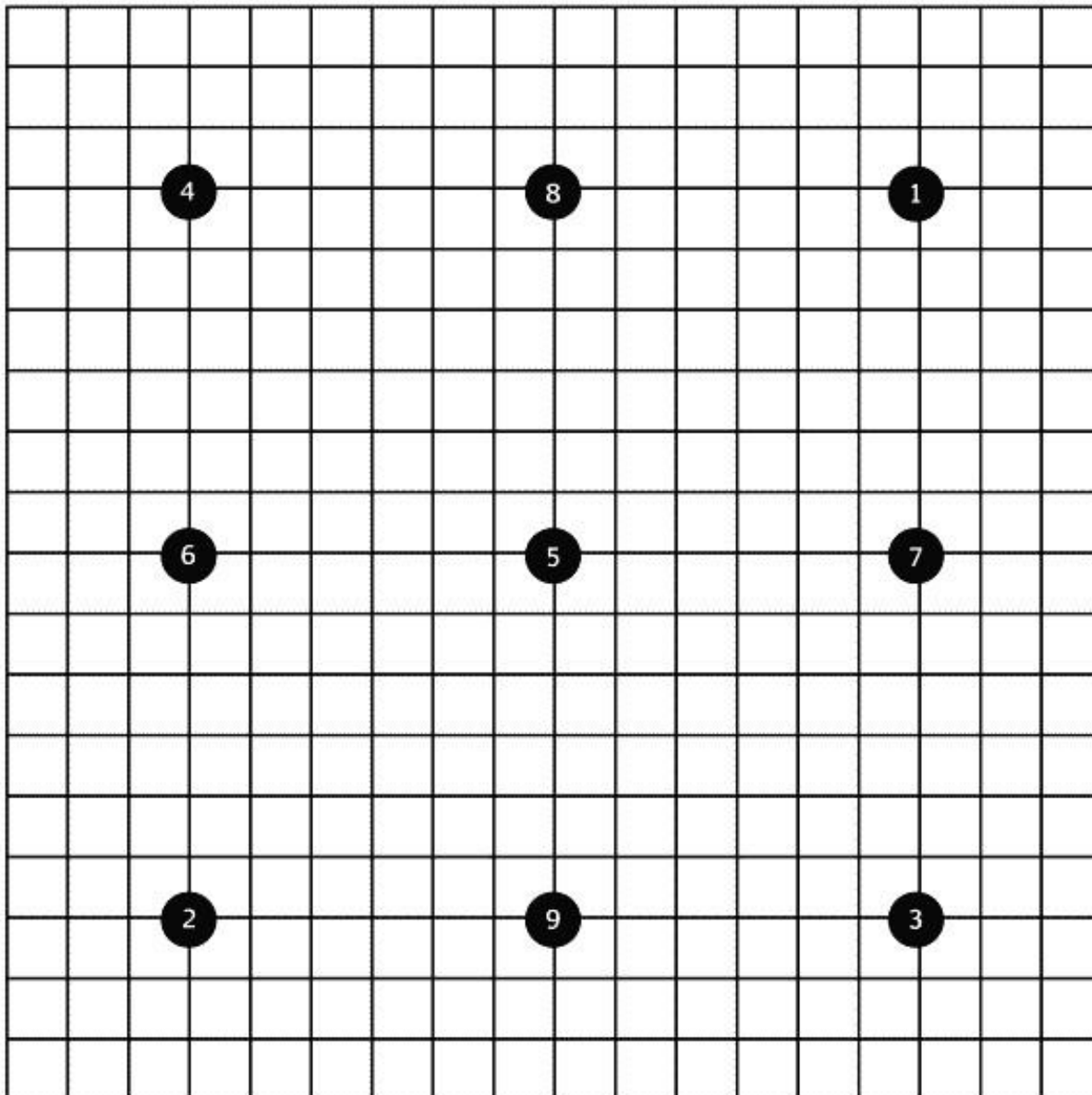
В одной из первых статей, прочитанных мной о го, содержалось любопытное утверждение: овладение этой игрой — не умение противостоять сопернику и побеждать его. Это совершенствование себя.

Азартные игры возникли еще на заре человеческой цивилизации. Когда вы бросаете кости, победителя определяет Госпожа Удача (а если точнее, то Госпожа Физика). Мастерство не имеет значения.

В отличие от игры в кости интеллектуальные игры обычно предполагают, что нужно перехитрить или переиграть других игроков. Кто проявит бо2льшую сообразительность и хладнокровие? Кто первым увидит свой шанс? Кто лучше использует слабости соперника? Именно к этой категории относятся шахматы: хороший игрок должен освоить тактику и уметь предсказывать действия противника.

Го в этом отношении уникальна: использование форы предусматривает возможность сделать сильных игроков слабее. Если противник начинает игру с несколькими камнями в главных стратегических пунктах доски, это существенно влияет на вашу стратегию.

Камни форы располагаются на следующих позициях.



При правильном распределении фору в матче между двумя приблизительно равными по силе игроками в го каждый из них выигрывает около половины партий. В случае поражения в следующей партии проигравший получает фору в один или два камня.

Из-за этой особенности го можно считать противоборством с самим собой. Конечно, вы принимаете решения в зависимости от действий противника, но победа в игре — не единственная цель.

По мере роста мастерства уменьшается количество камней фору, которые вам требуются. Вы уже на равных сражаетесь с сильными соперниками, а в конечном итоге начинаете сами давать фору более слабым.

История го на первый план выдвигает аспект самосовершенствования:

чтобы стать мастером, нужно совершенствовать свои мышление и чувства. Лучшие игроки видят в расположении камней признаки, указывающие на психологическое и эмоциональное состояние противника.

Согласно легендам, древние мастера могли прочесть запись партии и сказать, когда каждый из игроков испытывал такие чувства, как гнев, растерянность, зависть или жадность, а также точно определить момент, когда «входила служанка, чтобы подать чай».

Каменный ниндзя

За сотни лет правило форо трансформировалось в систему рангов: разница в ранге определяет количество форовых камней, которые слабейший игрок может выставить на доску перед началом игры.

Хотите верьте, хотите нет, но система «поясов» пришла в боевые искусства из го. С повышением мастерства растет ранг игрока.

Начинающие игроки получают ранг 35-го кю (нечто вроде белого пояса). По мере того как игрок становится сильнее, его ранг снижается, пока не достигает 1-го кю.

Следующий после 1-го кю ранг — 1-й дан (эквивалент черного пояса). Далее ранг повышается, пока игрок не получает самый высокий, 9-й дан. 10-й дан — это почетный титул, присваиваемый победителям самых престижных международных турниров.

Определить ранг игрока можно несколькими способами. Первый способ — это соревнования. Если игрок выигрывает большинство партий у соперников с рангом 12-й кю, но проигрывает большинство партий соперникам с рангом 8-й кю, значит, его сила приблизительно 10-й кю. Чем больше партий сыграно в квалификационных турнирах, тем точнее определяется ранг.

Второй способ определения ранга — решение задач го. Это структурированные головоломки, представляющие собой позицию, в которой игрок должен найти наилучший ход для достижения определенной цели, например «захвата группы черных» или «спасения белых». При повышении общего ранга игрока усиливается его способность решать подобные задачи.

За много веков существования го накопилась огромная библиотека задач. Эти задачи отличаются по сложности: игроку с рангом 20-го кю задачи ранга 10-го кю покажутся трудными, тогда как игрок 1-го дана посчитает их легкими, а решения — очевидными. Издательство Kiseido, специализирующееся на выпуске книг о го, публикует очень удобную серию ранжированных задач, которая так и называется — «Ранжированные задачи го для начинающих» («Graded Go Problems for Beginners») и очень полезна для этой цели.

Существование ранжированных задач превращает го в игру, которую

можно изучать. Решение задач из этих книг — превосходный способ самостоятельных занятий, а также выяснения своей силы как игрока.

Итак, я готов приступить к занятиям. Что мне для этого нужно?

Подготовка

Сложно играть в го, не имея доски и камней, и поэтому я приобрел красивый комплект от Yellow Mountain Imports [4], американской компании, которая предлагает принадлежности для игры в го со всего мира.

Доски и камни могут значительно различаться по цене и качеству. Дешевые доски и простые стеклянные камни обойдутся в несколько долларов, так что базовый набор для игры приобрести несложно.

На другом конце ценовой шкалы располагаются высококачественные доски из дерева *кайя*, традиционного материала, используемого в Японии. Такие доски могут стоить несколько сотен или даже тысяч долларов. Вы можете купить камни из натурального сланца и раковин, но за них тоже придется выложить кругленькую сумму.

После недолгих поисков я остановился на симпатичной доске *шин-кайя* (имитация *кайя*, обычно из канадской ели) и комплекте камней Yunzi производства китайской компании с таким же названием. Состав своих камней производитель держит в секрете: камни красивые, плотные на ощупь, издают приятный звук, когда вы ставите их на доску, и имеют разумную цену. К камням прилагаются две традиционные деревянные чаши, которые ставятся рядом с гобаном во время игры.

Кроме гобана и камней, я купил несколько книг по игре го для начинающих, в том числе:

- «Го: подробное введение в игру» («Go: A Complete Introduction to the Game»), Чо Чикун (2010);
- «Вторая книга го: что нужно знать после того, как вы выучили правила» («The Second Book of Go: What You Need to Know After You've Learned the Rules»), Ричард Бозулич (1998);
- «Как не нужно играть в го» («How Not to Play Go»), Юань Джоу (2009)»;
- «Лекции по основам го» («Lessons in the Fundamentals of Go»), Тосиро Кагэяма (1996);
- «Теория фусэки — это просто» («Opening Theory Made Easy»), Отакэ Хидео (1992).

Эти чрезвычайно полезные книги я нашел при помощи сайта Sensei's Library [5], который предлагает сотни страниц информации и комментариев об

истории и технике го. Одна из страниц — это список лучших книг о го, составленный сильнейшими игроками, настоящее сокровище на первом этапе обучения. Я изучил весь список, в котором больше 100 книг.

Теперь предстояло выбрать, какие книги читать первыми. Пособие «Го: подробное введение в игру» занимало первое место в списке для начинающих, так что думать тут было не о чем. Аналогичным образом, «Вторая книга го» рекомендовалась для изучения основ стратегии. Книга предполагала, что вы уже знаете правила, и в ней больше времени уделялось основным приемам.

«Как не нужно играть в го» — это пример инверсии, которому я очень обрадовался. Изучая распространенные ошибки, можно много узнать о том, какие действия ведут к успеху. Для большинства навыков инверсию приходится изучать самостоятельно, но в данном случае в моем распоряжении есть целая книга, написанная специалистом. Отлично!

«Лекции по основам го», которую еще часто называют «Желтой книгой го», лучшие игроки называют единственным текстом, который помог им значительно улучшить результаты. Похоже, книга слишком сложна для начинающего, и я решил отложить ее на потом.

И наконец, я выбрал книгу «Теория фусэки — это просто» потому, что даже элементарный анализ показывает, какую важную роль играет начало игры (фусэки). Поскольку партия начинается с пустого — если не считать форовых камней — гобана, то первые 30 или 40 камней образуют построение, определяющее дальнейший ход сражения. Если вы не знаете, как правильно начинать партию, а противник знает, то вы скорее всего проиграете. Именно поэтому начинать стоит с изучения фусэки.

За белым кроликом

Помимо всего прочего, немного облегчили изучение го современные технологии. Приложение SmartGo, доступное на iPhone и iPad, содержит программу искусственного интеллекта, базу ранжированных задач го, а также аннотации к известным играм.

SmartGo разработана для сенсорных экранов, так что вы можете ставить камень на «доску» прикосновением к соответствующему пересечению линий. Это значительно облегчает решение задач. Вместо того чтобы представлять правильный ответ, а затем заглядывать в книгу, проверяя себя, вы получаете мгновенную обратную связь. Эта быстрая петля обратной связи существенно облегчает работу над задачами, особенно многоходовыми.

Итак, я готов. У меня есть все необходимое для обучения игре. Осталось лишь одно... сделать так, чтобы я тратил свободное время на игру в го, а не во что-то другое.

Устранение препятствий

Го не обладает такой притягательностью, как видеоигра в стиле «экшн». Обучение игре в го требует времени и сосредоточенности. Я уже играю в другие игры, но у меня очень мало свободного времени. Если я хочу максимально быстрого прогресса в изучении го, то должен сконцентрироваться.

Это значит, что мне нужно устранить потенциальные препятствия. Самая главная и непосредственная угроза — другие игры. Время, потраченное на сражение с цифровыми монстрами, будет отнято у обучения го. Если я продолжу играть в видеоигры, то у меня не останется времени для занятий.

Помните, что я говорил? Время нельзя найти, его можно только *создать*.

Поэтому я решил не играть в другие игры, пока не потрачу как минимум 20 часов на изучение го. Если я хочу учиться быстро, то не могу позволить себе отвлекаться.

Для таких ситуаций существует надежная и проверенная тактика. Лучший способ изменить свое поведение — изменить структуру ближайшего окружения. Если вы хотите отказаться от какого-то занятия, сделайте его невозможным. А если не получится, то максимально трудным, дорогостоящим и неудобным. Чем больше препятствий придется преодолевать, тем меньше вероятность, что вы вернетесь к прежним привычкам.

Прощай, World of Warcraft... мы здорово повеселились.

Прежде чем пускаться в путешествие по миру го, я удалил свою учетную запись World of Warcraft и стер игру с компьютера. Если игра не установлена, то я не смогу в нее играть, даже если очень захочу. Я не буду ни при каких обстоятельствах играть в видеоигры, пока официально не завершится мой эксперимент с го.

Правила игры, повторное ознакомление

Лучше всего начинать с правил. Я уже их читал, но мне нужно увидеть, как это выглядит на гобане.

У го есть одно полезное свойство, облегчающее обучение: можно менять размер доски. Поскольку гобан симметричен, его можно делать больше или меньше, выбирая любое число линий. При этом правила и тактика не меняются.

Доски, используемые на соревнованиях, имеют 19 вертикальных и 19 горизонтальных линий, но для обучения лучше использовать размер 7 на 7. Этого достаточно, чтобы усвоить основные принципы игры. Доска таких размеров не вызывает у новичка чувство растерянности.

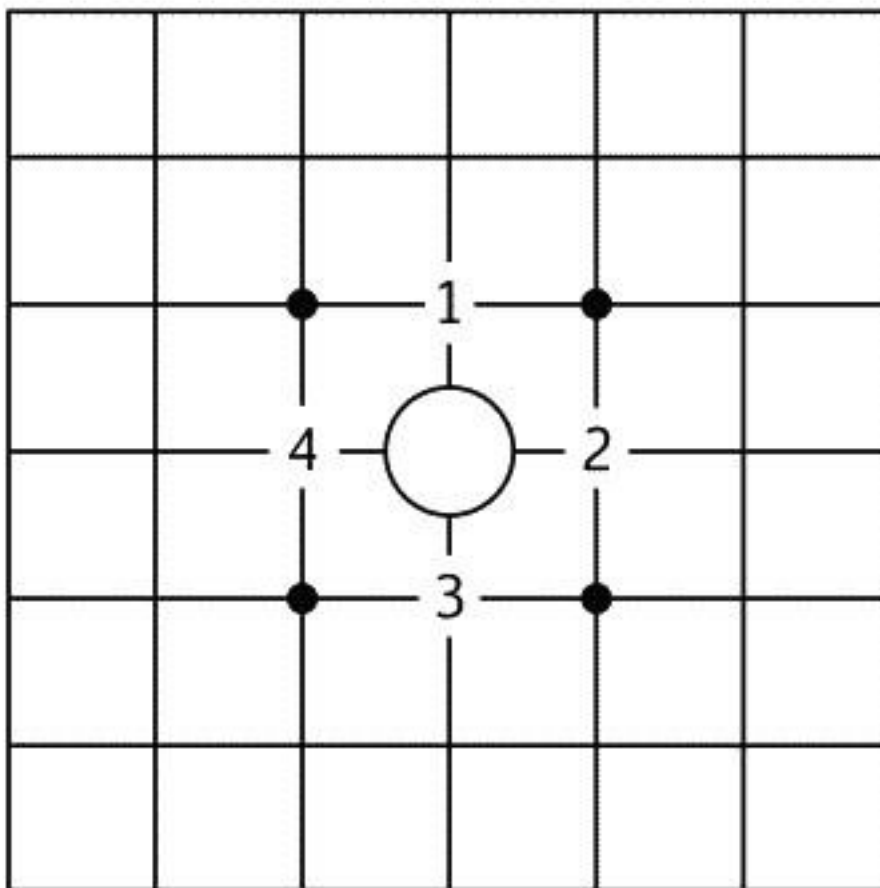
Давайте посмотрим, как применять правила в ходе игры.

Свобода или смерть

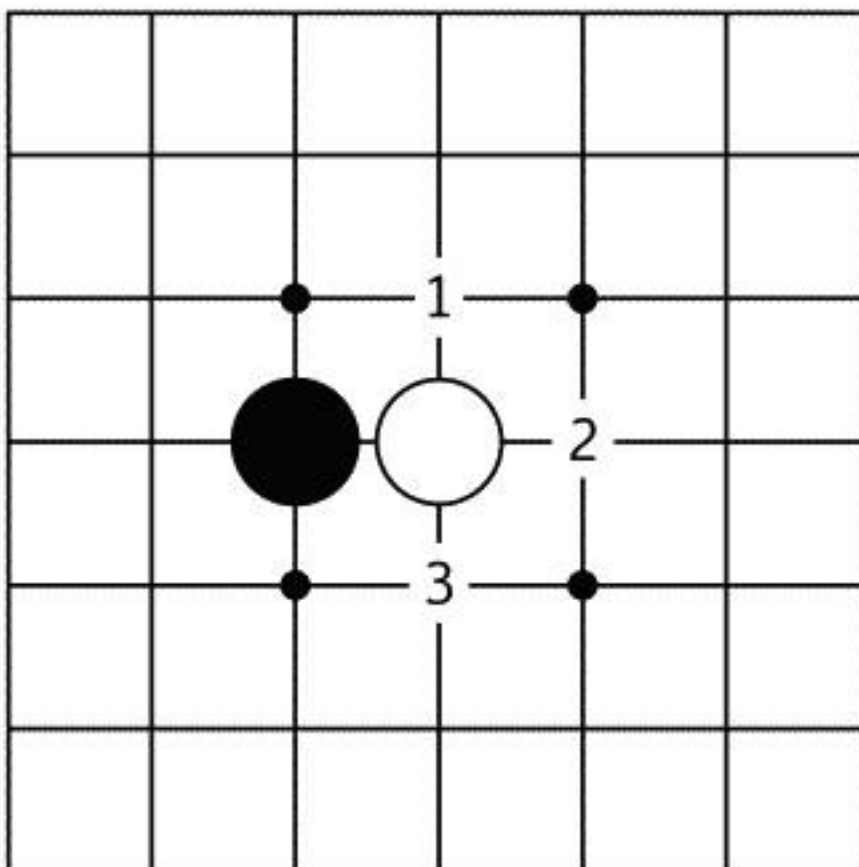
Начнем с третьего правила го: камни «захватываются», когда они со всех сторон окружены камнями противника.

Когда вы ставите камень на пересечение, то соседние пересечения получают название *степеней свободы*. Представьте себе солдата на поле боя: если рядом есть безопасное место для отступления, он не попадет в плен.

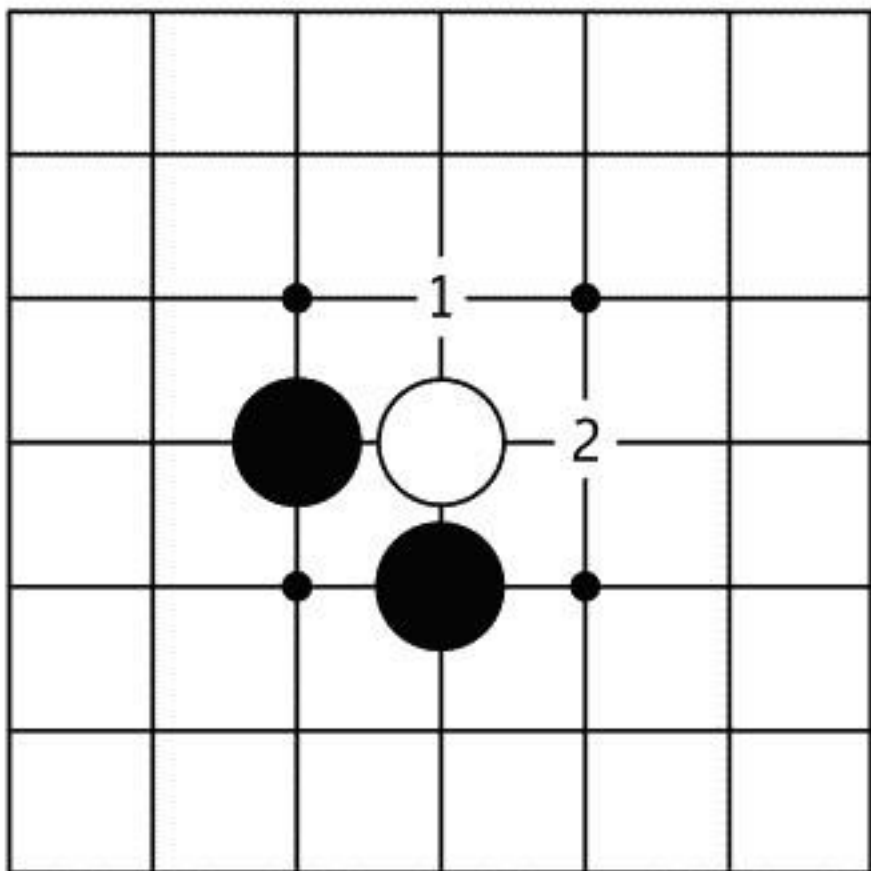
Максимальное число степеней свободы для любого камня — четыре.



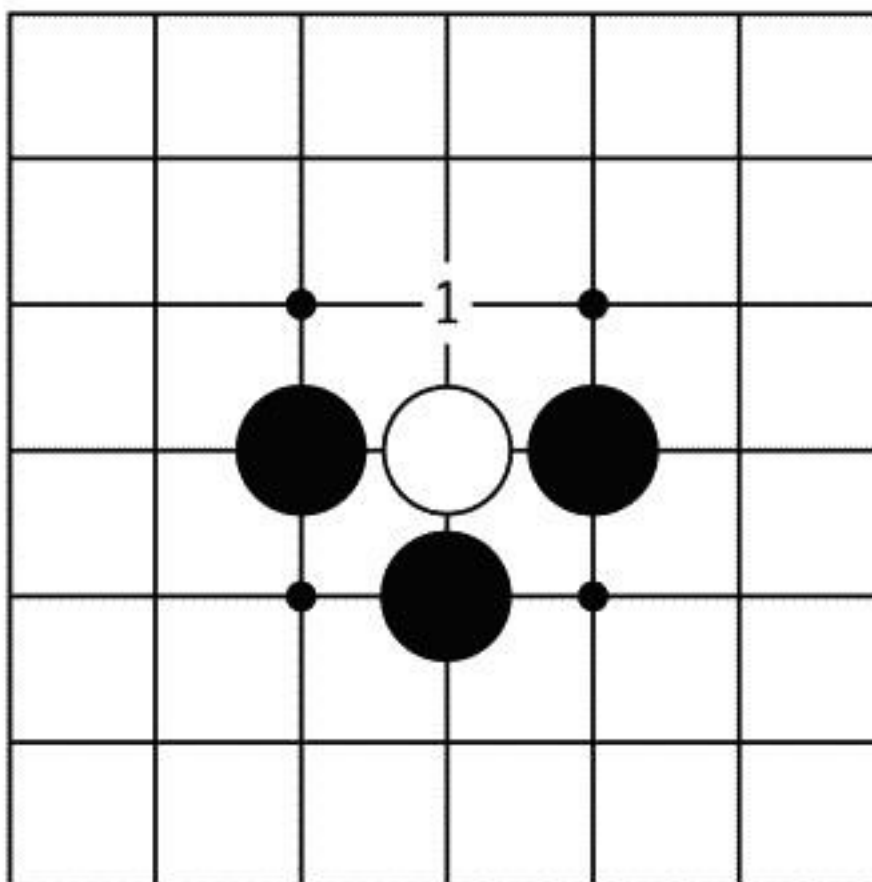
Число степеней свободы уменьшается, если противник атакует, ставя свой камень рядом с вашим. Эта атака уменьшает количество степеней свободы белого камня до трех.



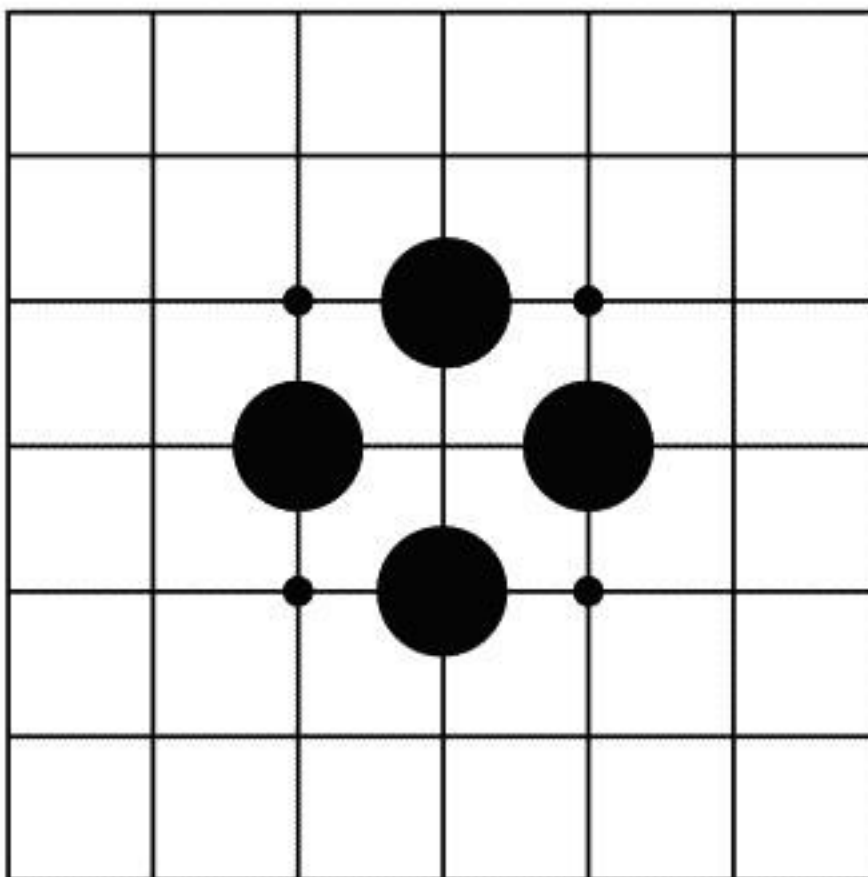
Следующая атака уменьшает число степеней свободы белого камня до двух.



Берегитесь, белые! Осталась одна степень свободы.



Такое положение называется *атари* [6]. Если черные уничтожат последнюю оставшуюся степень свободы белых, то белый камень попадет в плен и будет убран с доски.

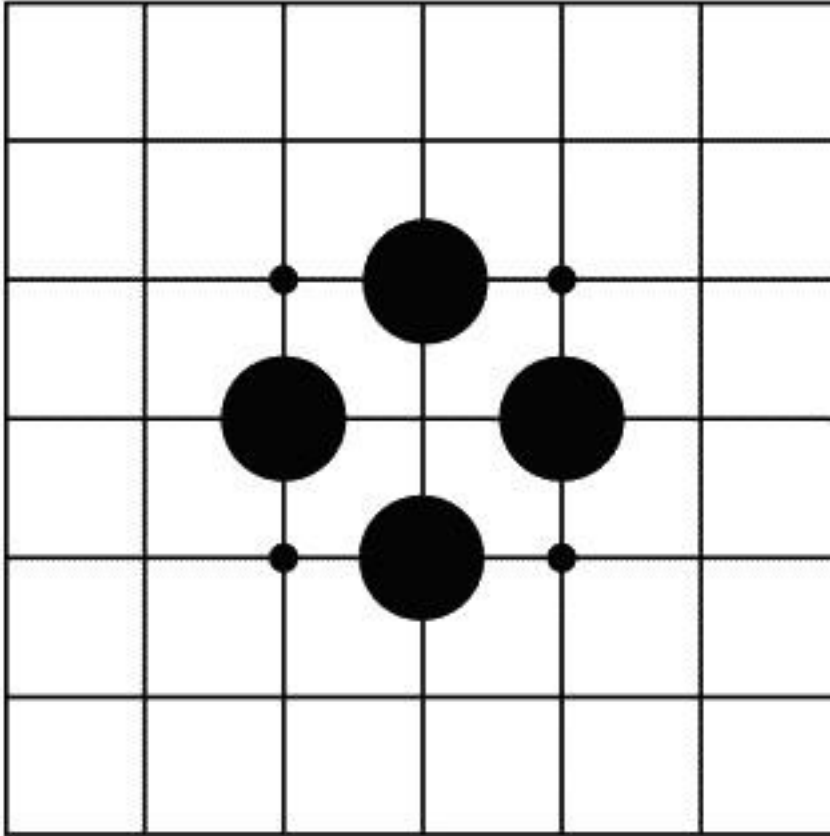


Пока все очень просто. Если вы не хотите, чтобы ваши камни захватили, то должны обеспечить им как можно больше степеней свободы. Если вы хотите захватить камни противника, ограничивайте их степени свободы.

Просьба: никаких самоубийств

Теперь пора перейти к четвертому правилу: запрещено ставить камень, который немедленно захватывается («самоубийство»).

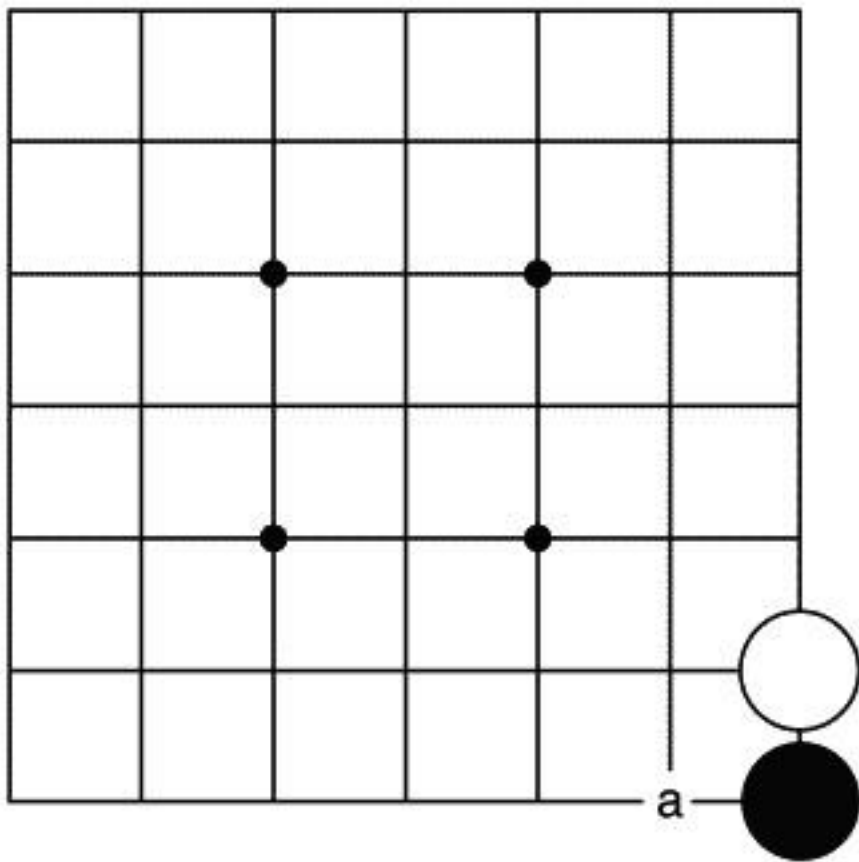
Взглянем еще раз на последнюю позицию.



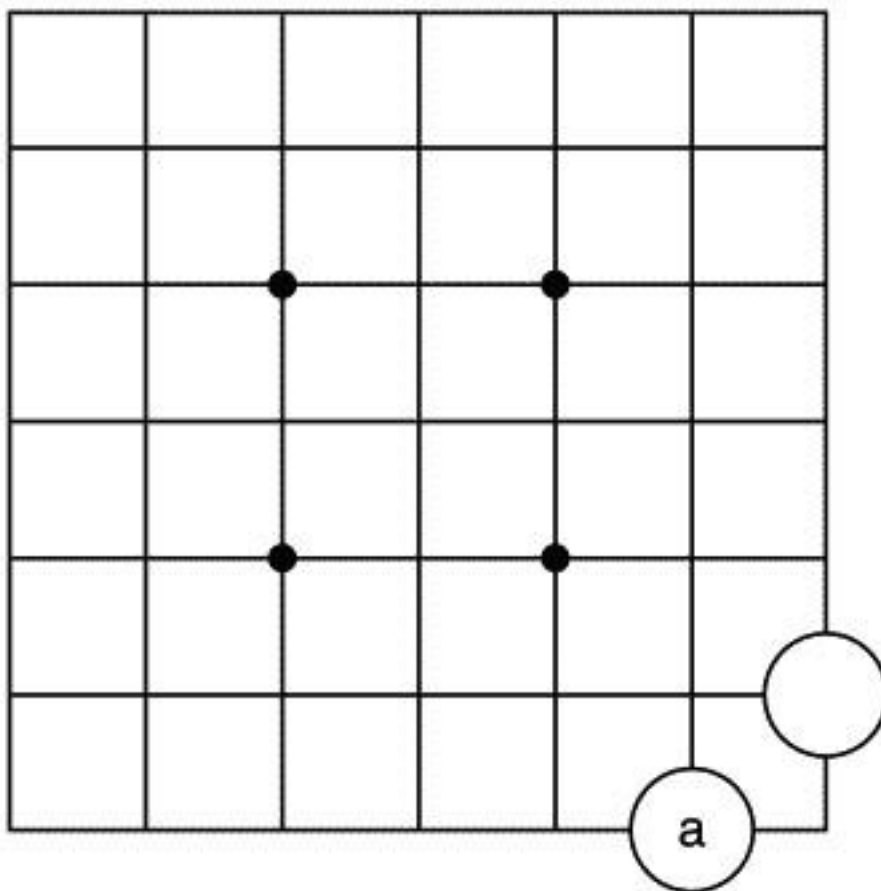
Эта форма называется *поннуки* и встречается в го довольно часто. У пересечения внутри черных камней нет степеней свободы, и если белые поставят сюда камень, то он будет немедленно захвачен. Четвертое правило запрещает такие «самоубийственные» ходы.

Загнать в угол

В центре гобана у камней, как правило, много степеней свободы, пока они не атакованы противником. По углам и у краев доски степеней свободы меньше, поскольку меньше соседних пересечений.



У черных положение *атари*. Если белые поставят камень на *a*, то черный камень будет захвачен (рисунок на следующей странице).



Этот пример иллюстрирует разницу между углами и центром доски. В углу для захвата нужны всего два камня, а в центре не менее четырех. По краям доски для захвата камня противника требуется три камня.

Следовательно, защищать территорию легче в углах. По краям сделать это немного сложнее, а в центре еще сложнее — для окружения противника требуется много камней.

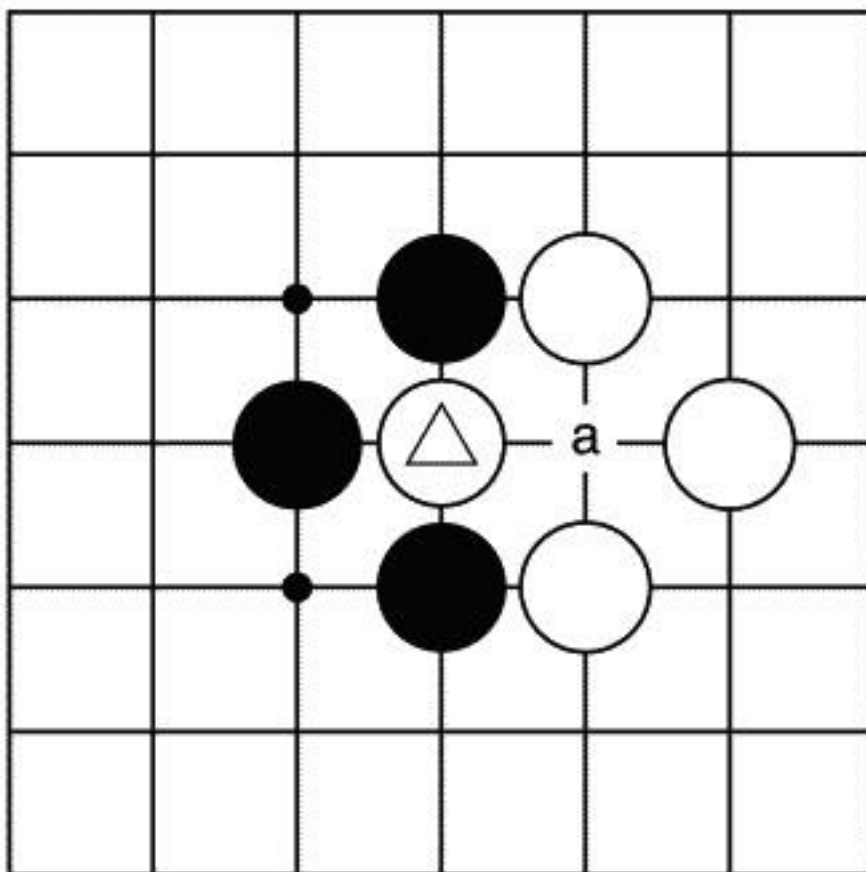
Вот почему при разборе партий сразу бросается в глаза, что опытные игроки придерживаются одинаковой стратегии. Сначала они укрепляют позиции в углу доски. Обезопасив эту территорию, они начинают наступление по краям. Центр доски — последний в списке приоритетов, и его оставляют на конец партии. Вот почему Сюсаку ждал до 127-го хода, прежде чем сделать «ход красных ушей».

До бесконечности, но не дальше

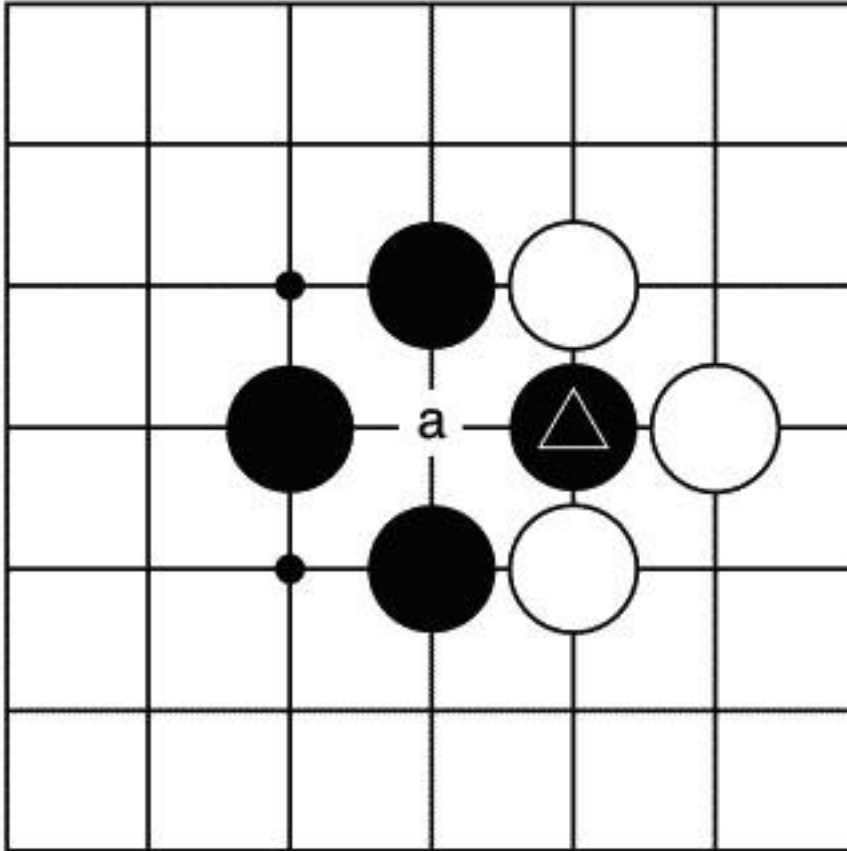
Теперь, когда мы усвоили правила захвата, перейдем к пятому правилу го: запрещено бесконечное повторение последовательности ходов.

Такая ситуация называется *ко*, что в переводе с японского означает «бесконечность». Подобные повторения возможны в го, однако в большинстве случаев они разрушают игру.

Вот как выглядит *ко*. Черные могут захватить камень белых, если поставят свой камень в пункт *a*.



Видите, что произошло? Мы получили ту же ситуацию, но с точностью до наоборот. Если белые поставят камень в пункт *a*, то смогут забрать камень черных.



Без правила, запрещающего ко, белые и черные могут действовать таким образом до истечения времени партии. Вот почему существует следующее правило: в ко-борьбе игрок, первым потерявший камень, не может взять камень противника следующим ходом, что предотвращает бесконечные повторения.

На турнирах иногда вводится правило под названием «супер-ко». Представьте, что на доске возникли две или больше позиции ко. Тогда игроки могут просто повторять последовательность ходов, перемещаясь от одной позиции к другой. Но как только такая последовательность устанавливается, вступает в действие правило супер-ко — либо противники делают другие ходы, либо игра прекращается.

Условия выигрыша

Теперь давайте посмотрим, как заканчивается игра. Правило шестое: игра заканчивается, когда у игроков заканчиваются камни, один игрок сдается или оба игрока пасуют.

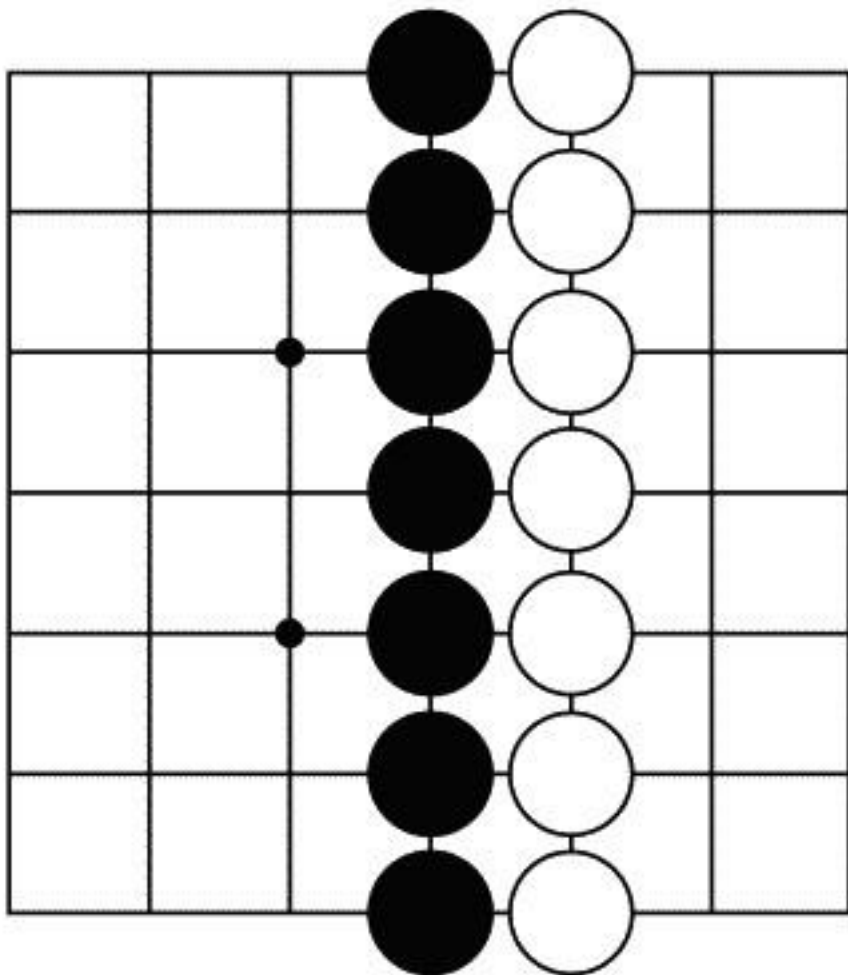
Довольно просто. Но как определить победителя?

Правило седьмое: побеждает игрок, к концу игры окружающий бо2льшую

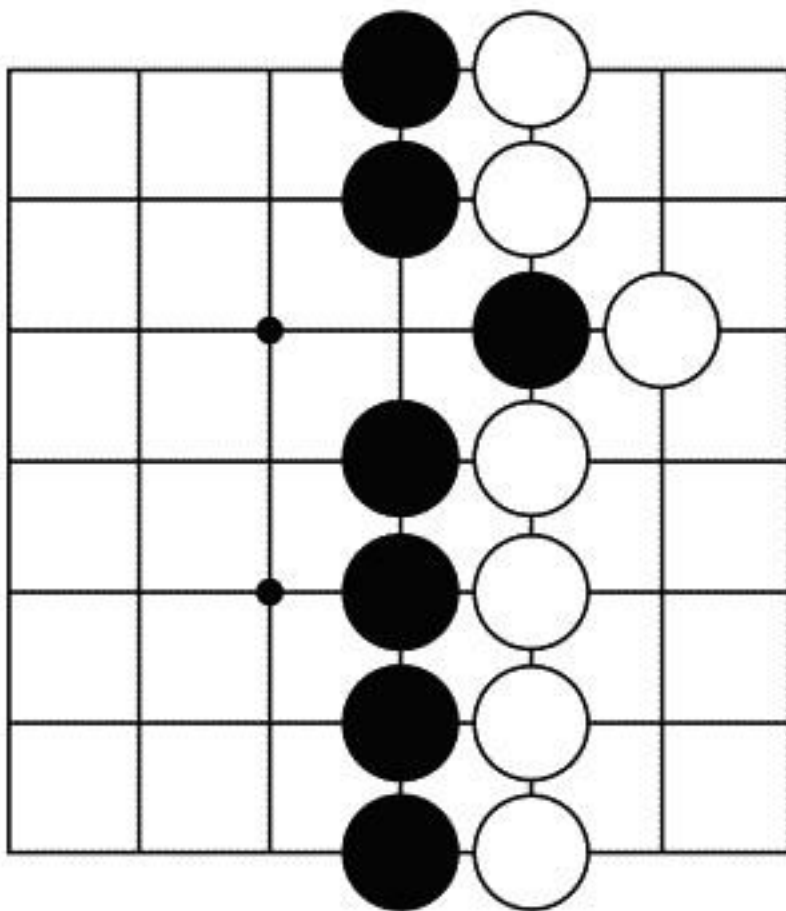
часть доски.

Рассмотрим следующий пример. На доске семь на семь имеется 49 пересечений. Предположим, что доска разделилась следующим образом (рисунок на следующей странице).

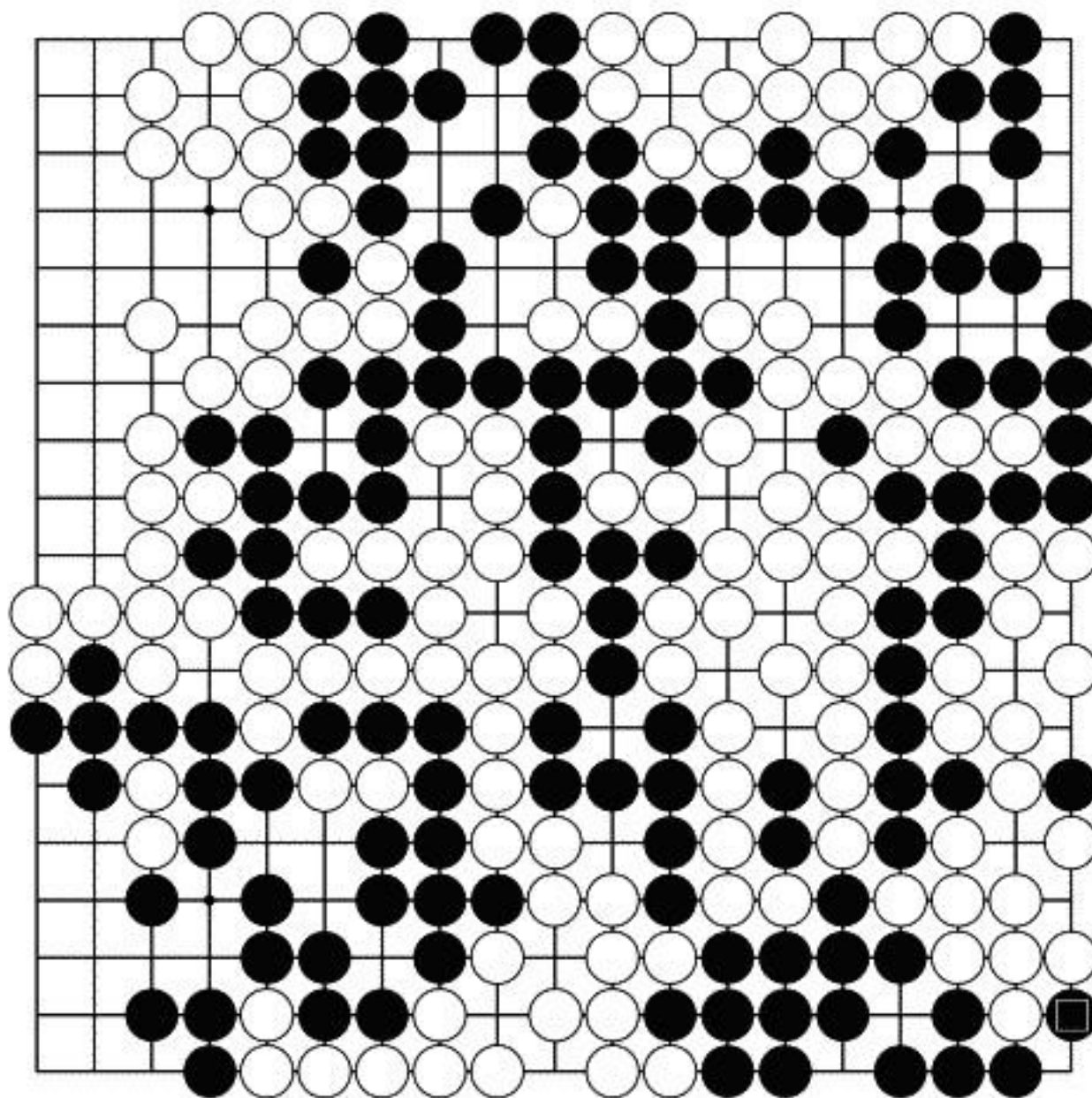
Черные контролируют левую часть доски, или 28 пересечений, а белые — правую, или 21 пересечение. Черные выигрывают.



А вот другой вариант, когда черным удалось еще немного потеснить белых. Теперь у черных 29 очков, а у белых 20.



Подсчет значительно усложняется, если доска больше, а группы разбросаны по всей территории, как в конце «партии красных ушей».

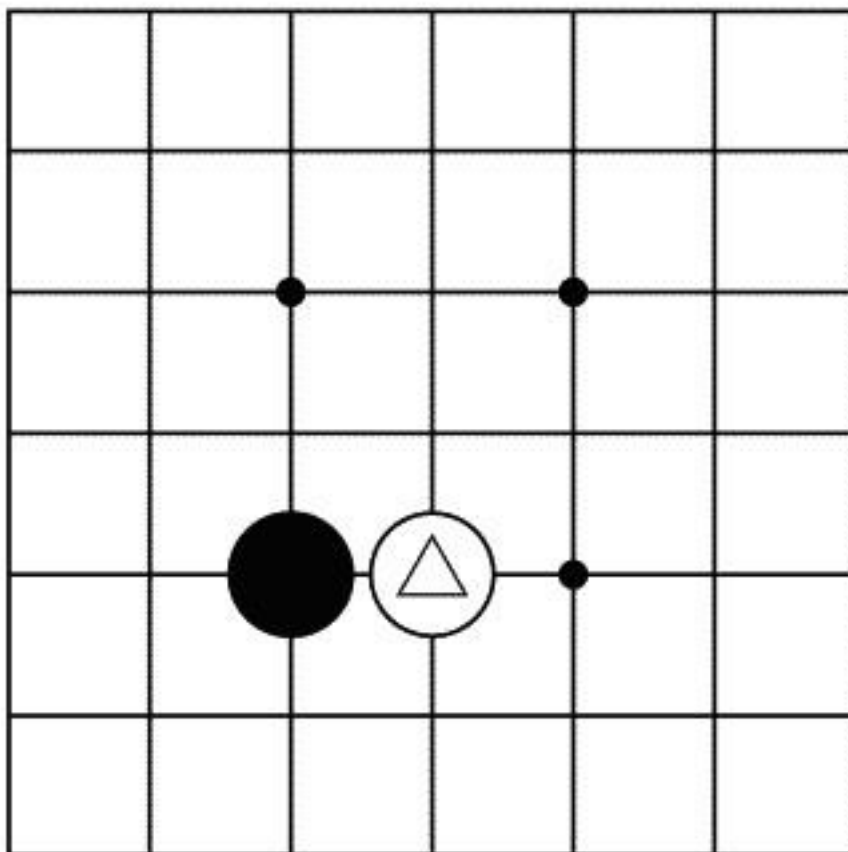


Общий принцип подсчета контролируемой территории таков. Если территория оспаривается, за нее очки не начисляются никому. Иногда опытные игроки уступают территорию противнику без борьбы, так что эта территория считается захваченной. Если по окончании игры возникает спор относительно контроля над территорией, игра возобновляется до разрешения спора.

Оценка контролируемой территории — это навык, требующий практики.

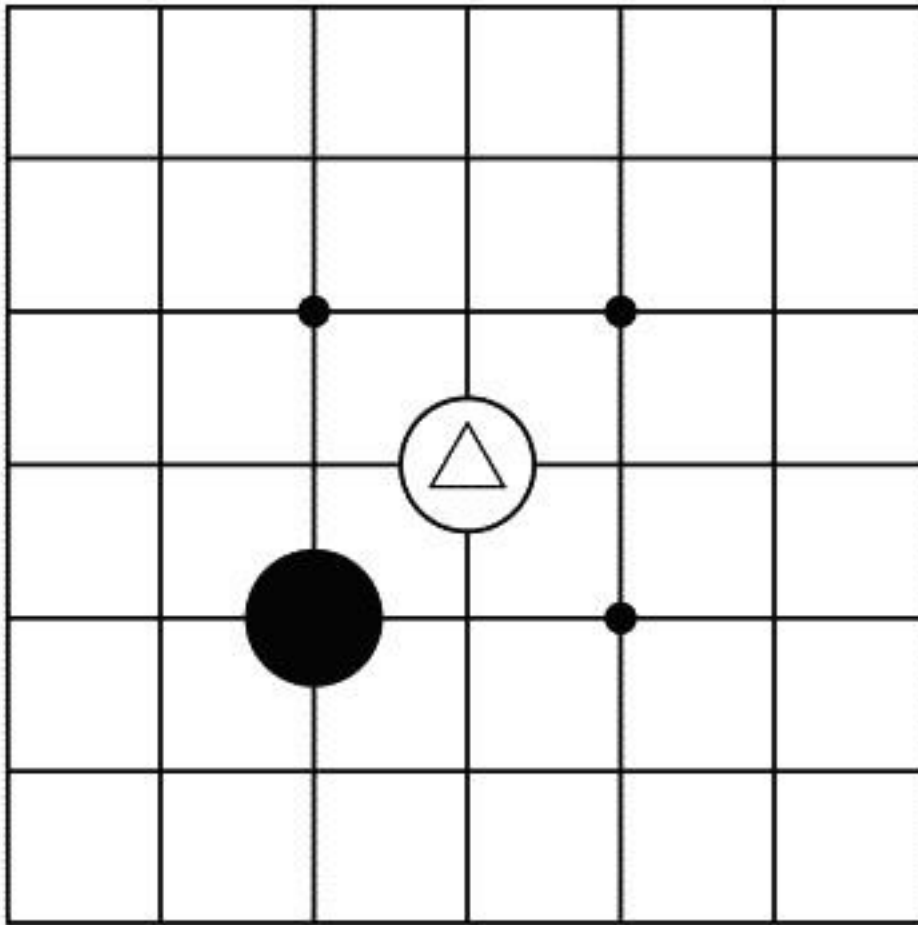
К бою!

Таковы основные правила. А как насчет тактики?



Первое, что нужно знать: как выглядит атака. Прямая атака уменьшает количество степеней свободы у камня противника. Вот так черные атакуют белый камень, отмеченный треугольником (рисунок вверху).

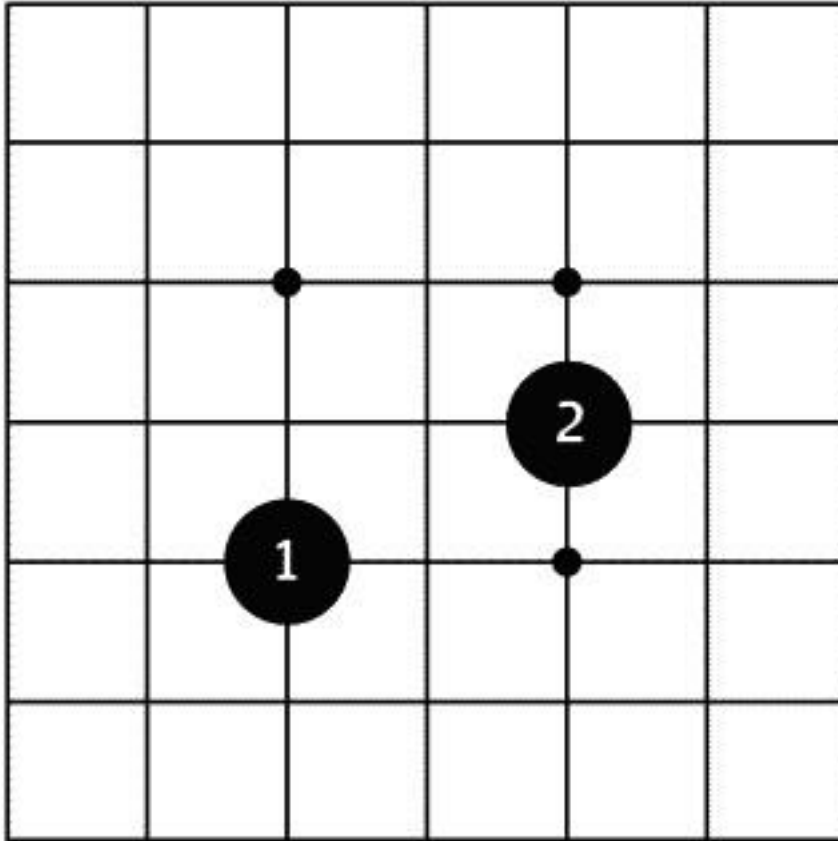
Бывают также косвенные атаки, когда вы ставите камни в непосредственной близости от камней противника. Косвенная атака служит для укрепления позиций перед прямой атакой. На приведенной ниже позиции черный камень угрожает белому, отмеченному треугольником.



Ход конем

Атаки являются основным элементом нападения. Основные элементы защиты именуется *соединениями*. Ходы, сочетающие защиту и нападение, называются *комбинированными*.

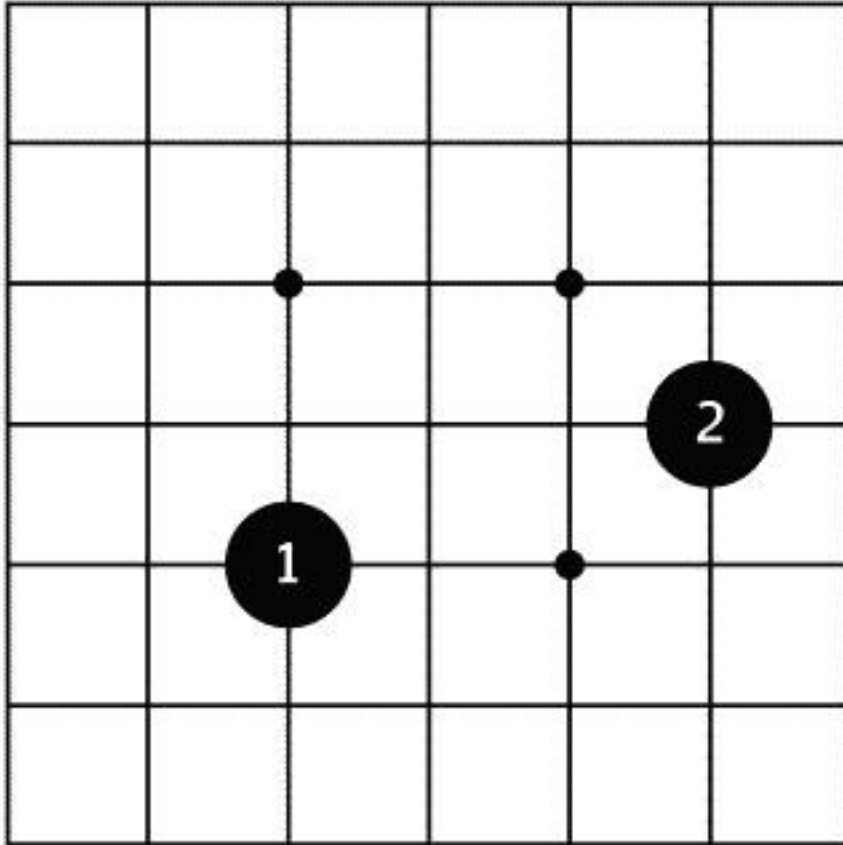
Один из часто встречающихся комбинированных ходов — «ход конем», по аналогии с шахматной фигурой.



Ход конем характеризует баланс между нападением и защитой: камень ставится близко к другим камням, и при необходимости их можно соединить, но в то же время распространяет ваше влияние дальше, чем на один шаг.

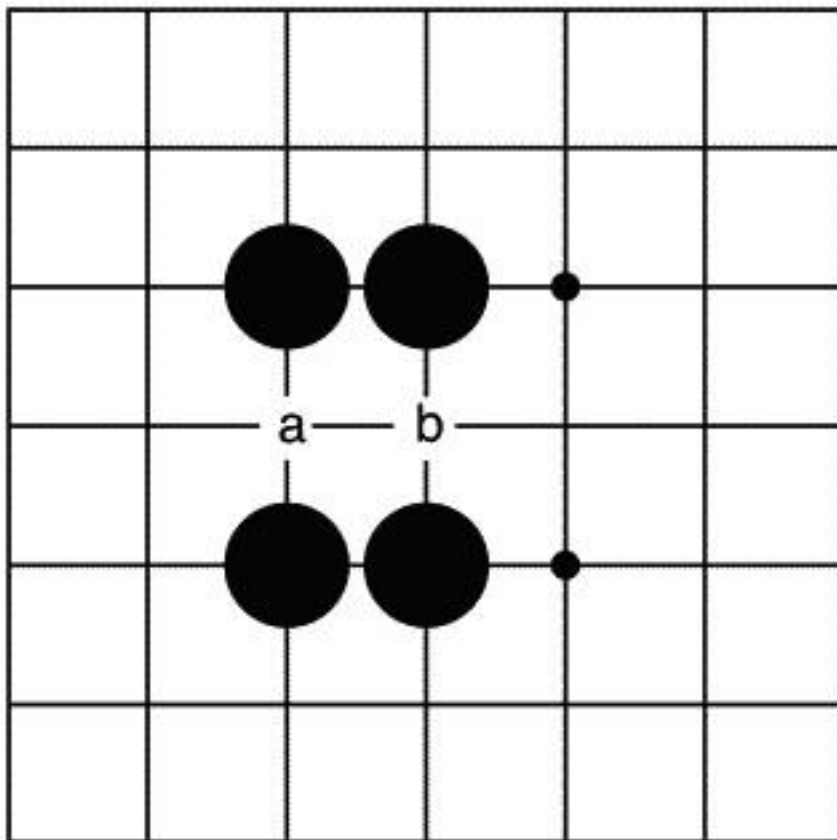
Существует также длинный ход конем, еще более агрессивный (рисунок на следующей странице, вверху).

Этим ходом защита жертвуется в пользу влияния: поле действий игрока расширяется, но при атаке на камень его будет сложнее защитить.



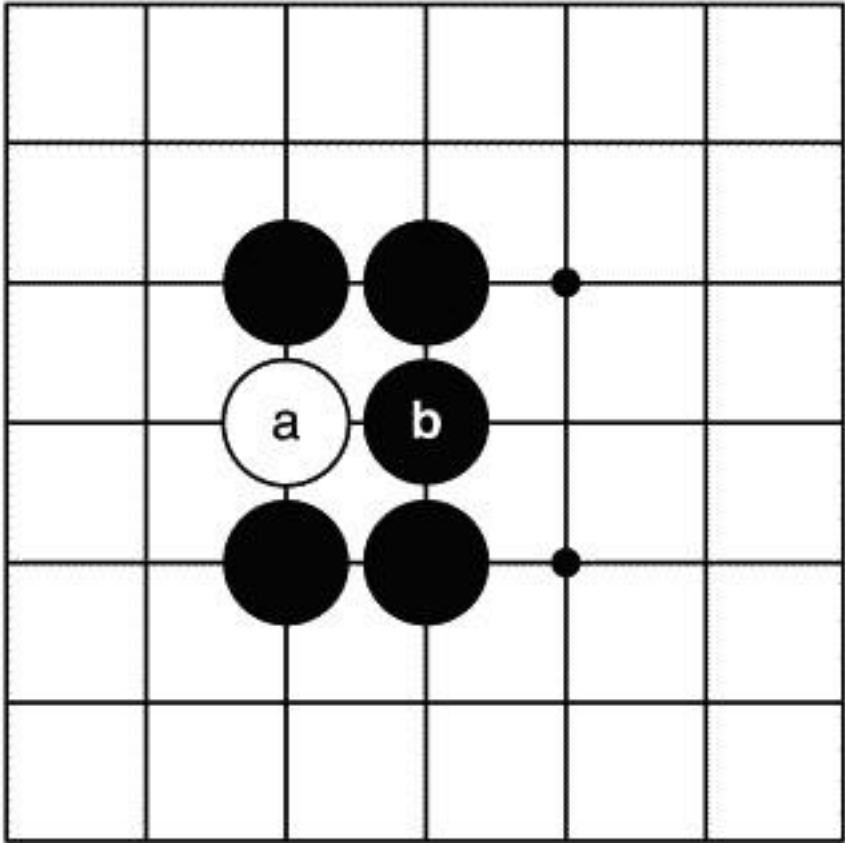
Бамбук прочнее стали

В стратегии защиты имеется очень прочная разновидность соединения под названием *бамбук*. Вот как она выглядит.

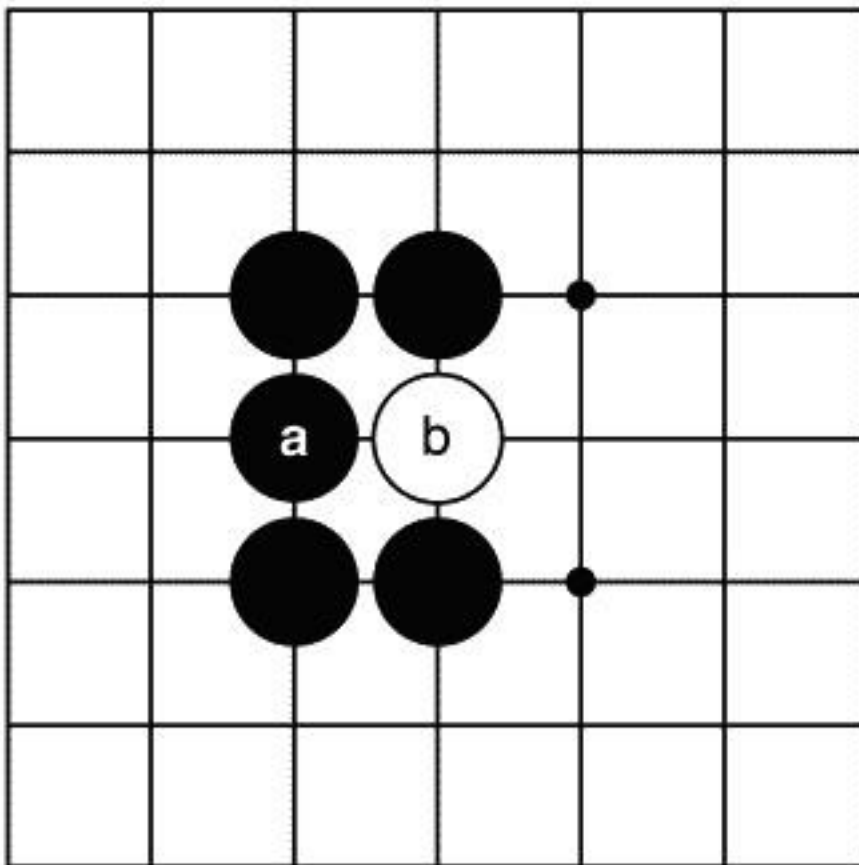


Эта форма соединения очень прочна потому, что при любой попытке ее атаковать не составляет труда соединить свои камни в сильную форму.

В данном случае если белые атакуют, поставив камень на *a*, черные соединяют свои камни ответным ходом на *b*.



Аналогичным образом, если белые атакуют, поставив камень на *b*, черные соединяют свои камни ответным ходом на *a*.



После того как черные соединяют бамбук, белым потребуется девять дополнительных камней, чтобы захватить всю группу. Такой уровень защиты нелегко разрушить.

Довольно просто: такие понятия, как атака, защита и захват, не вызывают трудностей, а ход конем и соединение бамбук — понятная и легко запоминаемая тактика.

Испытанный в сотне битв?

Теперь, когда я понял правила и запомнил элементарные приемы, мне нужно сделать выбор. Следует ли мне с целью обучения сражаться с реальными противниками или самый простой путь совершенствования навыка — это изучать игру по книгам и решать ранжированные задачи?

Опытные игроки на Sensei's Library расходятся во мнениях. Некоторые вспоминают «Искусство войны» Сунь-цзы, утверждая, что лучше быть «испытанным в сотне битв». Другие верят, что оптимальный способ совершенствования для новичков — решение задач го и изучение книг по

стратегии и тактике.

Я не знаю, что и думать. Неподалеку от меня есть клуб го, работающий по средам, но любители древней игры собираются ранним вечером, а это время я обычно посвящаю семье. Мне нравится играть, но в данный момент расписание меня не устраивает.

Есть другой вариант: играть на интернет-серверах. На них меряются силами игроки со всего мира, и серверы работают круглосуточно, семь дней в неделю. Более того, эти игры ранжированы, так что после большого числа игр у вас есть возможность оценить свою силу. Конечно, это не турнирное ранжирование, но уже кое-что.

Я решил попробовать свои силы на двух самых популярных онлайн-ресурсах: IGS (Internet Go Server) и KGS (Kseido Go Server). Играть на них просто: вы регистрируетесь при помощи своего адреса электронной почты, загружаете программу игры в го и входите в нее. Затем вы можете либо вызывать игроков на поединок, либо наблюдать за противостоянием других игроков в реальном времени.

Я зарегистрировался на обеих системах, затем стал подыскивать первого реального противника. Ранга у меня не было, поскольку я только что зарегистрировался, и поэтому я предложил сыграть другому игроку без ранга. Он согласился, и партия началась.

У нас обоих не было ранга, так что фору никто не получил. Я играл черными и считал, что вполне пристойно провел начало партии. Мы с соперником по очереди занимали территории по углам, потом вдоль краев доски. Ходов 30 я чувствовал себя прекрасно.

Затем белые перешли в атаку. Я не беспокоился: моя территория выглядела защищенной, и я соединил камни, усиливая свою позицию.

Ничто почерпнутое мной из книг не подготовило меня к тому, что произошло дальше. Шаг за шагом белые *громили* мою позицию.

Противник вторгся на территорию, которую я считал надежно защищенной. Камни, за которые я не волновался, оказывались захваченными.

Я пытался контратаковать, но все мои атаки отбивались. Я пытался защищаться, но безуспешно. Наконец я сдался.

Когда партия закончилась, я спросил соперника, давно ли он играет.

— Много лет, — ответил он. — Но я только что зарегистрировался на сервере, и поэтому у меня еще нет ранга.

Неудивительно, что он меня разгромил! Я признался, что это моя первая партия, и попросил совета, как можно усилить свою игру.

— Смотрите за играми ранга 20–30-го кю. Вы многому научитесь.

Я поблагодарил соперника за игру и вышел из программы.

Что же пошло не так?

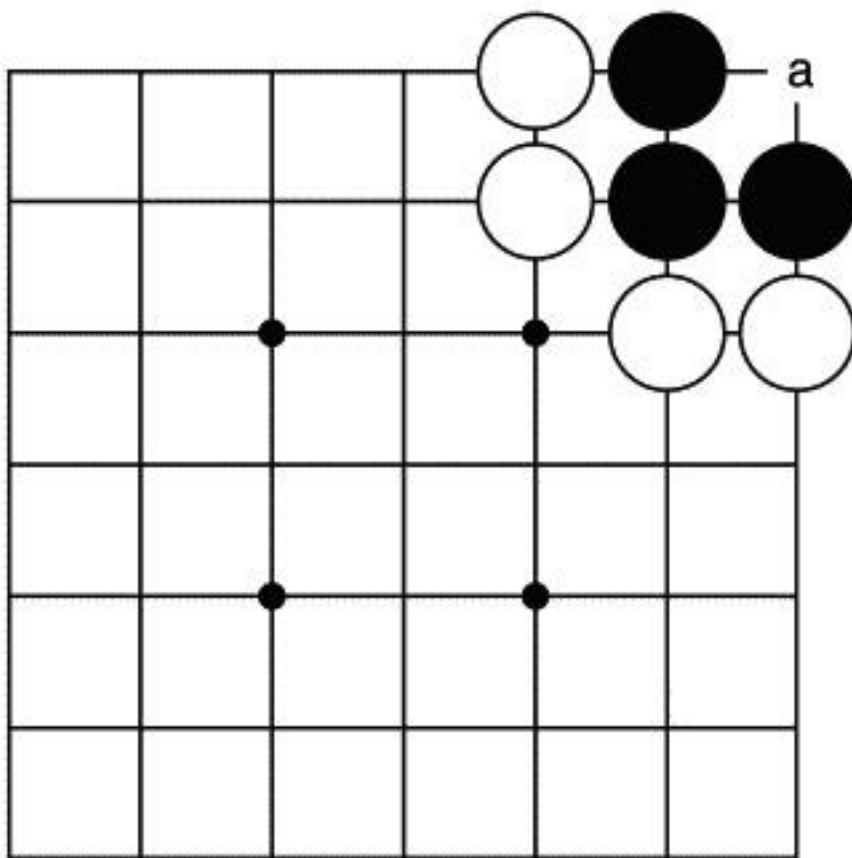
Вспоминая игру, я понял вот что: мне казалось, что я строю защитные

позиции, но на самом деле они ни от чего не защищали. Чтобы понять, почему мои группы были захвачены, нужно разобраться, как вообще атакуют большие группы камней.

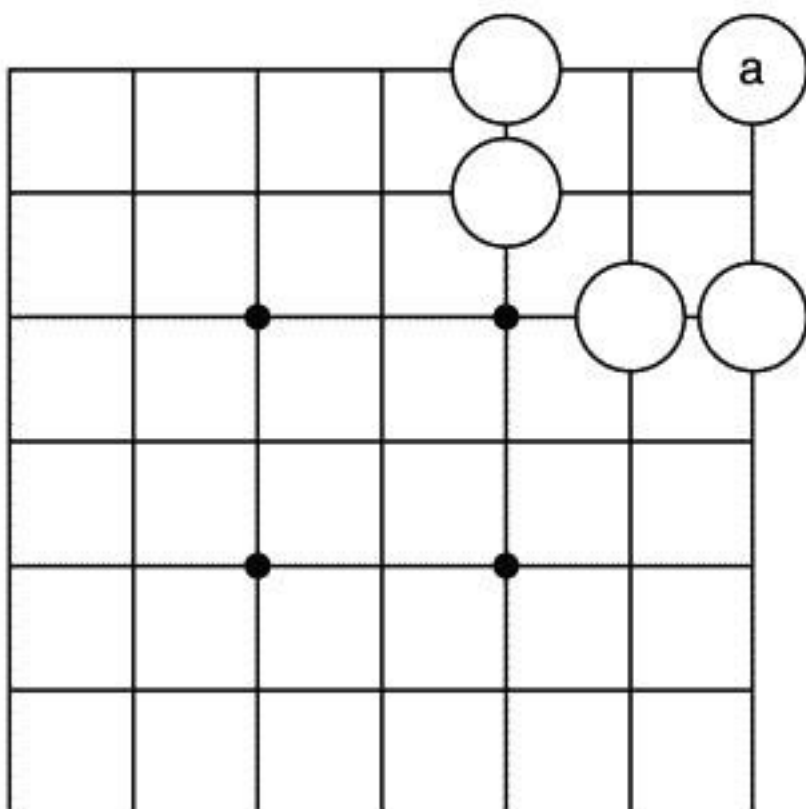
Один глаз — это плохо!

Помните правило, запрещающее самоубийственные ходы? У него есть одно важное исключение: если ход, который в другой ситуации считается самоубийственным, приводит к захвату одного или нескольких камней противника, так что создаются новые степени свободы, этот ход допустим.

Рассмотрим следующий пример. У пересечения *a* нет степеней свободы для белых, поскольку его окружают черные камни.

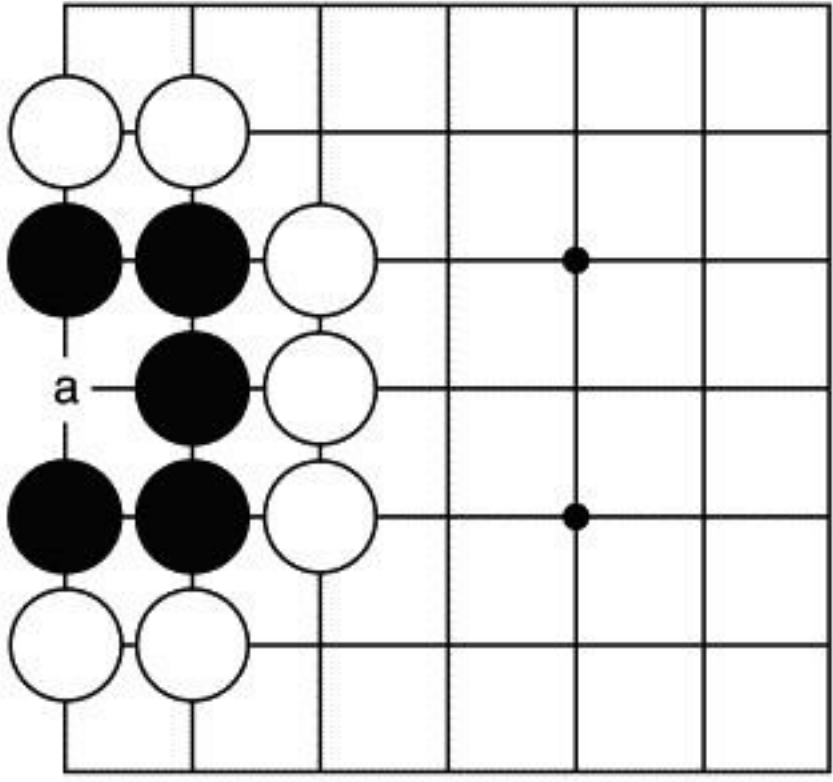


В обычной ситуации правило против самоубийств запрещало бы белым делать ход на *a*, но в данном случае черные камни окружены белыми. Пересечение *a* — это последняя оставшаяся у черных степень свободы. Когда белые ставят камень на *a*, черная группа оказывается захваченной (рисунок на следующей странице, вверху).

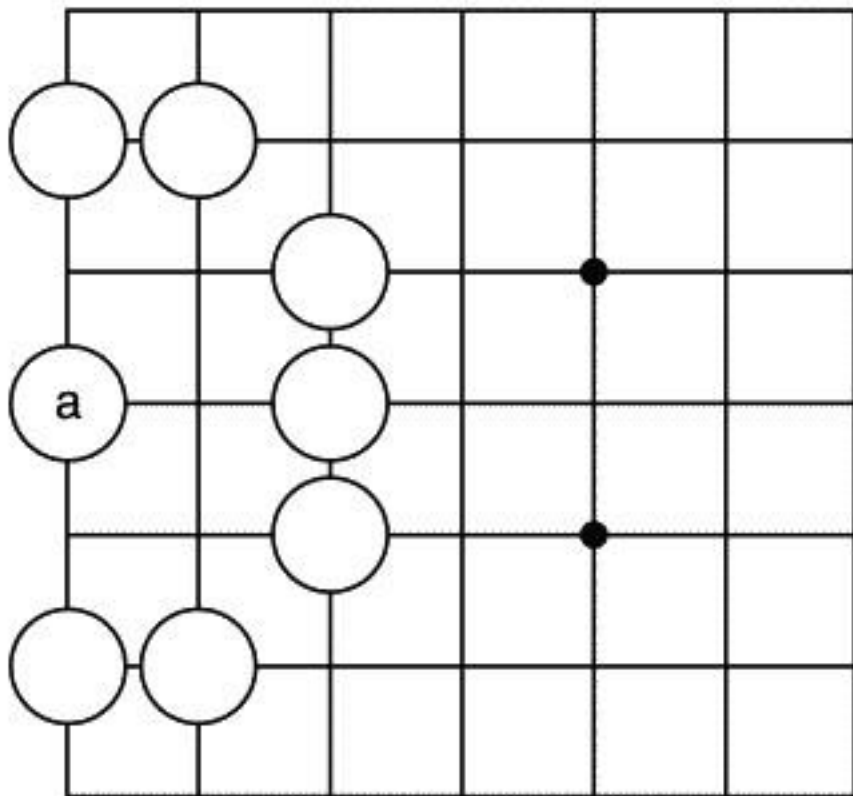


В го такого рода пункты называются *глазами*. В зависимости от ситуации они могут быть источником как силы, так и слабости. Если у группы только один глаз, она слабая.

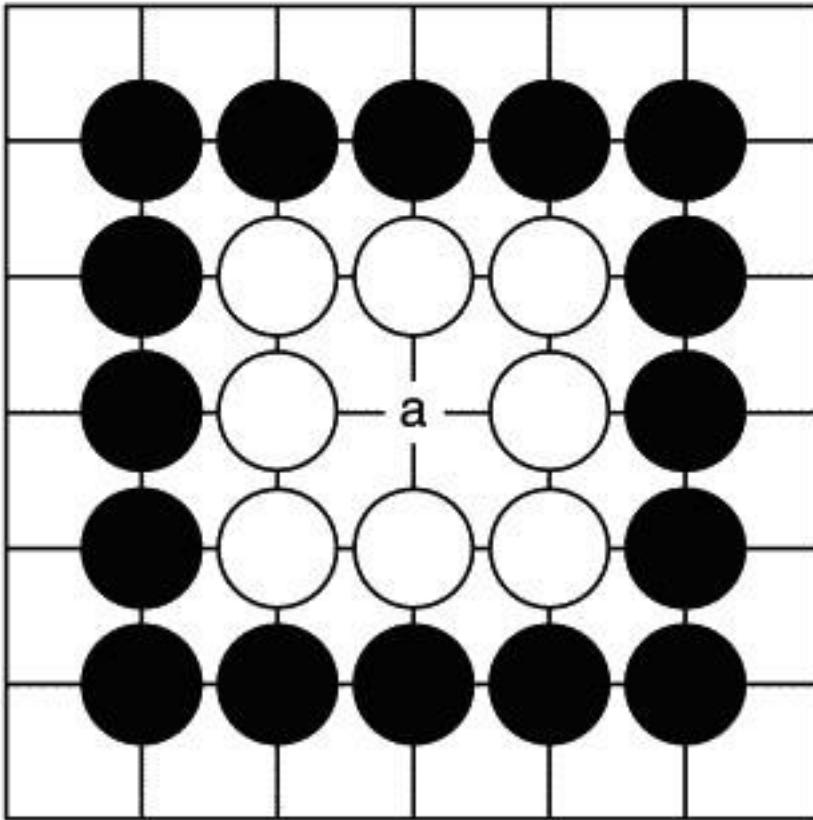
Вот здесь, например, у черных неприятности.



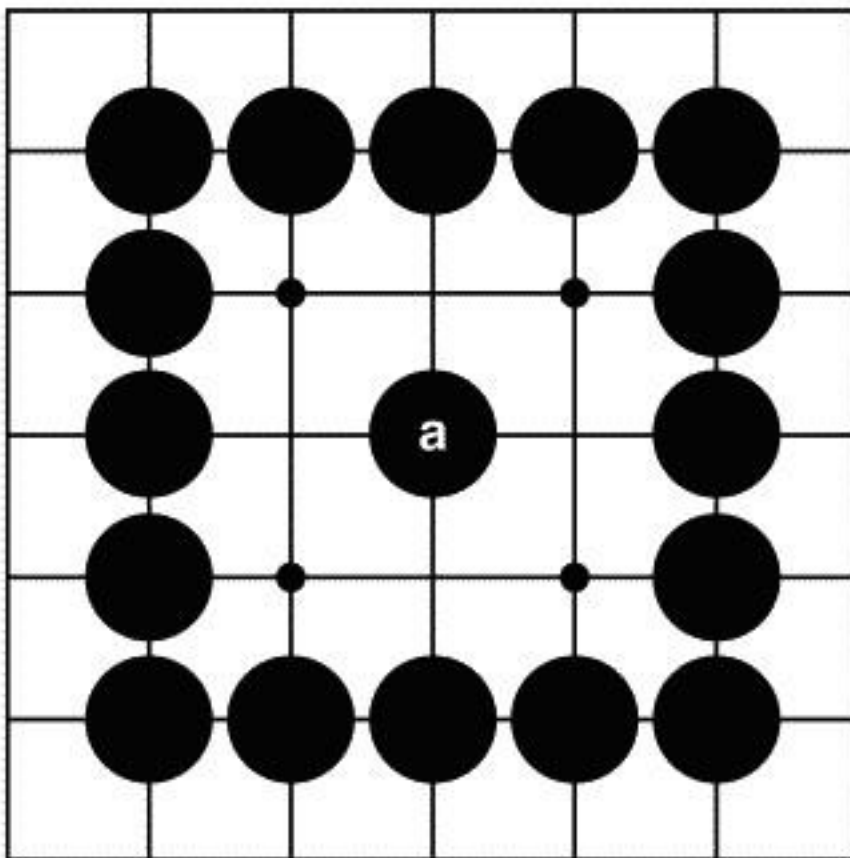
Белые захватывают камни черных ходом на *a*.



А здесь неприятности у белых.



Черные захватывают камни белых ходом на *a*.



Чтобы привыкнуть к этому, требуется определенное время. Группы с одним глазом легко захватить — по существу, они никак не защищены.

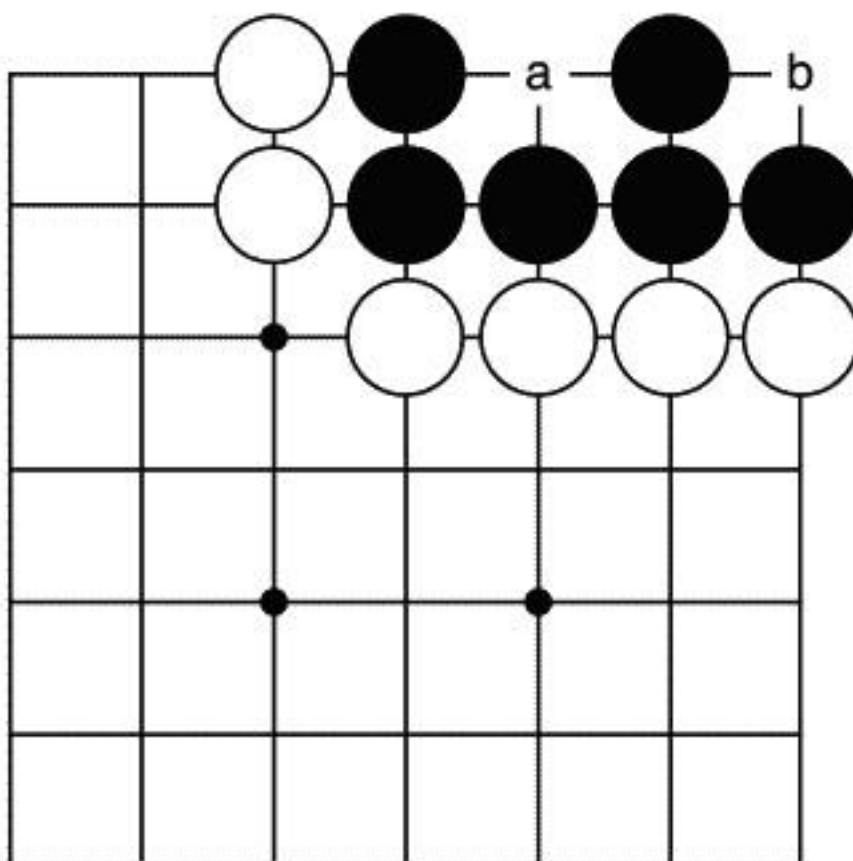
Два глаза — это хорошо!

Это не значит, что глаза вредны. Они необходимы. Данное утверждение кажется странным, пока мы снова не обратимся к правилу, запрещающему самоубийства.

Исключение из правила действует только в том случае, когда ход приводит к захвату группы противника. Если нет, правило запрещает данный ход.

Получается, что группы с двумя глазами несокрушимы. Это нечто вроде крепости, построенной в самом центре вражеской территории: сколько камней противник ни бросил бы в атаку, захватить группу ему не удастся.

Вот пример группы черных камней с двумя глазами.

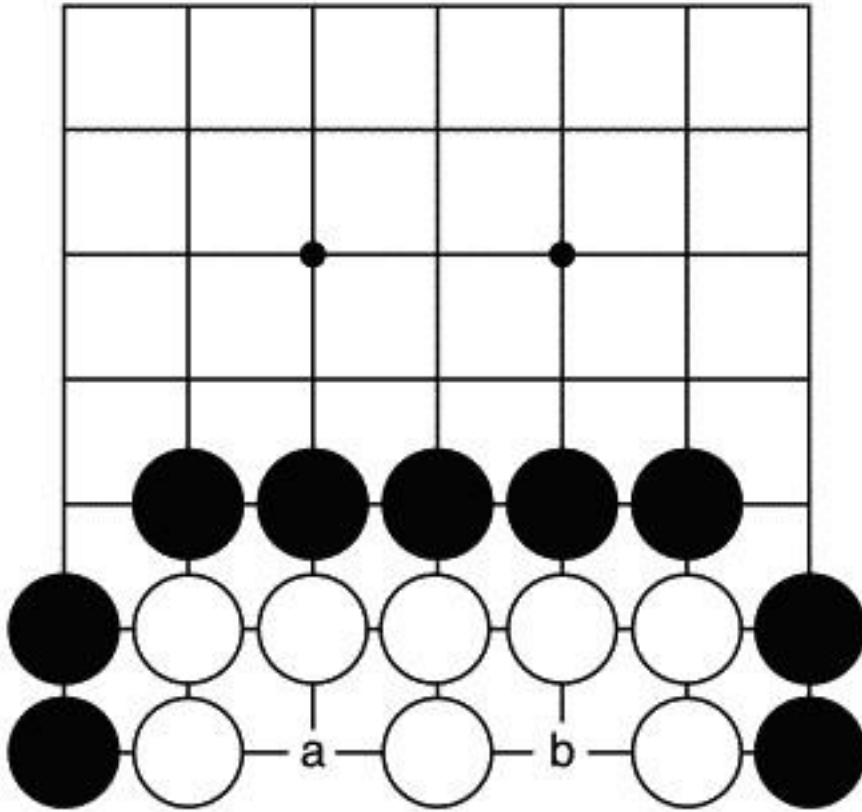


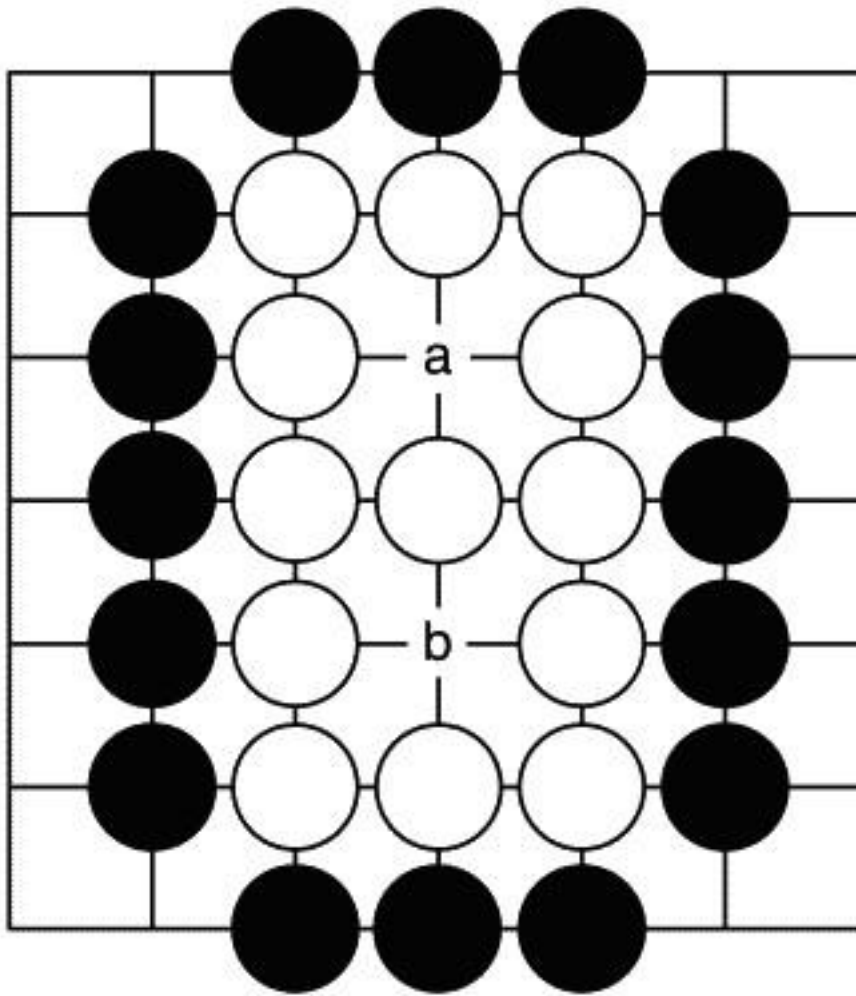
Ход белых на a — это самоубийство. Ход белых на b — тоже самоубийство. Поставить камень на a и на b одновременно белые не могут, и поэтому черные в безопасности — даже несмотря на то, что их группа окружена со всех сторон.

Здесь у группы белых камней два глаза на краю доски (рисунок на следующей странице, вверху).

Далее — у группы белых камней два глаза в центре (рисунок на следующей странице, внизу).

Вы обратили внимание, что для формирования двух глаз в углу доски нужны шесть камней, у края доски — 8, а в центре — 13? Эта разница в эффективности объясняет, почему углы всегда легче защищать, а центр труднее захватить.



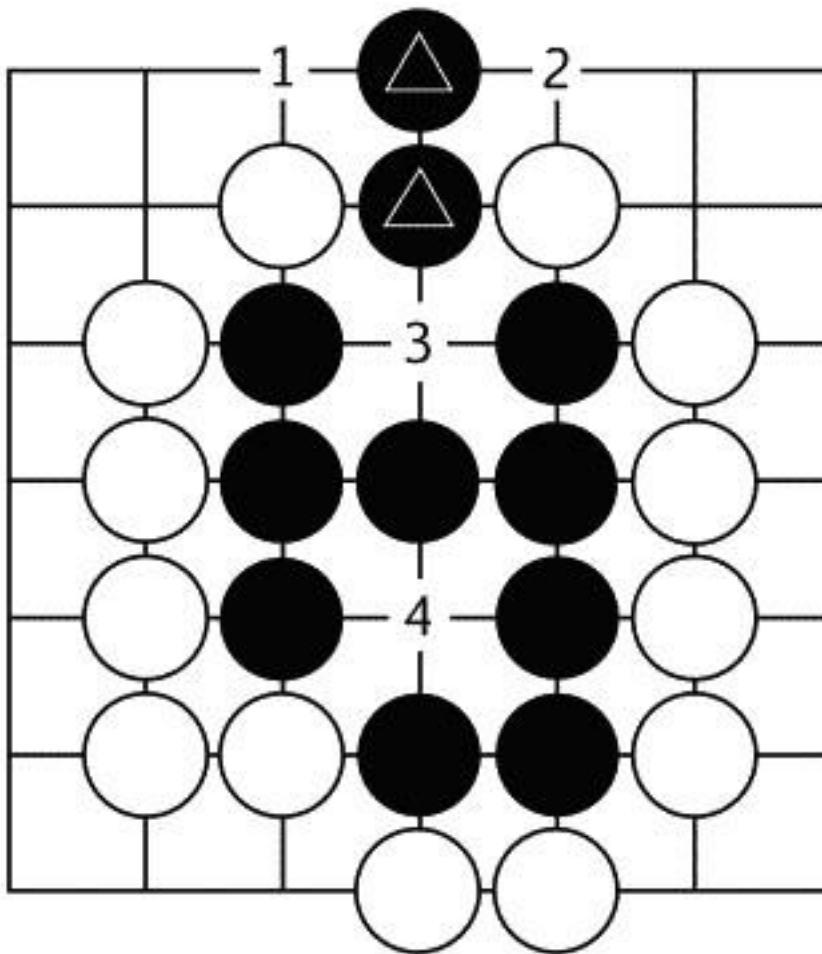


Ложный глаз (Саурана?)

Если два — или больше — глаза в группе делают ее неуязвимой, значит, наш приоритет — как можно быстрее сформировать глаза. К сожалению, не всегда можно определить, является ли форма, которая выглядит как глаз, *действительно* глазом.

Нередко группы с глазами, на первый взгляд прочные, оказываются уязвимыми для атаки. Эти формы называются *ложными глазами*, и они очень опасны.

Вот пример подобной формы. На первый взгляд, черная группа имеет два глаза, но камни в верхней части, обозначенные треугольниками, уязвимы для атаки.



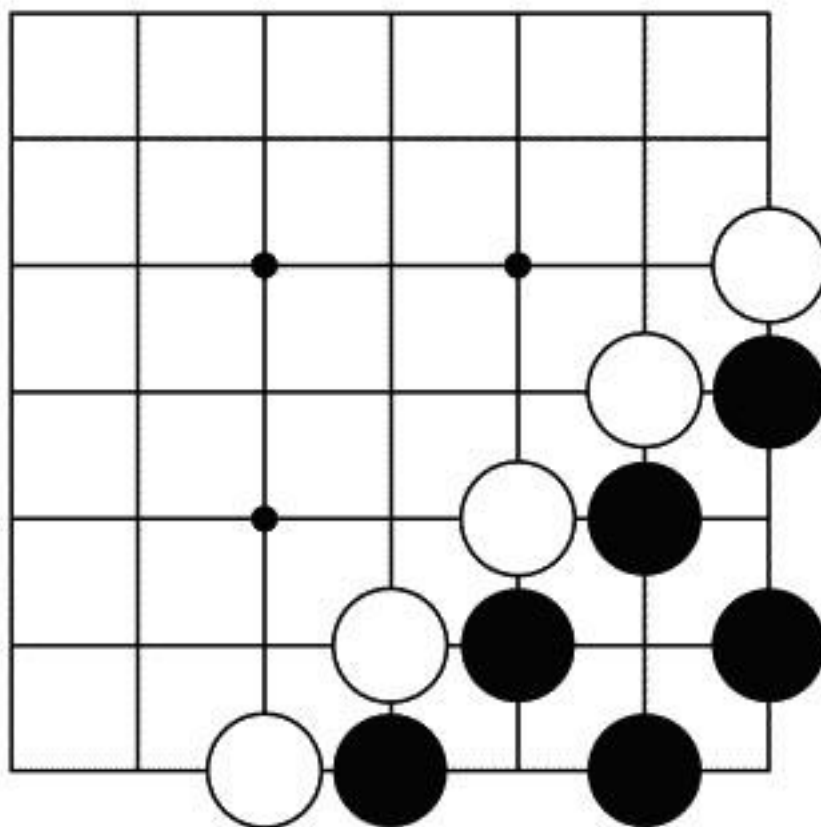
Предположим, что ход белых. Белые могут атаковать на 1 или 2, уменьшив количество степеней свободы. Черные могут попытаться спастись бегством вдоль верхнего края, но каждым следующим ходом белые будут отрезать им путь. Независимо от ответа черных белые захватывают отмеченные камни ходом на 3. Оставшаяся группа черных камней становится уязвимой, и вся группа будет захвачена ходом белых на 4. Вот так!

Поэтому в стратегии го очень важно создавать группы с двумя глазами, избегать ложных глаз, не давать противнику создавать собственные глаза и разрушать его построения при обнаружении ложных глаз. Для того чтобы создавать истинные глаза и различать ложные, требуются опыт и мастерство.

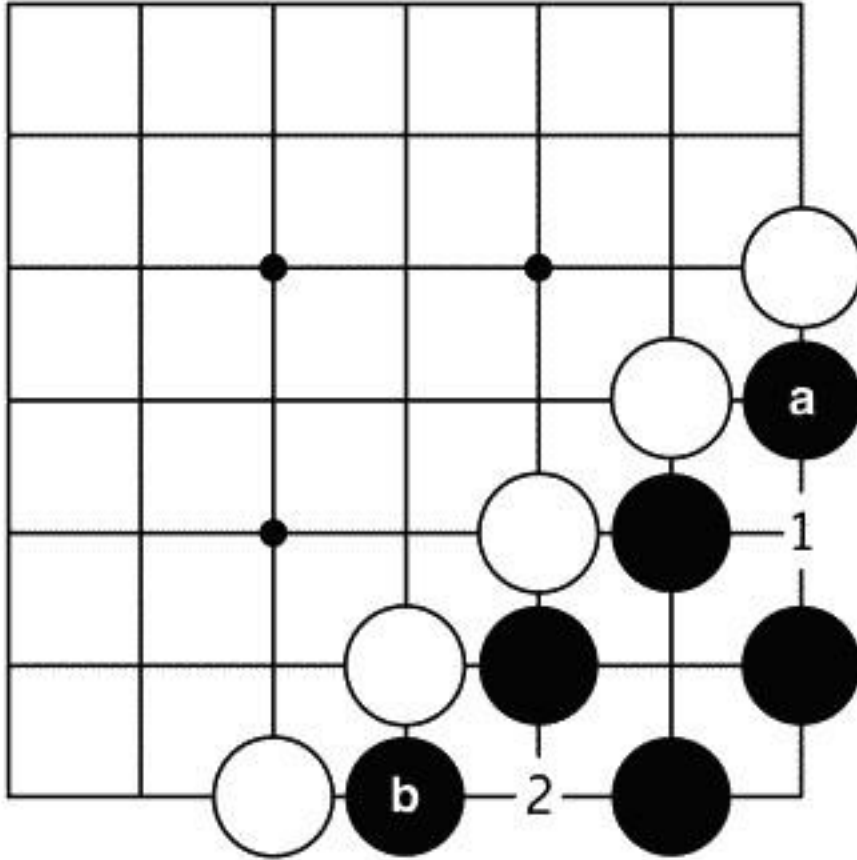
«Уничтожение»

Теперь мы приблизились к пониманию причин моего первого поражения. Я думал, что создаю надежно защищенную территорию, но был уязвим для атаки и не понимал этого.

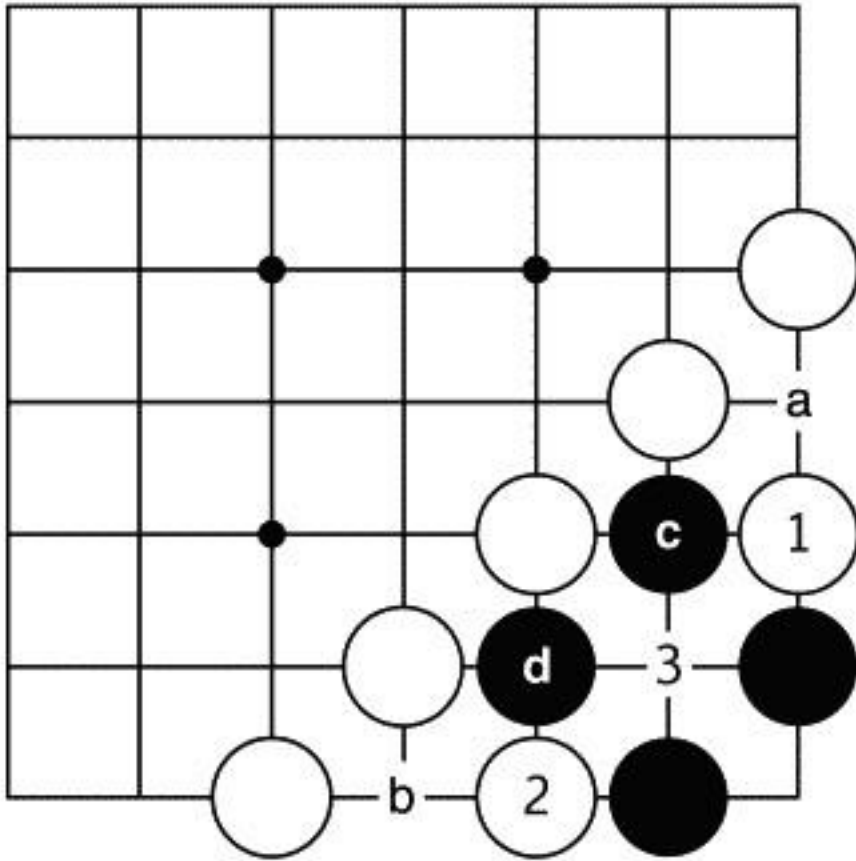
Вот несколько упрощенный пример. Посмотрите на глаза, созданные черными. Группа с четырьмя глазами должна быть неуязвимой, правда?



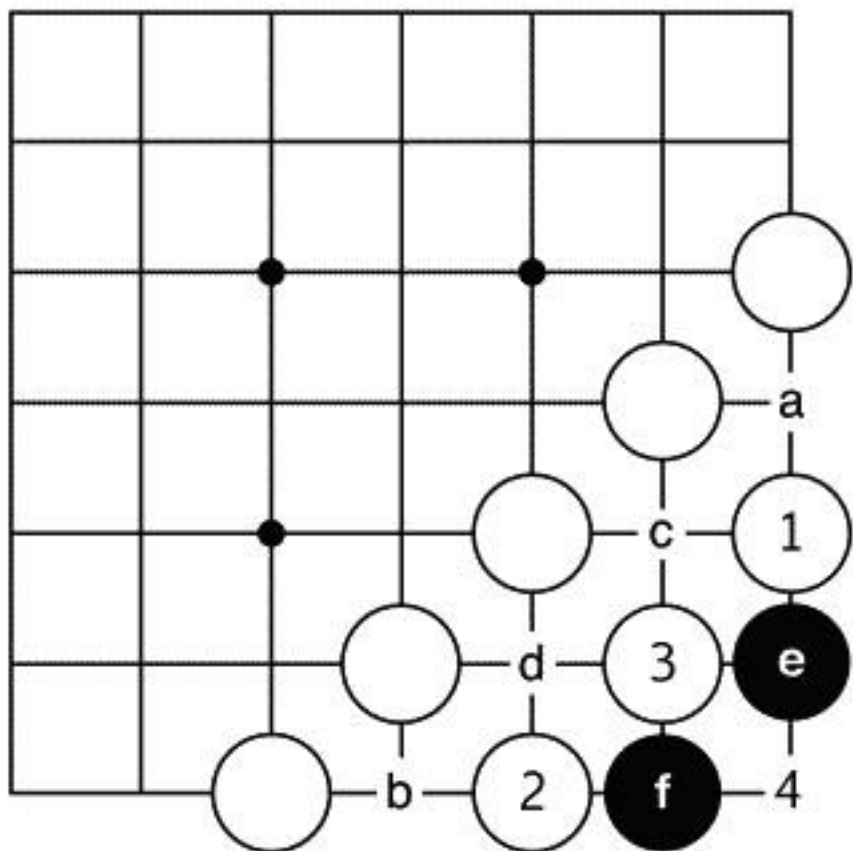
На беду черных, это не так. Если белые сделают ход на 1 или 2, то камни *a* или *b* будут захвачены.



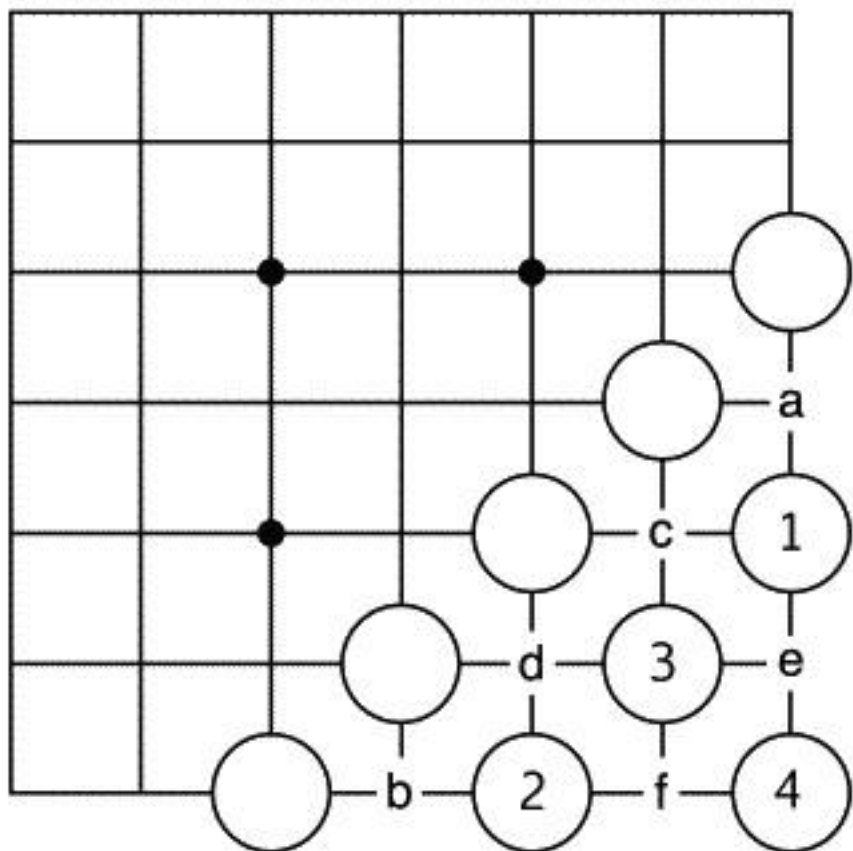
Затем, если белые сделают ход на 3, будут захвачены камни *c* и *d*.



И наконец, если белые сделают ход на 4, будут захвачены камни *e* и *f*.



В результате белые получают возможность полностью уничтожить «неуязвимую» группу черных. Глаза группы были ложными, и черные за это дорого заплатили.



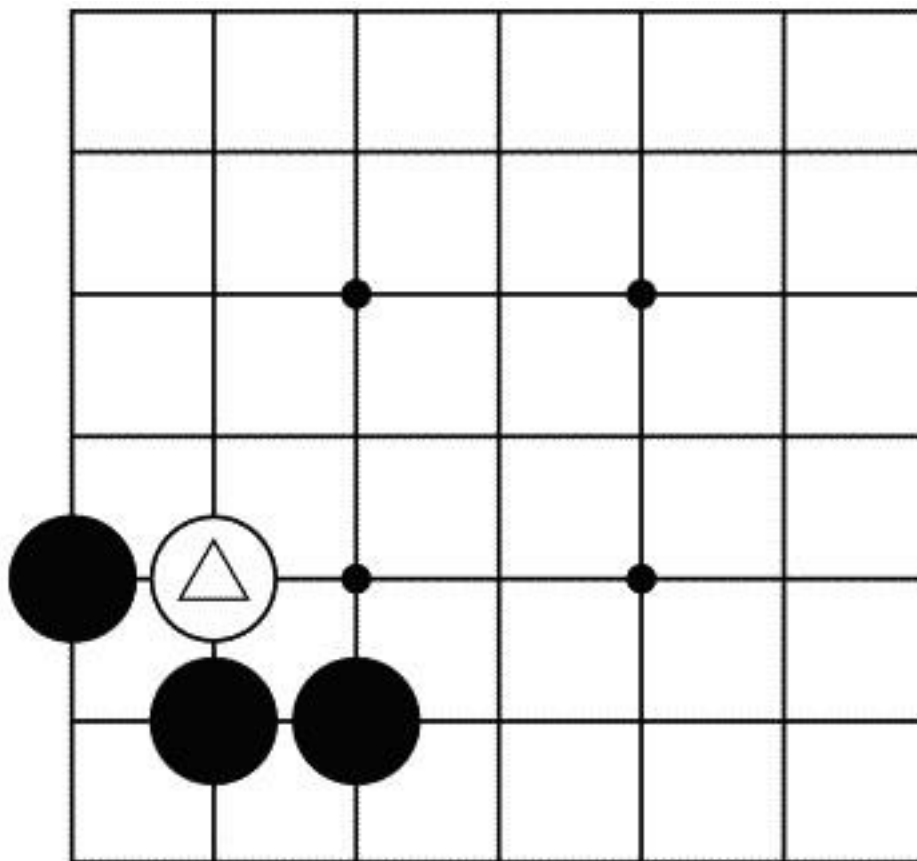
Именно это и случилось со мной. Я считал, что создаю защищенную территорию по углам и вдоль края, но в действительности два моих глаза оказались ложными. В результате противник смог уничтожить все, что я построил. Урок усвоен.

Лестницы

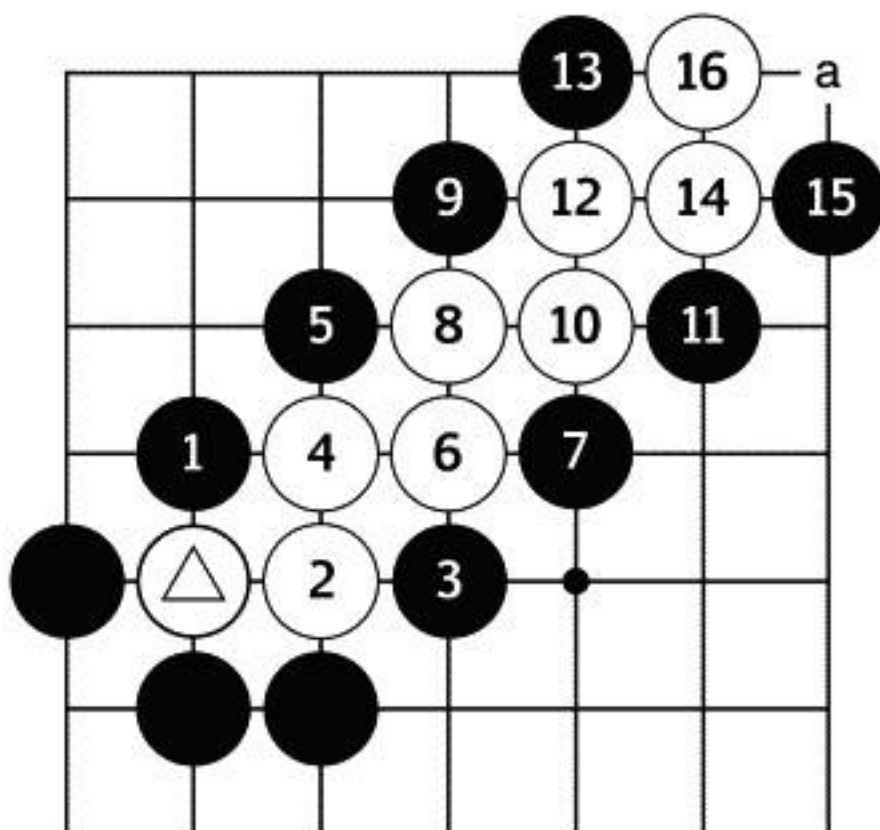
Несколько важных стратегических приемов я не мог понять, пока не увидел их в действии.

Первый называется *лестница*. В ответ на атаку другая сторона оказывается перед выбором: контратаковать или спастись бегством. Зачастую разумнее бежать. Если противник бросается в погоню, то начинается преследование.

Рассмотрим пример. Здесь белому камню угрожает атака.



Черные атакуют на 1, белые отступают на 2. Черные атакуют на 3, белые отступают на 4. Черные атакуют на 5, белые отступают на 6 и так далее. В конечном итоге получается следующая позиция.

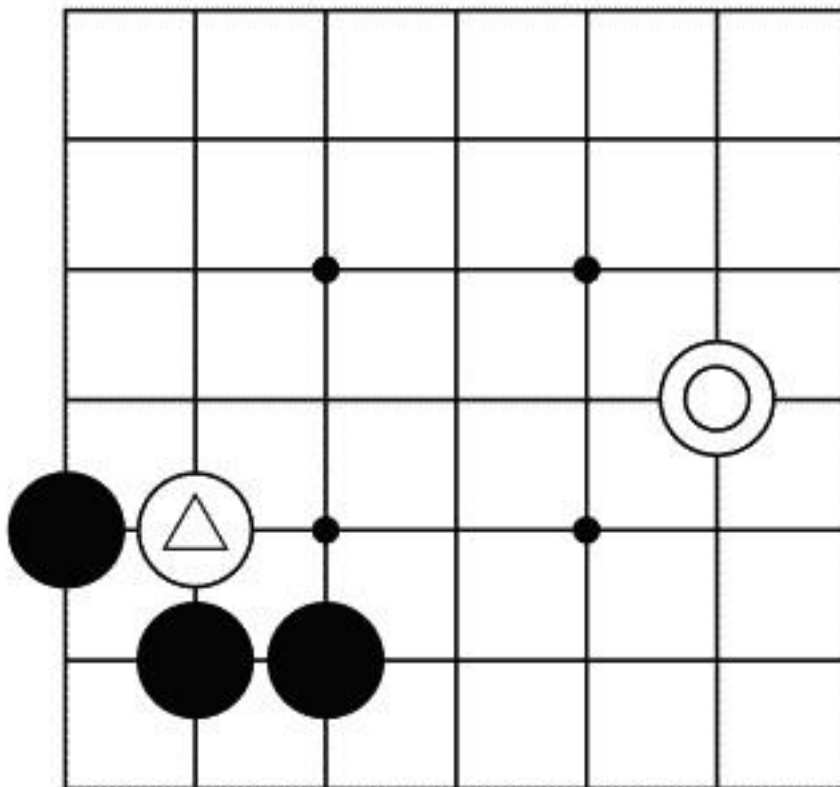


Теперь очередь черных, и белым больше некуда бежать. Когда черные сделают ход на *a*, все белые камни будут захвачены, и под контроль черных перейдет вся доска. Игра окончена!

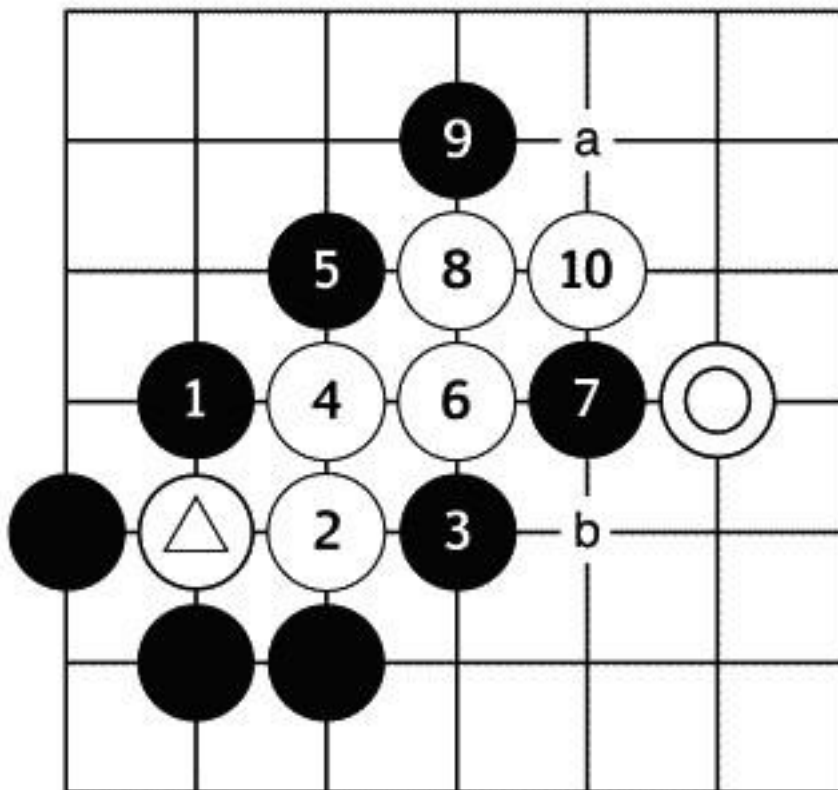
В го есть старая поговорка: «Если вы не знаете, что такое лестница, не садитесь играть». Рассмотренная ситуация как нельзя лучше подтверждает эти слова.

Теперь снова рассмотрим ту же позицию, но с небольшим изменением: у белых есть камень (отмеченный кружком) в дальней правой части доски. Что произойдет в этом случае? (Рисунок на следующей странице, вверху.)

Начало одинаковое. Черные атакуют на 1, белые отступают на 2. Черные атакуют на 3, белые отступают на 4.



Ходы повторяются, пока белые не отступают на 10. Образовалась следующая позиция.



Если черные атакуют на *a*, продолжая последовательность ходов, белые захватят черный камень на 7, сыграв на *b*. Если черные попытаются защититься, поставив камень на *b*, белые ускользнут от атаки черных ходом на *a*.

Один камень на противоположном краю доски спас белых от полного уничтожения. Именно эта особенность и делает игру го интересной: камень, который кажется изолированным и не участвующим в основном действии, способен изменить ход игры.

Распознавать лестницы — это искусство. На доске размером 19 на 19 есть много способов начать погоню или избежать атаки, построив лестницу. В этих случаях нужно уметь предсказывать, кто выйдет победителем в этой гонке преследования.

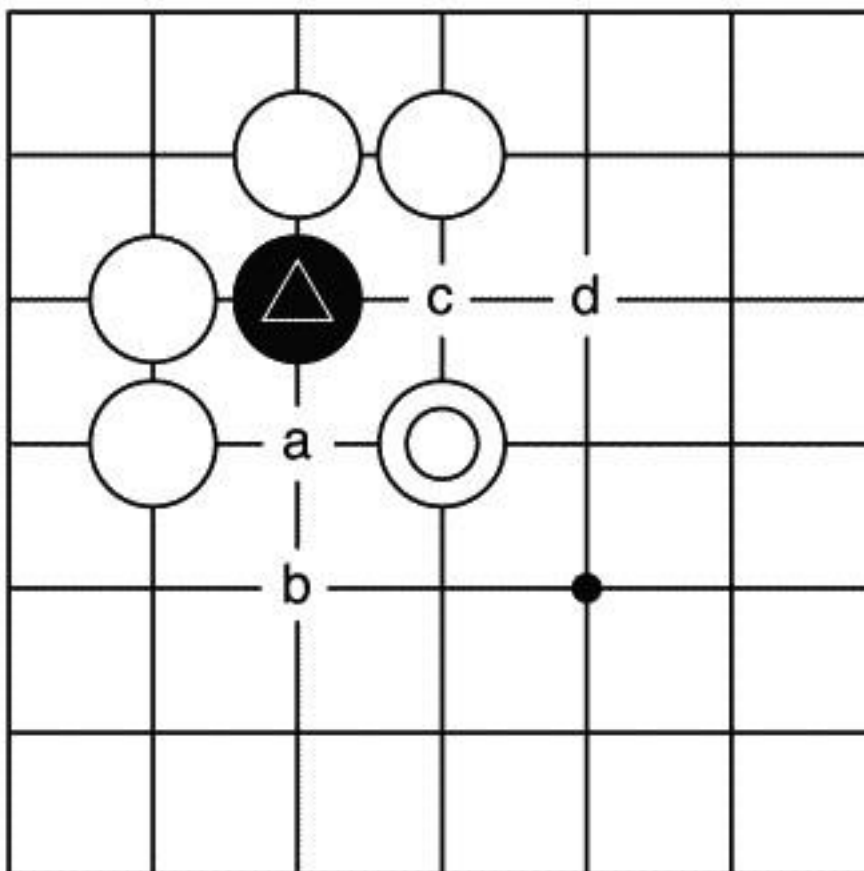
Если вы сомневаетесь в победе, лучше пожертвовать одним или двумя камнями, чем потерять всю группу. Если расчеты предсказывают победу, то в ваших интересах спровоцировать соперника на действия, которые приведут к катастрофе, что в конечном итоге обеспечит вам огромное преимущество.

Закидываем сети

Поставить камень на пути возможной лестницы — это очень удачный

ход. Еще одна хорошая стратегия — поймать противника в ловушку, окружив камнями.

На приведенной позиции белые хотят захватить черный камень (отмеченный треугольником). Но вместо прямой атаки белые набрасывают *сеть*, окружающую черный камень, но не плотную (рисунок на следующей странице).



Если черные отступают на *a*, белые блокируют их ходом на *b*. Если черные пытаются спастись, поставив камень на *c*, белые блокируют его ходом на *d*.

В сыгранной партии мой противник не только умел создавать и распознавать глаза, но и провоцировал меня на ненужное расширение позиции. Я оказывался втянутым в лестницы, в которых не мог победить, или попадал в сети, которые замечал слишком поздно.

Все эти приемы помогают опытным игрокам побеждать. Не зная основных форм, таких как глаза, лестницы и сети, вы неизбежно потерпите поражение от игроков, которые умеют их создавать и пользоваться полученными преимуществами.

Как можно больше практиковаться

После проигрыша я последовал совету соперника и попытался наблюдать за партиями противников с рангом от 10-го до 20-го кю. К сожалению, таких оказалось мало.

Интернет-серверы для игры в го не очень подходят для начинающих. Большинство серверов и программ существуют уже давно — интерфейсы разработаны еще в 1990-е годы, и разобраться в них не так просто.

Кроме того, на этих серверах сложно найти игроков с низким рангом. А если вы играете не очень хорошо, то постоянно проигрываете, что не доставляет никакого удовольствия. Большинство матчей, которые я нашел в сети, проходили между игроками ранга 8-го кю и выше.

Я пытался наблюдать за несколькими партиями, но результат в смысле обучения был невелик. Я мог следить за ходами игроков, но ничего не знал о причинах, побудивших их сделать эти ходы. Иногда ходы следовали один за другим так быстро, что я не успевал их проанализировать. Я не сомневался, что позиция каждого камня чем-то мотивирована, но отсутствие опыта не позволяло мне понять этот мотив.

Полноценная партия в го может длиться больше часа, и это стало еще одним препятствием. Несколько раз я начинал следить за игрой, но был вынужден отвлечься, чтобы успокоить Лелу или помочь Келси по хозяйству. А поскольку игры проходили в реальном времени, окончания партии я уже не видел.

Как показал мой опыт, метод «сотни сражений» тоже не самый лучший путь обучения новичка. Я решил сменить стратегию, отказавшись от метода проб и ошибок в пользу структурированного обучения.

Человек против компьютера

Каждый вечер я проводил 45 минут за изучением задач го, сначала с помощью программы SmartGo на iPad, а затем с помощью серии книг «Ранжированные задачи го для начинающих» издательства Kiseido.

После задач я играл партию против SmartGo на маленькой доске 9 на 9. На первых порах программа побеждала меня даже при форе в восемь камней. Какой позор!

Однако мои навыки быстро совершенствовались. Я научился использовать преимущество фора для наступления, стремительно атакуя противника и при первой возможности строя прочные соединения.

В игре против программы я довольно быстро уменьшил фору до шести

камней, потом до пяти. После 30 партий я подобрался к форе из четырех камней.

Оказалось, что очень трудно вести подсчет очков: иногда я полагал, что все в порядке, но, взглянув на счет, убеждался, что сильно ошибся. Это серьезная проблема, поскольку для победы в партии необходимо делать ходы, расширяющие контролируруемую территорию. Зачастую ход, который я считал правильным, был просто потерей времени и открывал возможности для противника.

Разновидности ошибок

Помимо решения задач, я читал книги из списка литературы, рекомендованного Sensei's Library. Они мне здорово помогли.

Особенно полезным оказалось изучение книги «Как не нужно играть в го». По мнению автора, Юань Джоу, новички делают одинаковые ошибки, которые им дорого обходятся.

- **Неопытные игроки просто отвечают на действия противника.** Например, столкнувшись с атакой, слабые игроки автоматически начинают защищаться, уходить или контратаковать на том же участке доски, не рассматривая другие возможные варианты.

- **Неопытные игроки не видят всю доску.** Иногда лучший ход совершается не там, где происходят активные действия, а на другом краю доски. Сражение захватывает, и слабые игроки могут не видеть возможности, возникающие в более спокойных частях гобана.

- **Неопытные игроки не делают самого выгодного хода.** Го требует мышления в терминах выгоды и затрат: на каждый ход тратится камень, и мы всегда стремимся получить максимум за свои деньги. Нередко жертва одного камня или даже двух позволяет получить большую выгоду в другом месте.

- **Неопытные игроки не ценят сэнтэ.** В го очень важна инициатива: вы стремитесь по возможности навязать противнику ответные ходы. Вы хотите, чтобы он не следовал своему плану, а беспокоился из-за возможных потерь и игнорировал приобретения. Тот, кто создает и сохраняет инициативу-сэнтэ, почти всегда выигрывает.

- **Неопытные игроки не умеют оценивать территорию.** В результате они тратят много времени на сражение местного значения или на захват угла доски, а в это время противник захватывает большую территорию.

- **Неопытные игроки завидуют позиции противника.** Джоу называет это «проблемой красного глаза», которая обычно приводит к тому, что игрок делает невыгодный для себя ход просто из ощущения, что территория

противника становится слишком большой.

● **Неопытные игроки склонны принимать желаемое за действительное.** Очень легко соблазниться возможностью захватить группу противника с помощью всего лишь двух камней, забывая, что вы не можете сделать два хода подряд, не учитывая ответный ход. В результате вы впустую тратите драгоценные камни, напрасно надеясь, что противник не разгадает ваших намерений.

Все это, особенно склонность принимать желаемое за действительное, в полной мере относится ко мне. Сколько раз я атаковал группу противника, надеясь, что он не заметит моих намерений? Сколько раз я втягивался в мелкие стычки, не обращая внимания на общую картину?

После прочтения «Как не нужно играть в го» моя игра существенно улучшилась. Главное, с чем мне пришлось бороться, — внутренняя убежденность, что захват камней противника служит кратчайшим путем к победе. Это справедливо для шахмат и шашек, но не для го.

Захват камней помогает выиграть, но не является условием победы. Цель го — захват *территории*, и во многих случаях для этого совсем не требуется захватывать камни противника. Этот урок я усвоил не сразу: отвоевывать камни было приятно, даже если мои действия приводили к обратному результату.

Народная мудрость

Афоризмы относятся к одним из самых распространенных средств обучения го. Поскольку игра очень сложна, эмпирические правила помогают игрокам вспомнить, как нужно действовать в стандартных ситуациях.

Одним из первых примеров такого подхода к обучению могут служить «10 золотых правил го», приписываемых игроку по имени Ван Цзисинь, который жил во времена династии Тан, правившей в Китае приблизительно с 600 по 900 год н. э. [7]

- Жадный не добьется успеха.
- Не спеши вторгаться на территорию противника.
- Атакуя противника, позаботься о себе.
- Пожертвуй камнем ради *сэнтэ*.
- Откажись от малого ради большого.
- В случае опасности прибегни к жертве.
- Строй плотные формы, избегай поспешных ходов.
- Ход должен отвечать противнику.
- Против сильных позиций играй осторожно.

- Ищи мира, уклоняйся от сражений в изолированной или слабой позиции.

Правила го афористичны. Существует огромное количество поговорок, связанных с этой игрой.

- Лучший ход соперника — ваш лучший ход.
- Не знаешь, как играть, — играй *тэнуки* (промежуточный ход).
- *Ханэ* (камень, окружающий для атаки) несет смерть.
- Новички играют *атару*.
- *Поннуки* стоит 30 очков.
- Сети лучше лестниц.
- Прежде чем бить, сожмите кулак.
- Жертвуйте малым, чтобы получить большое.
- Богач не ищет повода для ссоры.
- Тому, кто вскочил на спину тигра, трудно соскочить. (Не начинайте то, что не намерены заканчивать.)

Здесь есть над чем подумать: разумно не только знать поговорки и понимать причины их появления, но также видеть ситуации, когда их нужно игнорировать. Ведь одна из поговорок гласит: «Нельзя слепо следовать поговоркам».

Многие поговорки относятся именно к новичкам. Моя любимая звучит так: «Только после десятого удара вы увидите кулак». И ее продолжение: «А только после двадцатого сумеете защититься». Мой опыт подтверждает эту мудрость: сильные игроки могут атаковать меня так, что я не *распознаю* атаку.

Мне еще нужно многому научиться.

«Вопросы на миллион»

Прошел почти месяц занятий го, прежде чем я наткнулся на статью Сэма Блекли, в которой он описывал, как учился этой игре [8]. Сэм привел в качестве примера своего приятеля, который задавал себе четыре вопроса перед тем, как поставить на доску камень.

1. Могу ли я обеспечить *жизнь* одной из моих групп? (Формированием двух глаз, соединением и так далее.)
2. Могу ли я *убить* одну из групп противника?
3. Могу ли я *защитить* свою территорию?
4. Могу ли я *вторгнуться* на территорию противника?

Эти четыре вопроса охватывают суть четырех главных приоритетов при игре в го: жить, убить, защитить, вторгнуться. Отвечая перед каждым ходом на вопросы, именно в этом порядке, вы сразу же существенно прибавляете в силе — обретаєте способность принимать разумные решения и избегать невыгодного обмена.

Это очень эффективный прием. Список из четырех приоритетов запомнить легко. Задавая себе несколько вопросов перед каждым ходом и выбирая наиболее ценные ходы, вы приводите свои действия в соответствие с истинными целями.

Го требует принятия решений в условиях неполной информации и постоянно изменяющейся обстановки, так что этот прием необычайно полезен. Подобные вопросы позволяют структурировать неопределенную ситуацию, привести в нее логику. Более того, «вопросы на миллион» можно сформулировать и применить и для других сфер вашей жизни. Вполне возможно, что в ситуации, которая требует принятия решений на основе неполной информации, расстановки приоритетов и согласия на компромиссы, несколько простых и очевидных «вопросов на миллион» помогут принять правильное решение.

20-часовая отметка

Судя по моим записям, через месяц занятий я достиг 20-часовой отметки. На какой стадии я нахожусь и каковы мои успехи?

Как подсказывает мне статистика программы SmartGo, я решил 150 ранжированных задач и дошел до ранга 18-го кю. Я сыграл 33 партии и в настоящее время имею фору в 4 камня перед искусственным интеллектом.

Я также проштудировал первый том «Ранжированных задач го» из серии издательства Kseido с рангом от 30-го до 25-го кю и добрался до середины второго тома, в котором собраны задачи с рангом от 25-го до 15-го кю. Следовательно, мой ранг можно оценить как 18–20-й кю. Неплохо.

Применение метода

Давайте кратко опишем суть метода, который я использовал для обучения игре в го.

- Я выучил основные правила игры.
- Я купил гобан, камни и несколько книг по стратегии го для

начинающих.

- Я нашел несколько способов изучения главных стратегических принципов, решая ранжированные задачи го на компьютере и в книгах.

- На первом этапе я посвятил большую часть практики решению ранжированных задач, возвращаясь к упражнениям, которые я не мог выполнить или объяснить с первой попытки.

- Я решал задачи и играл партии в го, пока не достиг 20-часовой отметки. На этом этапе я без труда справлялся с задачами ранга 20-го кю — прогресс в 15 кю по сравнению с рангом новичка, составляющим 35-й кю.

Планы на будущее

В отношении го я испытываю смешанные чувства. Я узнал об игре достаточно, чтобы оценить ее глубину и понять, как далеко мне до настоящего мастерства. Я существенно продвинулся, но для того, чтобы стать действительно хорошим игроком, потребуетея пройти *очень* длинный путь.

С другой стороны, свободного времени у меня немного, а го, похоже, требует такой же сосредоточенности и преданности, как программирование. Го может приносить удовольствие, но в конце длинного дня эта игра слишком похожа на *работу*.

Кроме того, партии на доске 19 на 19 могут продолжаться больше часа, а при игре с реальными людьми не очень вежливо прерывать игру, если того требуют семейные обязанности. Поэтому до сих пор бо2льшую часть партий я сыграл против компьютера, поскольку он всегда под рукой и не возражает, если я ненадолго отвлекусь.

Я оценил красоту и сложность задач го, поэтому буду и дальше решать их, когда мне захочется посвятить свободное время головоломкам. Задачи го мне нравятся гораздо больше кроссвордов, sudoku и других подобных развлечений, так что программы для го останутся в моем телефоне, готовые прийти на помощь, когда у меня выдастся свободная минутка.

Я собираюсь научить игре в го Лелу, когда она немного подрастет. Правила достаточно просты и понятны даже четырехлетнему ребенку, а я с удовольствием вспоминаю, как в детстве играл в шахматы с отцом. Го — превосходное средство для усвоения таких понятий, как анализ, стратегия и компромисс, и я думаю, что в последующие годы мы с дочерью получим огромное удовольствие от игры.

А в целом у меня нет сильного желания тратить время на совершенствование своей игры. Я узнал о го достаточно, чтобы удовлетворить любопытство, а совершенствование навыка не входит в число моих приоритетов.

Это абсолютно нормально. Не существует универсального правила, требующего достигать вершин мастерства во всем, чему вы учитесь. Жизнь полна компромиссов, и нет ничего дурного, если вы познакомитесь с чем-то новым, изучите его, а затем решите переключиться на что-нибудь другое.

Чтобы радоваться жизни, не обязательно быть обладателем черного пояса во всем, чем вы занимаетесь.

8

Укулеле (гавайская гитара) Урок: уединение, практика, повторение

Сегодня, как и во всякий другой день, ты просыпаешься среди пустоты и страха. Не открывай двери кабинета, чтобы сесть за книги. Возьми в руки музыкальный инструмент. Пусть красота, в которую ты влюблен, станет твоим делом. Есть сотни способов преклонить колени и поцеловать землю.

Руми, персидский поэт XIII века

Несколько лет назад я прочел книги Тома Ходжкинсона «Как бездельничать» («How to Be Idle») (2005) и «Манифест свободы» («The Freedom Manifesto») (2007), которые заставили меня задуматься. Идея Ходжкинсона состоит в том, что мы слишком много работаем и нам было бы очень полезно расслабиться. Вместо того чтобы вкалывать из последних сил, мы должны научить себя расслабляться, получать удовольствие и не воспринимать все всерьез.

Эти книги призывают отказаться от идеологии потребления и карьеры. Вместо стремления к материальным излишествам Ходжкинсон рекомендует удовлетвориться скромными запросами и вернуться к занятиям той эпохи, когда жизнь была проще: садоводству, кулинарии, музыке.

В «Манифесте свободы» Ходжкинсон советует научиться игре на гавайской гитаре:

Это четырехструнное чудо очень дешевое, очень портативное, и на нем очень легко играть. Поэтому она еще проще, чем обычная гитара. Для большинства песен нужно выучить всего три аккорда: C F G.

Заведите себе укулеле, и вы больше никогда не будете скучать.

Совет Ходжкинсона мне пришелся по душе. В то время мы с Келси жили в 340-метровой студии в нью-йоркском районе Верхний Ист-Сайд, оба много работали, стремясь сделать карьеру, и платили безумную манхэттенскую арендную плату. Я бы с удовольствием отвлекся, а освоение игры на гавайской гитаре показалось мне делом, достойным внимания.

В старших классах я был членом всех музыкальных коллективов, которые только существовали в школе. Я уже говорил, что научился играть на трубе, но, кроме того, я пел в хоре, и мне это очень нравилось. Однако, поступив в колледж, я перестал музицировать. Книга Ходжкинсона напомнила мне, какое место раньше занимала музыка в моей жизни.

Проблема отчасти состояла в том, что труба слишком *громкая*, а играть на

ней одному не очень интересно. Я был занят другими делами и не хотел искать оркестр, к которому можно присоединиться. Кроме того, невозможно играть на трубе и одновременно петь — как в случае с гитарой или пианино.

После недолгих размышлений я купил гавайскую гитару Washburn Oscar Schmidt OU [5] Concert Ukulele. Она изготовлена из растущего на Гавайях дерева коа, прекрасно звучит и имеет вполне разумную цену — за гитару с футляром я выложил около 175 долларов.

Распаковав доставленную укулеле, я немного пощипал струны, но этим дело и ограничилось — раньше мне не приходилось играть на струнных инструментах, если не считать короткого эксперимента с гитарой в колледже. Я не знал, с чего начать, и вскоре меня увлекли другие идеи, а гавайская гитара отправилась в кладовку — собирать пыль в обществе уже лежавшей там обычной гитары.

Герой укулеле

Интерес к игре на укулеле снова пробудился у меня после рождения Лелы. Просто удивительно, как дети мгновенно меняют ваши приоритеты. Неожиданно для себя я стал меньше работать, больше времени проводить дома и снова вспомнил о музыке.

Один из моих близких друзей, Нейт Зиберт, каждый вечер перед сном играет на гитаре своим сыновьям, Джексону и Финли. Мальчикам это очень нравится, и за много лет музыка превратилась в устоявшийся, успокаивающий ритуал.

Кроме того, музыка полезна для детей: доказано, что раннее знакомство с ней способствует развитию когнитивных способностей. Точно не известно, как именно музыка помогает в развитии мозга, но сам этот факт неоспорим [1].

Мне хочется, чтобы Лела выросла в доме, где игра на музыкальных инструментах считается обычным делом, и я готов сделать все от меня зависящее, чтобы моя девочка была умной и счастливой. Поэтому обучение игре на гавайской гитаре показалось мне очень разумным времяпрепровождением.

Мой новый инструмент

Гавайская гитара у меня уже была, но мое внимание привлек другой инструмент. Оказалось, что кузен Келси, Эрик Смит, самостоятельно освоивший столярное ремесло, занимается изготовлением музыкальных инструментов. Компания Эрика, Crow Hill Guitars [2], специализируется на

акустических и электрических гитарах ручной работы, и Эрик выставляет их на музыкальных фестивалях по всей Америке.

Эрик решил испытать себя, изготовив свою первую укулеле-тенор. Знакомьтесь, это Grizzly.



Эта гитара — единственная в своем роде: корпус из африканского красного дерева и венге, с инкрустацией из испанского кедра, индийского палисандра, секвойи и клена. В гриф вмонтирована перламутровая раковина

«медвежья лапа», а резонаторное отверстие украшено «морскими ушками». Я влюбился в Grizzly, как только ее увидел.

Но тут имелось одно препятствие: инструменты Эрика стоят не одну тысячу долларов. В смысле качества они оправдывают свою цену, но достаточно ли серьезно мое решение учиться играть на укулеле, чтобы такие затраты оправдались?

Купить Grizzly я решил по двум причинам. Во-первых, даже новичку игра на инструменте высшего класса дает психологическое преимущество: звук в любом случае лучше. По сравнению с Grizzly моя старая Oscar Schmidt звучит глухо, и играть на Grizzly гораздо приятнее.

Во-вторых, мне будет очень стыдно иметь такой замечательный инструмент дома и не играть на нем. Кроме того, если я не научусь играть, то впустую потрачу солидную сумму денег. Вложение немалых средств станет для меня хорошим стимулом — эта покупка представляет собой яркий пример *предварительной настройки* с целью изменения поведения.

Дополнительные принадлежности

Помимо Grizzly, я приобрел еще два необходимых инструмента:

- Snark SN-6 Ukulele Tuner — недорогой электронный тюнер, он крепится на головке грифа, облегчая настройку каждой струны;
- D’Addario J71 Pro-Arte Ukulele Strings, Tenor — нейлоновые струны сверхсильного натяжения для классической гитары, придающие более глубокий звук и облегчающие быстрый перебор [3]. Если струна порвется, ее придется заменить, так что полезно на всякий случай иметь в своем распоряжении несколько наборов струн.

Кроме того, я купил три книги, посвященные игре на гавайской гитаре:

- «Укулеле для новичков» («Absolute Beginners Ukulele») Стивена Спроута, (2010), пособие для начинающих;
- «Аппликатура для укулеле» («Fretboard Roadmaps — Ukulele») Джима Белофф и Фреда Соколофф, (2006), в которой описываются сложные приемы игры;
- «Укулеле каждый день» («The Daily Ukulele») Джима Белофф и Лиз Соколофф (2012), справочник, содержащий аккорды широкого спектра песен — от фольклора до классического рока.

Увлекательная задача

Вскоре после покупки Grizzly я получил по электронной почте письмо от своего друга Криса Гильбо, автора книг «Стартап за \$100»[13] («The \$100 Startup») и «Искусство нонконформизма»[14] («Art of Non-Conformity»). Каждый год Крис организует любопытную конференцию World Domination Summit. Представьте себе больше тысячи людей, каждый из которых делает безумные/странные/интересные вещи в одной и той же сфере, — это и есть WDS.

Крис изначально приглашал меня поговорить о книге «Сам себе MBA», но за 10 дней до начала конференции один из приглашенных внезапно отменил свое выступление. Крис знал, что я работаю над этой книгой, и поэтому обратился ко мне с предложением: не соглашусь ли я провести следующий семинар о быстром приобретении навыка?

Конечно... Почему бы и нет?

На World Domination Summit я впервые представлю свои идеи на публике.

Я составлял план презентации, посвященной идеям, которые изложены в главах 1–3 книги, когда получил по электронной почте еще одно письмо от Криса — такие письма он рассылает всем участникам перед каждой конференцией. Помимо всего прочего, там содержался следующий совет: «Если вы умеете что-то особенное, продемонстрируйте это... Все, чем вы сможете удивить [участников], привлечет внимание и станет залогом успеха».

И я подумал... что игра на гавайской гитаре станет превосходной демонстрацией эффективности моих идей. Но смогу ли я овладеть инструментом за 10 дней?

Это будет нелегко, но мне кажется, что я справлюсь. В ответе Крису я написал, что принимаю приглашение.

Теперь у меня всего 10 дней, чтобы научиться играть на гавайской гитаре.

Я от тебя без ума

Поскольку до первого публичного выступления оставалось всего 10 дней, мне требовался план, причем быстро. Это обязательство я никак не могу отложить, и впервые за 12 лет мне предстоит музицировать на публике.

После того как я принял предложение, бунт подняла та часть моего мозга, которая желает, чтобы я предстал перед людьми в самом выгодном свете. О чем я думал? Я что, с ума сошел?

Сама по себе безумная идея осложнялась тем, что помощи мне будет ждать неоткуда: на сцене только я и Grizzly. Меня не поддержит оркестр, и если

я ошибусь, все это сразу услышат. Если у меня ничего не получится, публика увидит, как ужасно я играю.

Одна лишь мысль о подобном исходе заставляла меня нервничать. Я собираюсь играть на музыкальном инструменте, который никогда раньше не держал в руках, и, возможно, петь перед незнакомыми людьми. Невероятно!

Тем больше причин разобраться, во что я ввязался, прежде чем выходить на сцену. Мне предстоит много работы, а часы уже тикают.

Что такое успех

Первая часть целевого уровня мастерства уже сформулирована: через 10 дней я собираюсь играть перед публикой на гавайской гитаре. Начало неплохое, но как мне конкретизировать это требование?

Я взял блокнот и набросал план своего выступления. Всего у меня будет час времени. Если полчаса я отведу на рассказ о главных идеях, то остается 20 минут на музыку и 10 минут на вопросы.

То есть мне придется играть примерно 20 минут. На чем я бы хотел сосредоточиться?

Мое выступление не концерт. Это больше похоже на лекцию, и поэтому исполнение музыки должно лишь иллюстрировать интересные и важные идеи, касающиеся быстрого приобретения навыка. Кроме того, было бы неплохо продемонстрировать что-нибудь необычное в музыкальном плане.

Пока я составлял план выступления, мне в голову пришли кое-какие мысли... Это будет *весело*.

«Песня четырех аккордов»

Размышляя о том, что сыграть на публике, я вспомнил забавный видеоклип с выступлением одной группы, который мне попался несколько месяцев назад.

Axis of Awesome [4] — это австралийская пародийная рок-группа из Сиднея. Больше всего она известна своей «Песней четырех аккордов», попури из прошлых и современных хитов поп-музыки, количество просмотров которой на YouTube превысило 24 миллиона.

Смысл шутки вот в чем: музыканты демонстрируют, что бо2льшая часть популярных песен нескольких последних десятилетий строится на одних и тех же четырех аккордах. Повторяйте эти аккорды снова и снова, положите на них какие-нибудь стихи, и получите надежный хит.

Иммунитета нет ни у одного артиста: тут Элтон Джон, The Beatles, Джон

Денвер, U2, The Red Hot Chili Peppers, Боб Марли, Бейонсе, Леди Гага. И даже почтенная австралийская народная песня «Вальсирующая Матильда». Попурри длится пять минут, и каждые 5 или 10 секунд в нем появляется новая песня, сопровождаемая смехом и аплодисментами зрителей.

«Песня четырех аккордов» веселая и одновременно шокирующая. Демонстрируя основу хорошо известных мелодий, она показывает, что у популярной музыки действительно есть определенная формула. Существует множество вариаций одной и той же темы, но человеческий мозг, похоже, склоняется именно к этому методу создания запоминающейся мелодии.

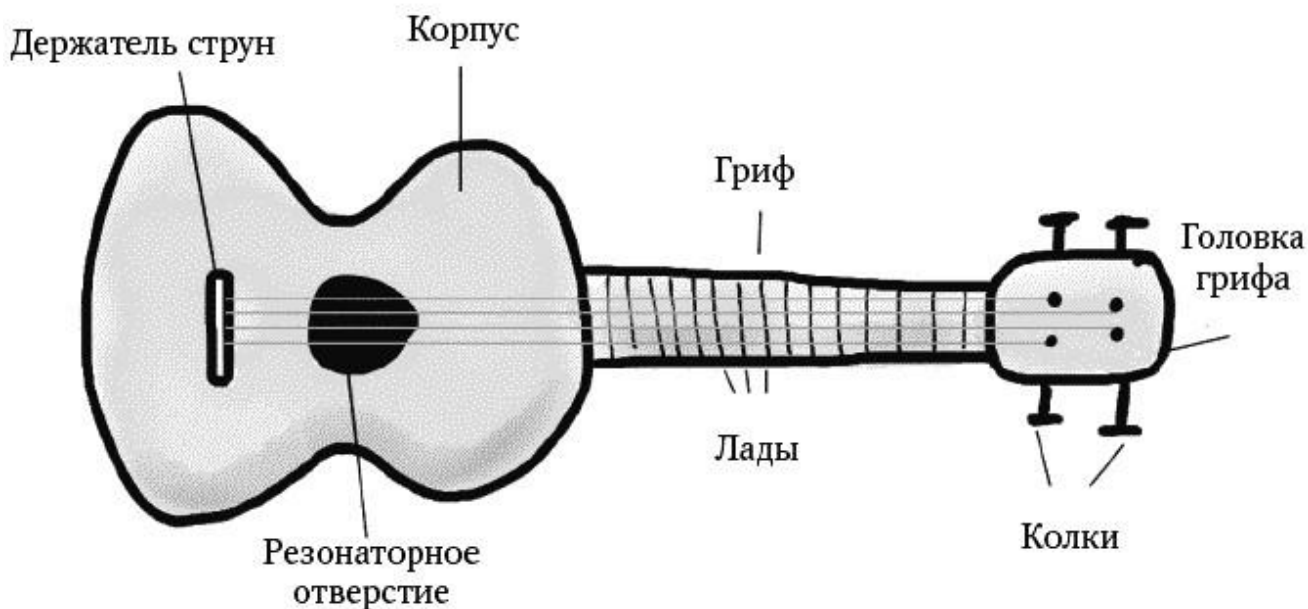
Вот эти четыре аккорда:

G / D / Em / C

Я собираюсь выучить «Песню четырех аккордов». Если музыканты правы, то я одновременно выучу все когда-либо написанные песни в жанре поп-музыки. Похоже, очень эффективное начало!

Выяснилось, что эти четыре аккорда очень легко играть, если знать, как устроен и работает инструмент. Давайте рассмотрим укулеле, чтобы понять, с чем мы имеем дело.

Анатомия гавайской гитары



У гавайской гитары четыре струны, натянутые вдоль корпуса инструмента. Длинная часть, которая отходит от корпуса укулеле, называется *грифом*.

На конце грифа укреплена деревянная *головка грифа*. В нее вставлены *колки*, которые удерживают верхние концы всех четырех струн. Струны проходят от головки над грифом, мимо открытого *резонаторного отверстия* к *держателю*, который расположен у основания корпуса. Держатель удерживает струны, туго натянутые над резонаторным отверстием.

Обычно на гавайской гитаре играют так: левая рука обхватывает гриф, а правая прижимает инструмент к груди. Музыкант перебирает струны пальцами правой руки (обычно указательным и или большим), а пальцы левой руки прижимают струны к грифу, чтобы изменить тональность их звучания.

Настройка

Каждая струна имеет свой *строй* — высоту звука, которую она издает, если ее не прижимать к грифу. Звучание струн настраивается поворотом соответствующих колков. Если поворотом колка струна натягивается, звук становится выше. Если поворотом колка струна ослабляется, звук становится

ниже.

Вот наиболее распространенный строй укулеле-тенора [5]:

- 4-я струна (нижняя) = G (196 Гц);
- 3-я струна = C (261,6 Гц);
- 2-я струна = E (329,6 Гц);
- 1-я струна (верхняя) = A (440 Гц).

Настроить гавайскую гитару очень легко, если у вас хороший инструмент. Тут в дело вступает тюнер Snark SN-9. Если вы прикрепите его к грифу, включите и дернете за струну, тюнер измерит частоту, преобразует ее в ноту и покажет, если она чуть выше или чуть ниже необходимого. Когда вы настроите нужную частоту колебаний струны, загорится световой сигнал, значит, можно переходить к следующей струне. Когда вы настроите все струны, можно приступать к игре.

Правильная настройка инструмента очень важна: если струны настроены неверно, гавайская гитара звучит так ужасно, что это слышат даже люди без музыкального слуха. Чем точнее настройка, тем лучше звучит инструмент.

Ноты и аккорды

Если вы хотите извлечь из укулеле одну ноту, достаточно ударить по струне, которая начинает вибрировать над резонаторным отверстием корпуса, создавая звук. Если вы прижмете струну к грифу, она временно станет короче, и при ударе по ней частота вибрации увеличится. Чем выше частота вибрации, тем выше будет звук.

Интересной особенностью укулеле и других струнных инструментов, таких как гитара и фортепиано, является возможность одновременного звучания нескольких нот. Если музыкант правой рукой ударит по всем струнам укулеле, то сможет одновременно извлечь четыре ноты. Опытный музыкант умеет прижимать струны к грифу таким образом, что издаваемые ими звуки гармонируют друг с другом, образуя *аккорд*.

Именно благодаря аккордам у струнных инструментов такой богатый, живой звук. Извлекая не отдельные ноты, а их сочетания, можно одновременно создавать мелодию и гармонию. В каком-то смысле аккорды позволяют сопровождать самому себе.

Еще одно преимущество струнных инструментов, в отличие от трубы, кларнета, флейты или саксофона, заключается в том, что музыкант использует только руки. При наличии опыта можно одновременно брать аккорды и петь, поэтому такие инструменты, как гитара и фортепиано, очень популярны у

певцов.

Извлечение аккордов

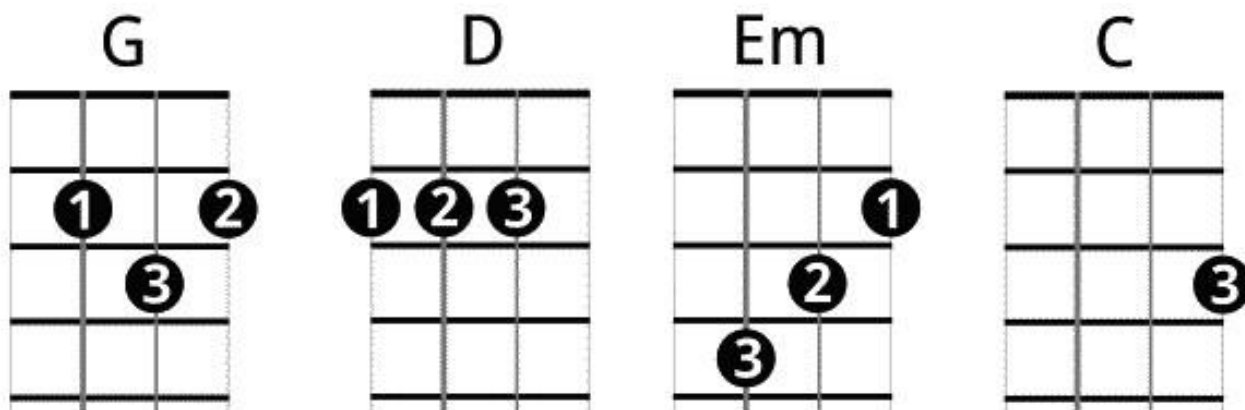
Поскольку я начинаю обучение с нуля, мне нужно сделать выбор: учиться играть отдельные ноты, читая партитуру, или разучивать аккорды. Поскольку аккорды мне незнакомы и я хочу одновременно играть и петь, то я склоняюсь ко второму варианту.

Обозначение расстановки пальцев, прижимающих струны к грифу, называется аппликатурой. Если вы прижмете струны так, как указывает аппликатура, получится аккорд.

Вот четыре аккорда в «Песне четырех аккордов»:

G / D / Em / C

А так они выглядят на укулеле.



Если держать укулеле вертикально, то гриф будет выглядеть так, как показано на рисунках. Жирная верхняя линия — это верхний край грифа укулеле, который называется *верхним порожком*. Каждая тонкая горизонтальная линия — это ладовый порожек, металлическая планка, немного выступающая над поверхностью грифа.

Ладовые порожки способствуют чистому звучанию струн, когда вы прижимаете их к грифу. Пальцы прижимают струны не к самим порожкам, а между ними. При этом струна одновременно прижимается к ближайшему нижнему ладовому порожку, укорачиваясь до строго определенной длины, и частота ее колебаний соответственно меняется.

Аппликатура указывает, какие струны и в каких местах необходимо прижать, чтобы получился тот или иной аккорд. Струны на них изображены

четырьмя горизонтальными линиями: слева располагается 4-я струна, верхняя, когда вы держите укулеле в положении для игры. Крайняя справа — 1-я струна, или нижняя.

Черные кружки обозначают места, где прижимаются струны, а цифры — пальцы, которыми они прижимаются.

1 = указательный палец

2 = средний палец

3 = безымянный палец

4 = мизинец

Как указывает аппликатура, чтобы сыграть аккорд G, я должен прижать три нижние струны указательным, средним и безымянным пальцами. Указательный палец должен прижимать третью струну над вторым ладом, средний палец — первую струну в том же месте, а безымянный палец — вторую струну над третьим ладом.

Описывать это довольно долго, но сделать несложно. После того как вы поймете, какой палец куда ставится, аккорды превращаются в обычный моторный навык: научиться соответствующим образом ставить пальцы на гриф и прижимать струны, чтобы получить нужный аккорд.

Для тренировки полезно иметь перед глазами аппликатуру наиболее распространенных аккордов. Недолгий поиск в интернете привел меня к «Обязательным аккордам для киви укулеле» («Kiwi Ukulele Indispensable Chord Chart») [6], загружаемому руководству по аккордам на укулеле. Руководство создано Майком Дикисоном, автором книги «Киви укулеле: новозеландская разновидность укулеле» («Kiwi Ukulele: The New Zealand Ukulele Companion») (2008).

Источников, в которых можно найти аппликатуру для укулеле, очень много — практически каждая книга по игре на гавайской гитаре содержит аппликатуры аккордов. Уникальным сборник Майка делает то обстоятельство, что аккорды в нем расположены по степени популярности. Самые распространенные, как в «Песне четырех аккордов», приведены первыми. Редкие аккорды, например G#sus4, располагаются в самом конце, поскольку почти не используются.

Такая разновидность частотного анализа чрезвычайно полезна: степенной закон применим и к музыке. Чтобы исполнять песни, не нужно знать сотни аккордов. Для большинства песен достаточно 10 или 12 аккордов.

Аккорды

Я взял аппликатуру «Песни четырех аккордов» и потратил какое-то время на то, чтобы запомнить расположение пальцев. Прижав пальцы к грифу в

нужных местах, я ударил одновременно по всем четырем струнам. Получилось: оказывается, я могу извлекать из инструмента красиво звучащие аккорды!

Самая большая трудность на начальном этапе — прижать каждую струну так, чтобы лад выполнил свою работу. Если прижать струну недостаточно сильно, чистого звука не получится. Вместо него вы услышите неприятное дребезжание, сопровождающее нужную ноту. Это отвлекает.

Запомнив четыре аккорда, я потратил один час на их тренировку, снова и снова повторяя последовательность:

G... D... Em... C...

G... D... Em... C...

G... D... Em... C...

Это первый этап приобретения моторного навыка: помочь мозгу ассоциировать определенные мысли, такие как названия аккордов, с движениями мышц левой руки. Позже я смогу играть аккорд Em, когда пожелаю, но для этого я должен научиться правильно ставить пальцы, достаточно сильно прижимать струны к грифу и переходить от одного аккорда к другому.

Я занимаюсь перед сном — та же стратегия, что и когда переучивался печатать вслепую. Это помогает мозгу с максимальной эффективностью консолидировать в памяти нужные движения.

Удары по струнам

На следующий день я уже хорошо исполняю все четыре аккорда и свободно перехожу от одного к другому. Совсем неплохо для одного часа занятий. Теперь, когда я получил представление о том, что делает левая рука, нужно сосредоточиться на правой.

В то время как пальцы левой руки прижимают струны, формируя аккорды, правая рука ударяет по струнам, извлекая звук. До сих пор я ударял по струнам в очень простой последовательности: считал до четырех и ударял на каждый счет.

1 2 3 4

1 2 3 4

1 2 3 4

Большинство песен, если их записать на нотной бумаге, имеют размер 4/4. Не углубляясь в детали, можно сказать, что песни в 4/4 имеют четыре ударные доли.

Услышав в следующий раз какую-нибудь песню, попробуйте мысленно сосчитать до четырех, подчиняясь ритму. Скорее всего, вы заметите, что последовательности совпадут.

Разумеется, если бы каждая песня состояла из нот или аккордов, повторяющихся в простой и предсказуемой последовательности, музыка наскучила бы нам очень быстро. Чтобы разнообразить мелодию, музыканты намеренно нарушают простой ритм с помощью так называемых *синкоп*.

Синкопой называют отступление от ритмической последовательности. Выглядит это так. Попробуйте ладонью одной руки отбивать простейший ритм на какой-либо плоской поверхности, например на письменном или обеденном столе. Удары должны быть сильными и равномерными. Одновременно ведите счет от одного до четырех.

1 2 3 4

1 2 3 4

1 2 3 4

Очень просто, правда?

Теперь усложняем. Второй ладонью отбивайте ритм *между* ударами первой.

1 и 2 и 3 и 4 и

1 и 2 и 3 и 4 и

1 и 2 и 3 и 4 и

Это простая синкопа. Удары на счет «и» выпадают из стандартного размера, что вносит разнообразие и делает ритм более интересным.

Опытные музыканты много экспериментируют с синкопами. Каждый ритм можно разбить на множество частей. Нередко ритм делится на четыре части, и отсчет ведется следующим образом:

1e&a 2e&a 3e&a 4e&a

Звучит это так: «один-е-и-а, два-е-и-а, три-е-и-а, четыре-е-и-а», где «е» — это длинное «и-и», а «а» — короткое «а».

Подчеркивая, или акцентируя одни такты и вообще пропуская другие, музыкант может создать самые разнообразные ритмы, делающие песню интересной.

Знать это очень важно, потому что за ритмом на укулеле следит правая рука. Если я хочу научиться чему-то более сложному, чем довольно скучные удары в такт, то должен потратить какое-то время, чтобы найти интересную последовательность ударов.

Немного поэкспериментировав, я остановился на такой последовательности:

1 &a e&a 3 &a e&a

Мне понадобилось некоторое время, чтобы натренировать правую руку на этот ритм, но в конечном итоге все у меня получилось. Сначала во время удара по струнам я брал простой аккорд С. Привыкнув, я стал перебирать последовательность из четырех аккордов, меняя их на первом и третьем такте:

G D

1 &a e&a 3 &a e&a

Em C

1 &a e&a 3 &a e&a

На этом этапе я уже исполняю настоящую музыку: левой рукой перебираю аккорды в нужном порядке, а правой ударяю по струнам, поддерживая ритм.

Впервые в жизни я играю песню на укулеле. Просто, но эффективно!

Я снова и снова повторяю эту последовательность. Привыкнув, закрываю глаза и пытаюсь играть, не глядя на гриф и струны. Ошибок становится больше, но я все же могу играть и возвращаться назад, чтобы исправить ошибки.

Неплохо — всего после двух часов занятий.

Петь и играть одновременно

Теперь, освоив аккорды и научившись поддерживать ритм, я должен освоить еще один навык — петь и играть одновременно.

«Песня четырех аккордов» смешна потому, что ее составляющие легко узнаваемы. Без текста создается впечатление, что я снова и снова повторяю одно и то же. Конечно, «фишка» именно в этом, однако без слов песня не имеет смысла.

За долгие годы работы в бизнесе я познакомился со многими интересными людьми. Среди самых интересных — Дерек Сиверс, основатель CDbaby.com. Дерек — хороший певец и гитарист, поэтому я обратился к нему за советом, как действовать дальше. Вот его рекомендация:

Выучи песню, сначала просто напевая ее без музыки. Важно отделить знание инструмента от знания песни. Ты должен уметь спеть всю песню, не берясь за инструмент. (Качество не имеет значения: пой в голос, мурлычь себе под нос, насвистывай — все что угодно.)

Выучив песню, затем пой названия нот вместо стихов. Закрой глаза и запоминай песню в таком виде.

И, наконец, добавляй инструмент — одновременно играй и пой ноты.

Это превосходный пример разбиения: сложный процесс разбивается на более простые части. Чтобы петь и играть одновременно, необходимо хорошо знать слова и мелодию песни, без труда воспроизводя их, когда берете аккорды и бьете по струнам. Отделив на первом этапе песню от инструмента, вы обеспечиваете себе возможность твердо ее выучить и не слишком задумываться, когда добавите инструмент.

Замена слов названием нот помогает почувствовать момент смены аккордов во время игры. А поскольку у стихов есть свой ритм, вы можете использовать и их, чтобы запомнить, когда нужно перейти к другому аккорду.

Тоже облегчение!

И наконец, когда вы поете названия нот синхронно с аккордами и ударами по струнам, то помогаете мозгу соединить все вместе. Если у вас все получилось, то перейти от названия нот к стихам будет просто — ведь вы их уже запомнили.

Я последовал совету Дерика и начал учить слова «Песни четырех аккордов», слушая ее снова и снова, одновременно запоминая мелодию. Затем записал слова в блокнот и пытался воспроизвести их по памяти.

Потом я стал играть песню на укулеле, насвистывая мелодию и мысленно повторяя названия аккордов. Оказалось, что момент смены аккордов определить совсем несложно, и я начал одновременно играть и насвистывать.

Перейти к словам оказалось проще, чем я думал. Если я забывал слова или путал аккорд, то останавливался, возвращался назад и пробовал снова.

После пяти часов занятий я уже мог полностью сыграть и спеть свою первую песню. Местами получалось не очень складно, но это уже песня.

Осталось пять дней до выступления. Я достаточно неплохо справляюсь, играя самому себе, но что будет на публике?

Доведение до автоматизма

Чтобы убедиться в своей способности выступить в назначенный день и час, оставшееся время я посвящаю репетициям. Занимаюсь я всегда непосредственно перед сном. Моя цель проста: я хочу переходить от одного аккорда к другому, поддерживать ритм ударов по струнам и петь, прикладывая как можно меньше умственных усилий.

Этот этап занятий не особенно увлекателен, но очень важен. Пока все мои действия составляли когнитивную фазу приобретения моторного навыка: анализ, разбиение и эксперимент. Теперь мне требуется перейти к ассоциативной фазе и позволить своим мышцам действовать самостоятельно, без участия сознания.

Каждый день я играю и пою эту песню, снова и снова — уже сбился со счета. Переход от аккорда к аккорду дается легче, ритм ударов по струнам стал устойчивым, и я выучил почти все слова. Несколько переходов между песнями оказались довольно трудными, и я тренирую их снова и снова.

Неожиданно оказывается, что пора ехать в Портленд. Но поеду я не один: на этот раз со мной будет Grizzly.

Пора на сцену

Мое выступление должно состояться в последний день конференции. Хорошей новостью можно считать тот факт, что это расписание дает мне как минимум еще два дня тренировки. Но есть и плохая новость — два лишних дня переживаний.

В день выступления я настраиваю Grizzly и в последний раз репетирую — целый час. Затем беру в руки укулеле и выхожу на сцену.

Я не знал, сколько у меня будет слушателей, и поэтому испытываю облегчение, увидев, что помещение относительно невелико. Сомнительно, что мне удалось бы завладеть вниманием большого зала.

Места постепенно заполняются, и вскоре их уже не хватает. В комнате 40 стульев, но народу пришло больше, многим пришлось выстроиться вдоль стен. Все ждут.

Рассказав о своих идеях и методе, я беру укулеле, и публика явно приходит в волнение. Я изо всех сил пытаюсь не показать своего страха.

Была не была. Я начинаю играть и петь.

Получается вполне прилично — гораздо лучше, чем я думал. Конечно, это не назовешь выступлением, достойным «Карнеги-холла», но я доиграл до конца, не перепутав аккорды, не сбившись с ритма, не забыв слов. Слушатели улыбались, отбивали такт ногами, смеялись словам песни, а в конце аплодировали.

Мои усилия не пропали даром: я смотрел не на гриф или струны, а на зрителей и получал удовольствие от выступления.

Гитарный перебор

Закончив петь, я продемонстрировал аудитории еще один трюк, который разучил во время занятий: песни кажутся сложнее, если последовательно перебирать струны, а не ударять по всем одновременно. Аккорды при этом не меняются.

Эта техника называется *перебором*, и существует много вариантов такого исполнения. Последовательность, которую продемонстрировал я, очень проста: сначала 1-я струна, затем 4-я, затем 2-я и, наконец, 3-я, на ударной доле. Если повторять эту последовательность правой рукой и одновременно брать аккорды левой, то каждая следующая нота естественным образом дополняет предыдущую. В результате получается красивое и сложное звучание, хотя на самом деле это не сложнее, чем удары по струнам.

Мое выступление имело успех и, кроме того, доставило мне

удовольствие. Я рад, что решил рискнуть, что заставил себя упорно заниматься. Перейти от полного неумения до первого выступления на публике всего за 10 часов практики — это совсем неплохо.

Один-четыре-пять

На конференции, рядом с главной аудиторией, я случайно познакомился с еще одним участником, Мелиссой Динвидл. Актриса по профессии, Мелисса захватила с собой в поездку укулеле, так что мы взяли инструменты и немного поимпровизировали.

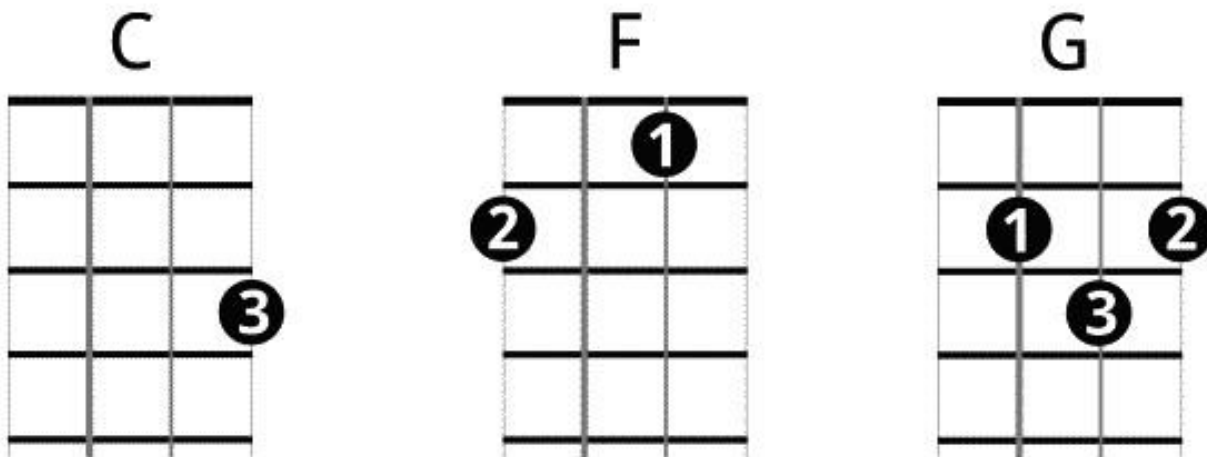
В тот день Мелисса познакомила меня с песенной структурой под названием «12-тактовый блюз», основой которой служит последовательность аккордов 1–4–5.

Не углубляясь в теорию музыки, скажем, что у каждой ноты есть ассоциирующийся с ней аккорд, а также «семейство» аккордов, дополняющих основные. Если вы умеете играть основной аккорд, а также аккорды соответствующего семейства, значит, вы умеете играть тысячи песен.

Возьмем, к примеру, детскую колыбельную песенку «Мерцай, мерцай, звездочка» («Twinkle Twinkle Little Star»). Всю песню можно сыграть с помощью трех аккордов, в такой последовательности:

C / F / G

Вот их аппликатура.



В данном случае C — основной, или I аккорд. F и G относятся к тому же семейству. F — IV аккорд, а G — V аккорд.

Используя эти три аккорда, можно сыграть много шуточных песенок,

например:

- «Песенку Алфавит»;
- «Песенку Фрости-снеговика»;
- «Песенку Элмо».

Это значит, что дома я могу научиться играть детские песни для Лелы, а также кое-что из блюза.

Остальные 10 часов занятий посвящены разучиванию новых песен, которые мне по силам сыграть. Эти часы быстро проходят за штудированием «Укулеле каждый день» (2010), освоением новых способов удара по струнам из «Укулеле для новичков» (2009), а также изучения основ музыкальной теории по «Аппликатуре для укулеле» (2006).

Применение метода

Давайте кратко опишем суть метода, который я использовал для обучения игре на укулеле.

- Я приобрел укулеле, струны, тюнер и другие необходимые принадлежности.
- Я решил сосредоточиться на освоении самых популярных последовательностей аккордов и ударов по струнам, что позволило мне играть тысячи популярных песен.
- Научившись брать аккорды и ударять по струнам, не глядя на гриф, я тренировался, пока не научился поддерживать ритм ударов и менять аккорды, не задумываясь.
- Освоив основные аккорды и удары по струнам, я начал разучивать песни, мурлыча их во время игры, запоминая, в каких местах текста меняются аккорды, а затем добавил слова — пока не научился петь и играть одновременно.
- Я экспериментировал с приемом перебора струн вместо ударов при сохранении аккордов неизменными, что позволило играть более сложные песни.
- Я познакомился с последовательностью аккордов I, IV, V, что помогло мне разучить другие песни.

Планы на будущее

Мне нравится играть на гавайской гитаре. Изучая песенник для укулеле, разыскивая новые песни в интернете, я смогу понять, как играть практически любую понравившуюся мне песню. Одни аккорды сложнее, другие легче, а вариантов ударов по струнам и переборов существует не одна тысяча, и на их изучение мне понадобится уйма времени.

Одно из преимуществ игры на укулеле состоит в том, что это занятие не требует напряжения сил. Если не считать первого эксперимента, я играю не для того, чтобы подготовиться к выступлению, и не потому, что хочу стать профессиональным музыкантом. Когда я беру в руки инструмент, то могу просто расслабиться, немного побренчать, разучить новую песню, технику перебора или удара по струнам.

Лела уже достаточно взрослая, чтобы запоминать песни, и тут пригодилась моя старая, запыленная акустическая гитара Oscar Schmidt — теперь она «укулеле» Лелы, вторая гитара служит либо приманкой, пока я репетирую, либо нашим основным инструментом, если дочь хочет сидеть у меня на коленях и бить по струнам, пока я перебираю аккорды. У Лелы уже есть свои любимые песни, и она просит поиграть на «укуяе» перед сном.

Это здорово, когда в нашей жизни есть музыка!

9

Виндсерфинг

Урок: окружающая среда имеет значение

Наше оружие — не плоть и кровь, а эпоксидная смола, углеволокно, однослойная пленка, а также наше эго, мечты и несокрушимая воля.

Крис Зейтвотгель, виндсерфер

Я посреди озера, отчаянно пытаюсь поднять парус. Это трудно: мышцы рук, спины и ног буквально заходятся от боли.

Сильный северо-западный ветер гонит маленькие волны с белыми гребешками, которые подбрасывают подо мной доску. Я позволяю парусу вновь опуститься в воду и делаю паузу, чтобы отдохнуть и приноровиться к раскачивающейся доске.

Я смотрю на запад — приближается буря, затягивая горы черными облаками. Пора возвращаться.

Я наклоняюсь, беру стартшкот и тяну за него, поднимая парус из воды — наверное, 30-й раз за последние 30 минут. Надеюсь, что смогу вернуться до того, как поднимется ветер.

Буря приближается... Я не очень хорошо управляюсь с этой странной штуковиной, но должен научиться. Грести к берегу — это уже не актуально.

Жизнь на воде

У меня остались очень приятные воспоминания о летних месяцах, проведенных на воде. В подростковом возрасте я активно участвовал в бойскаутском движении и несколько раз проводил лето в качестве вожатого в двух лагерях на севере Огайо — в Camp Firelands и Camp Avery Hand.

В последний раз я удостоился чести работать помощником директора по водным видам спорта под началом легендарного Боба Слини. Боб, как обычно, вел занятия в бассейне: плавание, спасение на воде и заплыв на милю. В моем ведении были всевозможные плавсредства: байдарки, академическая гребля, парусный спорт и моторные лодки.

Я наслаждался каждой минутой, прожитой в лагере. Каждый день я бо2льшую часть времени проводил на воде, обучая юных скаутов правилам безопасной навигации. Работа сложная и утомительная — но приносящая удовлетворение.

Уехав учиться в колледж, я на время расстался с водой. Моя первая работа в Цинциннати отнимала очень много времени, а река Огайо, ближайшая к городу водная артерия, не очень подходила для прогулок на веслах. Позже, когда мы с Келси поселились в самом центре Нью-Йорка, единственное место, где мы могли сесть в лодку, — это гондолы в Центральном парке. После того как мы обосновались в Колорадо, я почувствовал, что с водой придется распрощаться навсегда: эта часть планеты не может похвастаться обилием водных пространств.

Однако недавно у меня появилась возможность приезжать на маленькое частное озеро к северу от города, в нескольких милях от моего дома. Я ужасно рад: впервые за *много лет* я смогу провести время на воде.

Вирус виндсерфинга

Первой мне в голову пришла мысль об академической гребле — озеро достаточно велико и подходит для двухвесельной шлюпки. К сожалению, стандартные гребные шлюпки чрезвычайно громоздки, и даже самая маленькая из них очень длинная и относительно хрупкая, что очень неудобно, поскольку у меня нет возможности хранить большое или длинное судно у озера.

Байдарка легче и дешевле, но плавать на ней в одиночку довольно скучно: длина большинства байдарок позволяет разместиться в них вдвоем. Моторные лодки и скутеры исключаются, поскольку владельцы озера ограничивают мощность мотора девятью лошадиными силами, и вообще я предпочитаю средства передвижения на мускульной тяге.

Изучая разные варианты, я наткнулся на видеосъемки Джеймса Дугласа, специалиста по биологии моря, профессора Университета Флорида Галф Кост. Джим не только известный биолог, но и опытный виндсерфер, который в своем блоге [1] публикует ценную информацию об этом виде спорта.

На кадрах видео Джим готовит к спуску виндсерф класса Formula с водозащищенной HD-видеокамерой, прикрепленной к концу гика и снимающей все происходящее [2]. Потом он *летит* над водой, прыгает на небольших волнах, выполняет быстрые, эффектные повороты.

Меня зацепило, и я просмотрел это видео несколько раз подряд. Раньше я никогда не вставал на виндсерф, и мне вдруг захотелось попробовать.

Виндсерфинг отвечает всем моим требованиям: я могу заниматься им в одиночку, а снаряжение для него не слишком большое и громоздкое. Кроме того, — насколько мне известно — это не очень дорогостоящее занятие. Естественно, придется купить необходимое оснащение, но базовый комплект меня не разорит.

Похоже, виндсерфинг может доставить удовольствие, но прежде чем

принимать решение, я должен убедиться в безопасности этого вида спорта. По своему характеру я не адреналиновый наркоман и не любитель острых ощущений. Если виндсерфинг окажется опасным занятием, я с удовольствием займусь чем-нибудь другим.

Если я собираюсь встать на виндсерф, то должен *заранее* представлять возможные риски. Что может случиться?

Опасен ли виндсерфинг?

Виндсерфинг часто называют «экстремальным» спортом, однако степень экстремальности зависит от того, какой разновидностью виндсерфинга вы хотите заняться. На спокойных, закрытых, маленьких озерах, похожих на то, где я собрался учиться, это опаснее, чем управлять маленьким яликом, но совсем не похоже на безумство.

Иное дело — виндсерфинг в океане. Океан, конечно, добавляет увлекательные элементы, такие как прыжки на волнах, но и несет с собой значительный риск. Сильный прибой существенно затрудняет выход на воду, а потеря оснастки вдали от берега может стать серьезной проблемой. По сравнению с океаном виндсерфинг на внутриматериковом озере относительно безопасен.

Тем не менее, где бы вы ни занимались виндсерфингом, вас подстерегают опасности, самые серьезные из которых — риск утонуть и переохлаждение. Избежать их никак нельзя. Если не подготовиться к тому, что в ветреный день вам придется стоять на пластиковой доске посреди большого водного пространства, рукой удерживая большой парус в вертикальном положении, неприятности неминуемы.

У риска утонуть две главные причины: травма и усталость. Если падение приведет к серьезной травме руки или ноги либо к контузии, то существует опасность потерять сознание в воде. Усталость может помешать взобраться на доску после падения или вернуться к берегу, что тоже повышает риск утонуть.

Лучшие меры безопасности — это 1) надевать спасательный жилет и 2) всегда выходить на воду в присутствии других людей, на воде или на берегу. Если вы попадете в беду, вам обязательно помогут [3].

Не менее опасно переохлаждение. При низкой температуре, высокой влажности и сильном ветре организм очень быстро охлаждается. Если температура тела опускается ниже 35 °С, начинают отключаться сердце, легкие и нервная система. Если температура не восстановится до нормы, в интервале от 35 до 38 °С, человека ждет летальный исход.

Переохлаждение особенно опасно тем, что развивается постепенно. По мере падения температуры тела появляются такие симптомы, как дрожь,

спутанность сознания, потеря координации. Переохлаждение, которое наступает незаметно, является вашим главным врагом — ведь для возвращения на берег нужны физическая сила и трезвый расчет.

При температуре воды выше 18 °С риск переохлаждения не очень велик. Если же вода холоднее, то прежде чем спускаться на воду, необходимо утеплиться, надев гидрокостюм.

Гидрокостюмы изготавливают из тонких и эластичных водоотталкивающих материалов, таких как неопрен. Современные гидрокостюмы различаются по толщине (она измеряется в миллиметрах), причем на туловище ткань более плотная, чем на руках и ногах. Такая конструкция преследует две цели: усиленная изоляция туловища эффективнее предотвращает потерю тепла, а тонкий материал на руках и ногах не стесняет движений. Для виндсерфинга в холодную погоду нужна и термоизоляция, и свобода движений.

Два гидрокостюма обеспечат вам защиту при самой разной погоде. Облегченный вариант, оставляющий открытыми руки и ноги, подходит для теплой погоды: 3 мм защиты на туловище и 2 мм на руках и ногах (3/2) вполне достаточно. При температуре воды ниже 18 °С лучше всего пользоваться полным гидрокостюмом 5/4, с ботинками, перчатками и неопреновым капюшоном или шапочкой [4].

У любителей виндсерфинга большой популярностью пользуется побережье Калифорнии, поэтому Департамент судоходства и водных путей штата выпустил краткую памятку по безопасности [5].

1. Учитывайте прогноз погоды и волнение моря.
2. Обязательно сообщите кому-нибудь, где вы собираетесь плавать и когда планируете вернуться.
3. Оденьтесь по погоде.
4. Наденьте сертифицированный спасательный жилет, снабженный свистком.
5. В жаркую, влажную, солнечную погоду пейте больше воды.
6. Проверьте снаряжение на признаки повреждения и износа.
7. Не плавайте в одиночку.
8. Не плавайте при ветре, дующем с берега.
9. Холод может убить. При появлении дрожи немедленно возвращайтесь на берег, чтобы согреться.
10. Всегда оставайтесь рядом с доской — не пытайтесь добраться до берега вплавь.

Кроме того, Департамент предлагает простой список необходимых действий перед выходом на воду.

Перед спуском...

1. Тщательно проверьте страховочную стропу.
2. Берегитесь темных облаков на горизонте — шторм приходит быстро.
3. Если у вас есть сомнения, не выходите на воду.
4. Опытный моряк всегда попытается выбрать самый безопасный образ действий, прежде чем прибегать к экстренным мерам.

Это просто здравый смысл, но помнить о нем очень важно. Виндсерфинг — это здорово. В отличие от смерти. Однако при должной подготовке и планировании главные опасности виндсерфинга можно минимизировать.

С чего начинать?

Как вы можете догадаться, север штата Колорадо не относится к самым популярным местам для виндсерфинга. Ветер — если вы найдете место для катания — тут дует часто, но в отличие от других мест, таких как долина реки Колумбия, он непостоянен и непредсказуем, а также не очень силен.

Кроме того, на севере Колорадо не так много доступных водных пространств. Отроги Передового хребта — это высокогорная и полупустынная степь, здесь нет больших естественных водоемов. Большинство «озер» тут представляют собой искусственные резервуары, предназначенные для сбора воды на нужды местных фермеров и питающиеся небольшими реками, которые несут в долину воду от растаявших снегов.

В результате местные природные условия прекрасно подходят для сплава на байдарках, но никак не для виндсерфинга. Я не знаю ни одного человека, который регулярно занимается здесь виндсерфингом, и, похоже, в местных магазинах не продаются доски и другое снаряжение для этого вида спорта. Попытки найти инструкторов для виндсерфинга на севере Колорадо выявили, что ими, кажется, интересуюсь я один. Ближайшее место, где я могу брать уроки, — это Денвер, в часе езды от моего дома. При таком расстоянии возникают трудности с расписанием: подходящий ветер дует не каждый день, и запланированные занятия, возможно, придется отменять.

Конечно, было бы идеальным начать обучение с персональным инструктором, но работа и семейные обязанности пока не позволяют мне надолго уезжать из дома. Я не сомневаюсь, что при соответствующем снаряжении смогу самостоятельно выйти на воду и освоить технику виндсерфинга.

Но тут возникает одно препятствие: где купить снаряжение, если поблизости нет специализированных магазинов? Насколько мне известно, доски

и паруса слишком велики, и их не доставляют такие почтовые службы, как FedEx и UPS.

Если я не смогу купить все необходимое, этот проект закроется раньше, чем успеет начаться.

Необходимое снаряжение

Первым источником информации для меня стал сайт Джима, на котором есть огромное количество постов с советами для начинающих. Одна из первых статей, которую я прочел, называлась «Ответы на 18 главных вопросов о виндсерфинге» («Top 18 Windsurfing Questions Answered») [6]. Она содержала и рекомендации, касающиеся необходимого снаряжения.

Судя по всему, минимальный комплект состоит из доски, паруса, спасательного жилета и гидрокостюма. Я прочел каждый пост о виндсерфинге на сайте Джима, составил список из предметов, казавшихся мне полезными, попытался сократить каждый пункт до двух вариантов, а затем отправил Джиму письмо по электронной почте, поблагодарив за информацию и обратившись за советом.

Самое главное — определить тип доски и паруса. Видео Джима с полетом на виндсерфе класса Formula, конечно, впечатляет, но я уверен, что начинать с такой доски — не самая лучшая идея. Еще один вариант — доска под названием Rio, выпускаемая компанией Starboard. Считается, что закругленная Rio прекрасно подходит для новичков. Кроме того, это универсальная доска: на ней вы не будете летать, как на самых продвинутых досках, но она подходит практически для всех погодных условий.

Я не знал, как поступить, и обратился за советом к Джиму:

Я абсолютный новичок в виндсерфинге — еще ни разу не становился на доску. Очень надеюсь, вы посоветуете мне, с чего начинать.

Я планирую купить полный комплект снаряжения и пытаюсь выбрать между досками Starboard Rio Medium и Starboard Formula 167.

Похоже, Rio больше подходит для начинающих, но я хочу научиться [скользить по воде] как можно быстрее. Я буду кататься только на озерах и при несильном ветре (7–9 км/ч). Насколько мне известно, доски Formula более быстрые и больше подходят для слабого ветра.

Вот мой вопрос: не безумие ли учиться на доске Formula? Я легко переношу первые неудачи и предпочитаю купить одну доску, а не несколько. Если я начну с Formula, не будет ли слишком велик риск поломки снаряжения или травмы?

Буду благодарен за любой совет — ваша помощь неоценима!

Приблизительно через час пришел ответ от Джима.

1. Утверждение, что доски Formula предпочтительнее при слабом ветре, неверно. Действительно, они легче глиссируют при слабом ветре (13–15 км/ч), чем любые другие доски, однако, чтобы реализовать их потенциал, нужно быть опытным виндсерфером и уметь управляться с хлопающим парусом площадью 11–12 квадратных метров. А если доска Formula HE глиссирует, управлять ею очень сложно, поскольку она короткая, широкая и не имеет шверта. При водоизмещающем режиме (реально при ветре менее 18 км/ч) более длинная и узкая доска с килем (как Rio M) гораздо быстрее и проще в управлении, чем Formula.

2. Маловероятно, что вы получите травму, обучаясь на доске Formula, если только не сорвете спину, пытаясь поднять большой парус. Причина такой безопасности проста — вам вряд ли вообще удастся привести доску в движение! Тем не менее вы можете повредить доску, потому что Formula имеет тонкое покрытие, которое легко пробивается, если во время падения уронить снасти.

3. Вероятно, Rio M — наилучший выбор для быстрого обучения, чтобы освоить глиссирование и привыкнуть к ножным петлям. Можно также остановиться на GO 171.

4. Я бы на вашем месте приобрел несколько парусов разного размера, чтобы расширить диапазон тяги при любой силе ветра. Для начального этапа и для сильного ветра понадобится парус площадью меньше 6 квадратных метров... У меня на сайте есть специальный калькулятор, который подскажет, какой парус вам нужен для глиссирования при данной силе ветра [7].

5. Наибольший шанс глиссирования у вас — это небольшой парус и сильный ветер, поскольку маленький парус при сильном ветре требует меньше умения, чем большой парус при слабом ветре.

Это бесценная информация. Джим развеял некоторые неверные представления относительно типа доски, с которой нужно начинать, а также типа паруса. Любезность Джима, поделившегося своими знаниями, позволила мне сэкономить несколько тысяч долларов и избавила от разочарования.

Джим также рассказал мне об Isthmus Sailboards [8], магазине в городе Мэдисон, который продает снаряжение для виндсерфинга в интернете. Я позвонил в Isthmus, и Гэри Стоун, один из его совладельцев, помог мне составить список всего, что потребуется новичку.

Вот каким получился мой заказ:

- доска Starboard Rio M;
- парус Chinook Powerglide 4,7 кв. м;
- гик Chinook Sport AL;
- мачта Chinook 400 см;
- удлинитель мачты Chinook US;
- нижняя часть мачты Chinook US;

- натяжной стартшкот Chinook;
- полный гидрокостюм Mystic Crossfire 5/4 + неопреновые ботинки, перчатки и шапочка;
- облегченный гидрокостюм Mystic Crossfire 3/2 + неопреновые туфли Vibram Five Fingers;
- спасательный жилет Dakine Surface.

Общие затраты составили около трех тысяч долларов, включая доставку. Виндсерфинг — дорогое удовольствие, но если я буду должным образом ухаживать за снаряжением, оно прослужит долго.

Сначала я склонялся к тому, чтобы купить парус большего размера, но Гэри уже 20 лет обучает людей виндсерфингу, и ему не составило труда меня отговорить. Большие паруса прекрасно подходят для слабого ветра, но только если вы умеете с ними обращаться.

Недостаток же их состоит в том, что они тяжелые и их труднее поднимать из воды. Если я начну с большого паруса, до конца не освоив искусство балансировки и маневра, то, по словам Гэри, «прокляну свою жизнь».

Проклинать свою жизнь мне совсем не хочется, и по совету Гэри я выбираю парус меньшего размера. Еще одно доказательство того, что перед тем, как принимать решение, есть смысл посоветоваться с опытными инструкторами. Поговорив полчаса с Гэри, я разместил свой заказ в Isthmus.

Выяснилось, что коммерческие компании, занимающиеся перевозкой грузов, без проблем доставляют предметы размером с виндсерф в любую точку страны, так что получить снаряжение было легче, чем я думал. Гэри предположил, что доставка займет около недели. Тем временем я занялся теорией.

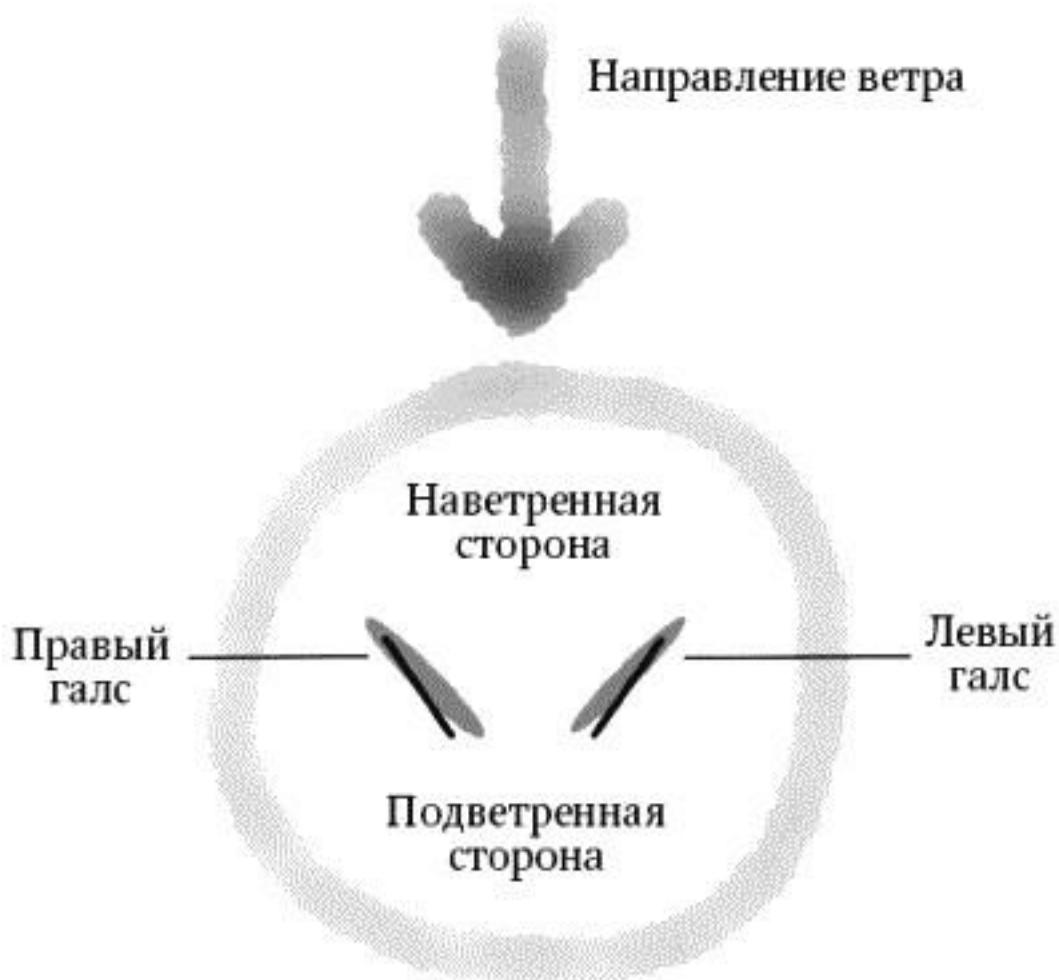
Стой, салага!

Немного побродив по интернету, я нашел несколько источников информации для новичков:

- «Дух и искусство виндсерфинга: пособие для начинающих» («A Beginner's Guide to Zen and the Art of Windsurfing») Фрэнка Фокса (1988);
- «Виндсерфинг» («Windsurfing») Питера Харта (2005);
- «Как освоить виндсерфинг за выходные» («Learn Windsurfing in a Weekend») Фила Джонса (1992);
- «От новичка к победителю» («Beginner to Winner») (DVD) Джема Холла (2006).

В них довольно подробно излагаются терминология, теория и основы

техники управления виндсерфом. Вводный DVD Джема Холла особенно понятно объясняет, как управлять доской на воде: иногда сложные манипуляции легче объяснить, продемонстрировав на практике. Книжки тут плохие помощники, потому что даже самые лучшие иллюстрации не идут ни в какое сравнение с подробным обучающим видео.



Мое внимание сразу же привлекла одна особенность: в виндсерфинге используется традиционная морская терминология. Вполне логично, поскольку виндсерф — это судно, в котором корпус заменен доской. Но я все равно удивился. Виндсерферов называют «моряками», а термины, используемые в пособиях, взяты непосредственно из парусного спорта.

Прошло много лет с тех пор, как я ходил под парусом, и мне требовалось кое-что вспомнить. Если вы не знакомы с терминологией, то обучающие пособия могут раздражать, поэтому важно знать основные понятия. Вот некоторые из них.

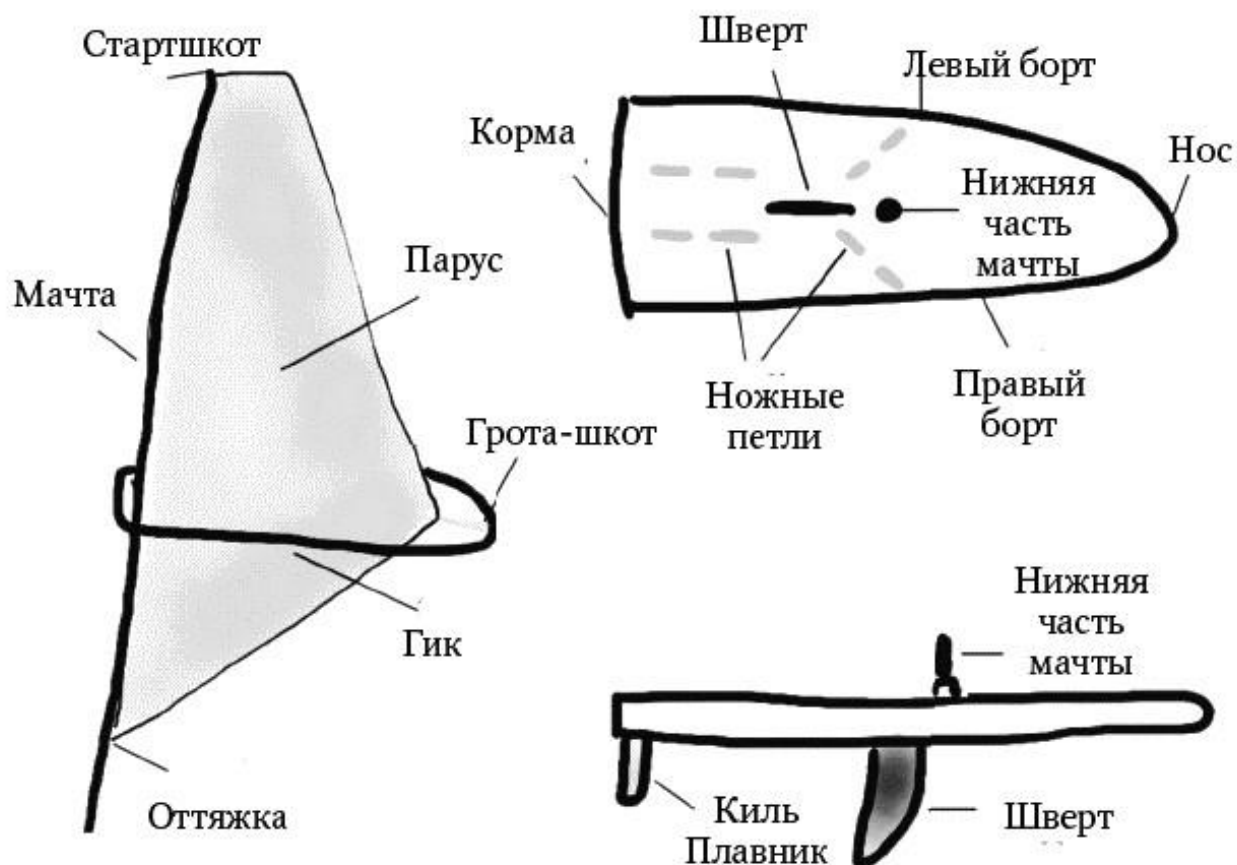
- **Направление относительно ветра** — такие направления, как север, юг,

восток и запад в парусном спорте не имеют особого смысла, поскольку ветер меняется. Направление движения указывается по отношению к направлению ветра: в *наветренную сторону* означает движение навстречу ветру, а в *подветренную сторону* означает движение по ветру.

- **Направление относительно судна** — аналогичным образом «лево» и «право» указываются относительно судна.

- **Вперед и назад** — передняя часть судна называется носом, задняя — кормой. Соответственно указываются и направления к носу или к корме.

- **Поворот** — для обозначения поворота используются два термина: *оверштаг* и *фордевинд*. Разница между ними определяется тем, какой конец доски во время поворота пересекает ветер. Если вы идете против ветра (в наветренную сторону) и поворачиваете, это оверштаг. Если вы идете по ветру (в подветренную сторону) и поворачиваете, это фордевинд.



Соединение этих терминов вызывает сложности. Если вы идете против ветра и поворачиваете нос судна вправо (на правый борт), то вы идете «левым галсом», поскольку ветер дует в левый борт судна.

Это сбивает с толку. Чтобы расшифровать инструкции в книге и представить это на воде, потребовалось несколько часов. Такой процесс мысленной симуляции окажется полезным, когда я выйду на воду: представляя, как те или иные понятия соотносятся с реальным судном, я быстрее все вспомню, когда потребуется.

Каждая часть виндсерфа имеет свое название.

- **Доска** — мы уже знаем, что такое нос, корма, правый и левый борт, но этого недостаточно. У виндсерфа есть и другие конструктивные части, такие как большой стабилизатор (*шверт*) в центре и маленький стабилизатор (*киль*, или *плавник*) на корме, которые погружены в воду. На поверхности доски находится место для крепления мачты (*нижняя часть мачты*), которая крепится механическим основанием (*основанием мачты*). Обычно на поверхности доски также расположены ножные петли, которые помогают виндсерферу сохранять равновесие и оставаться в правильном положении, когда виндсерф заскользит по воде.

- **Парус** — паруса виндсерфа имеют треугольную форму и крепятся к доске с помощью большого вертикального шеста, который носит название *мачты*. Мачта вставляется в специальный карман паруса (*мачткарман*), который тянется от нижней части до верхушки (*топа*). После того как мачта установлена, перпендикулярно к ней специальным зажимом крепится горизонтальная опора (*гик*). Три стороны паруса привязываются к мачте и гика нейлоновой веревкой (*тросом*), которая оборачивается вокруг конца гика, а затем продевается через специальное крепежное устройство под названием *утка*. Когда парус полностью снаряжен, мачту крепят к основанию, и виндсерф готов к выходу на воду.

Правильное снаряжение виндсерфа требует внимательности и опыта. Различают несколько способов крепления основания мачты к доске, и каждый вариант изменяет центр тяжести виндсерфа. В зависимости от типа доски также различаются способы крепления кормового плавника и ножных петель, что также влияет на поведение доски на воде.

Аналогичным образом, способ вооружения паруса оказывает огромное влияние на управление виндсерфом. Две основные переменные — это *оттяжка галсового угла* и *оттяжка шкотового угла*, которые определяют, насколько туго натянуты тросы, крепящие парус к нижней части мачты и к дальнему концу гика. Уменьшение оттяжки галсового и шкотового угла увеличивает «карман» паруса — и это значит, что парус лучше ловит ветер и создает большую тягу. Увеличение оттяжки галсового и шкотового угла уменьшает размер кармана, ослабляя тягу паруса и усиливая его управляемость при сильном ветре.

Займемся физикой

Парус — удивительное устройство. Вопреки распространенному убеждению, виндсерф (да и любое парусное судно) движется не потому, что ветер толкает парус. На самом деле все несколько сложнее.

Принцип действия паруса основан на создании разницы давления в передней и задней частях. В большинстве ситуаций воздух обтекает парус, и поток создает область низкого давления впереди (со стороны носа судна) и область высокого давления сзади (со стороны кормы). Совместно воздействие двух этих зон с разным давлением создает силу, которая перемещает судно в направлении области низкого давления.

В результате ветер не только толкает, но и *тянет* парус. Точно так же работает крыло самолета.

Это важно знать, когда вы вышли на воду и пытаетесь привести виндсерф в движение. Если ваша умозрительная модель работы паруса имеет вид «максимизировать площадь паруса, чтобы в него дул ветер», то вам будет трудно увеличить мощность паруса, а также плыть в любом направлении, отличном от того, куда дует ветер.

Всю эту информацию нелегко отслеживать, но я рад, что у меня есть время изучить основы теории, прежде чем я попытаюсь перейти к практике. Виндсерфинг по большей части относится к моторному навыку, но если я не понимаю принцип движения судна, то рискую потратить много времени на тренировку неверных движений.

Держим нос по ветру

Я не только читал книги, но и стал обращать более пристальное внимание на погоду. До того как заинтересоваться виндсерфингом, я не обращал внимания на ветер — разве что когда он был необыкновенно сильным. Теперь я обнаружил, что постоянно слежу за ветром, внимательно разглядывая верхушки деревьев, чтобы оценить его скорость.

Тут мне помогают современные технологии: такие сайты, как Weather Underground [9], Windfinder [10] и iWindsurf [11], собирают сведения о ветре со всего мира, что значительно облегчает получение информации о скорости и направлении ветра в любом месте.

Естественно, значение имеет лишь тот ветер, который дует над водным пространством, где вы собрались прокатиться на виндсерфе. Чтобы измерять скорость и направление ветра, я купил устройство под названием Kestral 3000 Pocket Wind Meter [12]. Этот удобный маленький прибор позволяет мне

измерить скорость ветра на озере точнее тех значений, которые я могу найти в интернете. Прибор также измеряет температуру воздуха и воды, так что я не буду выходить на воду, если риск переохлаждения слишком велик.

Наблюдая за ветром несколько дней подряд, я стал замечать определенные закономерности. На озере по утрам обычно дует легкий восточный ветер, постепенно он усиливается, а после обеда, обычно около четырех часов пополудни, сменяется на северо-северо-западный. Поскольку я буду выходить на воду с восточного берега озера, лучше делать это в послеобеденное время. Тренировки в четыре часа дня вполне вписываются в мой график: если правильно спланировать рабочий день, я смогу отлучиться из дома на час или два, ничем особенно не жертвуя.

Не бывает виндсерфинга без ветра

Я также обратил внимание на то, что ветер на озере непредсказуем. День на день не приходится. Иногда ветер есть, иногда нет. Исходя из того, что я прочел в книгах, при ветре слабее 8–10 км/ч ставить парус не имеет смысла.

Это значит, что мне нужен резервный план. К счастью, у меня есть идея: на нашем озере я видел людей на паддл бордах. Это занятие мне кажется увлекательным и вполне подходящим. Если я добавлю к своей экипировке весла, то ветер уже не обязателен, и в безветренную погоду стоячая гребля станет хорошей альтернативой виндсерфингу.

У меня уже есть спасательный жилет, два гидрокостюма. Осталось приобрести лишь приличный паддл борд и длинное весло. После недолгих поисков я остановился на следующем снаряжении:

- паддл борд Ocean Kayak Nalu 11’;
- весло Quick Blade Kahana Elite 80’’.

Паддл борд доставили одновременно с виндсерфом. Имея виндсерф и паддл борд, я смогу проводить время на воде при любом ветре.

Сборка

Теперь, получив все снаряжение, я должен выяснить, как соединить доску и парус — этот процесс называется *вооружением* паруса.

К счастью, Isthmus меня подстраховал. Вооружение — это самая большая трудность для новичков, поскольку представляет собой сложный процесс, состоящий из нескольких этапов. Пытаясь выяснить, какой трос куда идет,

легко запутаться.

Компания Isthmus разрешила эту проблему, снабжая покупателей подробными видеоинструкциями, на которых профессионалы в режиме реального времени вооружают парус. Обучающее видео [13] идеально подходит для такого типа обучения: я несколько раз посмотрел инструкции и записал самое главное, прежде чем приступить к делу.

При помощи этих заметок я смог вооружить свой виндсерф примерно за 30 минут — не так плохо для первой попытки. Я в гидрокостюме, виндсерф готов, ветер дует. Пора в путь.

Первое испытание

Я несю доску и парус к кромке воды, присоединяю мачту к основанию, затем вхожу в воду по пояс, толкая перед собой виндсерф, — на такую глубину, чтобы опустить шверт. Северо-западный ветер, дующий со скоростью около 20 км/ч, создает на поверхности озера довольно заметную рябь. Но мне все равно: я приготовился.

Я забираюсь на доску, встаю, затем наклоняюсь, чтобы взять стартшкот, плетеный эластичный трос, прикрепленный к мачте. Потянув за стартшкот, я смогу поднять парус из воды, пока он не встанет вертикально, перпендикулярно доске.

По мере того как парус поднимается из воды, грота-шкот естественным образом, под напором ветра, поворачивается в подветренную сторону. Это позволяет мне поднять парус без дополнительных усилий, так что я остаюсь практически на том же месте, держась обеими руками за мачту. Это «нейтральное» положение является исходным для виндсерфинга, и я очень доволен собой. Пока все прекрасно.

Моя следующая задача — *выбрать* парус: снять одну руку с мачты, ухватить за гик и потянуть. Этим я усилю тягу паруса, и виндсерф придет в движение.

Боюсь, у меня ничего не выйдет...

На милость матери-природы

Я не был готов к тому, что случилось.

Я потянул к себе гик, и потом все произошло одновременно. Виндсерф заскользил по воде, причем быстрее, чем я ожидал. Доска качалась у меня под ногами, и я запаниковал и потерял равновесие. Центр тяжести моего тела сместился назад, и, не успев понять, что происходит, я оказался в воде.

Мои руки по-прежнему сжимали гик, и последнее, что я увидел, прежде чем погрузиться под воду, это мачта, которая стремительно опускалась мне на голову.

Я не говорил, что шлема у меня не было?

Не знаю, что я ожидал от падения с виндсерфа, но явно не таких сильных ощущений. К счастью, я не получил по голове, но у меня перехватило дыхание, я наглотался противной озерной воды. Оказавшись под водой, я сразу не мог понять, где поверхность, а где дно.

К счастью, под водой я пробыл совсем недолго — благодаря спасательному жилету вынырнул на поверхность, кашляя и отплевываясь. Потом обнаружил еще одну проблему: я ничего не вижу.

У меня слабое зрение, и я ношу очки. (Контактные линзы мне не подходят.) Когда я выходил на воду, у меня мелькнула мысль о том, чтобы закрепить их на затылке резинкой, чего я, конечно же не сделал. В результате при первом же падении они свалились с носа и утонули — отныне им суждено украшать дно озера.

Моя первая прогулка на виндсерфе продолжается всего пять минут, а я уже успел намокнуть, ослепнуть и испугаться, едва избежав контузии.

Не слишком обнадеживающее начало.

Стиснув зубы, я поплыл к виндсерфу. Пока я приходил в себя, ветер успел отогнать доску метра на четыре. Я ловил доску, забирался на нее и снова тянул за стартшкот.

Самоспасание

За 40 минут, проведенных в тот день на воде, я падал всеми мыслимыми способами — назад, вперед, вбок. Я до тошноты наглотался озерной воды, а мои руки, ноги и спина ныли от бесконечных попыток поднять парус.

В конце концов я решил, что хорошего понемножку. Я замерз, мои силы подходили к концу. Пора было возвращаться.

К сожалению, я не умею управлять этой проклятой доской, а каждая попытка поднять парус заканчивается тем, что я глотаю очередную порцию воды. Ветер сносит меня на юг, прочь от того места, где я входил в воду. И я не в состоянии направить виндсерф назад, даже если бы от этого зависела моя жизнь.

Наконец я прихожу к выводу, что пора применить на практике метод самоспасания, который описывался в одной из книг. Я ложусь на доску и располагаю парус так, чтобы он лежал на мне, топом к корме. Затем начинаю грести руками, постепенно приближаясь к берегу.

Доска движется очень медленно. Я не успел отойти далеко от берега, но

мне потребовалось 10 минут интенсивной гребли, чтобы вернуться назад. Руки — не самые эффективные весла даже в благоприятных обстоятельствах; а мои обстоятельства таковыми не назовешь.

Наконец я выхожу на берег, однако на этом мои испытания не заканчиваются. До точки старта довольно далеко, и поэтому мой первый выход на воду завершается «позорной прогулкой». Я заставил свое измученное тело отнести снаряжение к машине, приехал домой и рухнул без сил.

Посттравматический анализ событий

Отдохнув, я проанализировал свой первый опыт. Что случилось? В чем причина неудачи?

Во-первых, ветер был слишком сильным для новичка, в первый раз ставшего на виндсерф. Будь у меня хоть какой-то опыт, такой ветер не стал бы большой помехой, но для первого раза он оказался слишком силен. Урок усвоен.

Во-вторых, я вообще не умел удерживать равновесие на доске. Я не занимался ни серфингом, ни скейтбордом — ничем таким, что требовало балансировки на движущейся поверхности. Волны, которые поднимал ветер, раскачивали виндсерф, и это сбивало меня с толку.

В-третьих, тяга паруса смещает центр тяжести доски. И если не смещаться в противоположном направлении, чтобы компенсировать это воздействие, падение будет практически неизбежным. Если я сразу же не отпущу парус, почувствовав, что теряю над ним контроль, то скорее всего окажусь под водой. Чтобы не падать, мне нужно научиться перемещаться по доске в любом направлении, особенно при порывах ветра.

В-четвертых, у меня не хватает защитного снаряжения. Мне явно нужен шлем, который защитит голову от падающей мачты. Мне просто повезло, что я не получил травму. Эту ошибку я повторять не намерен — заказ на шлем немедленно отправляется в магазин.

Учимся на собственном опыте

Я могу предпринять кое-какие меры, чтобы моя вторая тренировка не стала копией первой.

Во-первых, можно выбрать не такой ветреный день. Наверное, это будет не так увлекательно, но зато позволит тренироваться.

Во-вторых, я могу попрактиковаться удерживать равновесие на доске, выйдя на ней без паруса, с одним веслом. Убрав парус, я почувствую, как

удерживать равновесие на доске, насколько далеко можно наклоняться в ту или иную сторону, не опрокидываясь в воду. Конечно, это не идеальная имитация, поскольку парус смещает центр тяжести доски, но все же лучше, чем постоянные падения.

В-третьих, свое следующее занятие с парусом я могу посвятить балансировке. Приложив должные усилия, я пойму, как движения паруса влияют на поведение доски. Научусь перемещать свое тело, чтобы компенсировать тягу паруса, почувствую, когда нужно подтянуть парус, чтобы не опрокинуться назад, а когда отпустить, если порыв ветра вдруг делает виндсерф неуправляемым.

Следующие несколько дней выдаются безветренными, и у меня есть возможность попрактиковаться в балансировке на доске. Опыт успешен: к концу занятия я обретаю уверенность, а чуть позже, в ветреные дни, у меня появляется шанс попрактиковаться и в удержании равновесия на подпрыгивающей на волнах доске. Мне все еще непривычно видеть у себя под ногами воду, но вскоре неуверенность проходит, и движение даже действует на меня успокаивающе.

Через несколько дней устанавливается солнечная и теплая погода с довольно свежим, но не слишком сильным ветром. Я вооружаю парус, спускаю виндсерф на воду и становлюсь на доску в нейтральном положении. Неужели повторится мое первое занятие?

Ничуть не бывало: за весь день я упал только два раза. Отдельная тренировка баланса очень помогла, и мне удавалось избежать падения, когда я чувствовал, что виндсерф выходит из-под контроля. В тот день мне пришлось много раз поднимать парус из воды, но это все же лучше, чем ее глотать. Я учился усиливать тягу паруса и вдруг почувствовал, что виндсерф заскользил по воде.

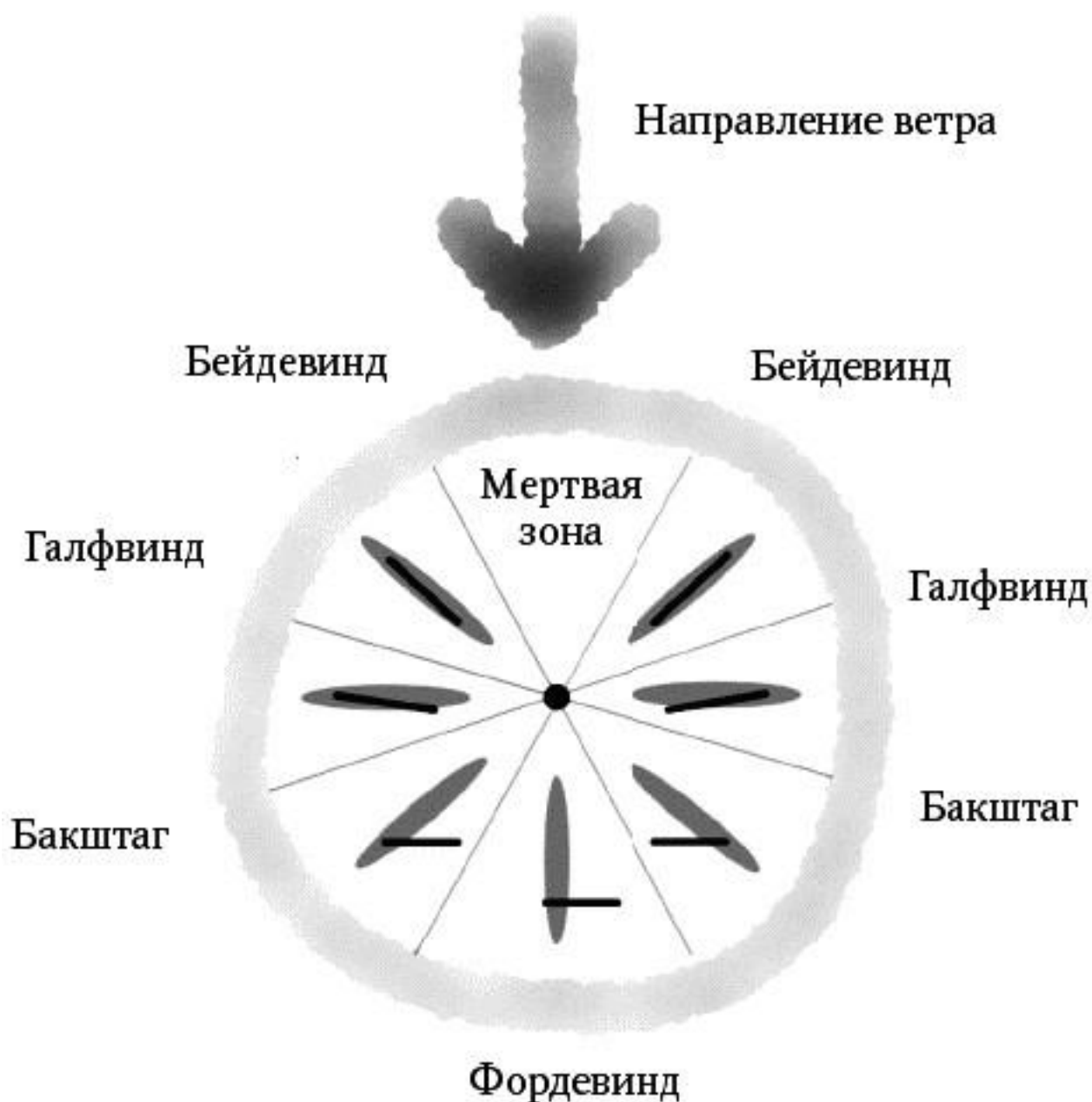
Как поворачивать эту штуку?

Привести в движение виндсерф и не упасть — это уже победа. Но торжествовать рано, поскольку появляется еще одна насущная проблема: повороты.

Я экспериментировал с приемами, о которых узнал из книг и DVD: поворачивать доску ногами, удерживая парус в нейтральном положении. Полезно также помнить, что виндсерф (и парусное судно вообще) не может двигаться прямо против ветра.

Существует мертвая зона, занимающая примерно 45 угловых градусов в направлении, с которого дует ветер. Попытавшись проложить курс в пределах этой зоны, вы обнаружите, что «зависли» — остановились или даже стали

двигаться назад. Чтобы плыть против ветра, нужно отвернуть доску от ветра как минимум на 45 градусов, выбрать парус, а через некоторое время повернуть оверштаг в другом направлении. Меняя галсы, вы можете плыть в нужном направлении, даже прямо против ветра.



Очень важно также уметь поворачивать виндсерф на ходу. Во время движения смещение паруса влево или вправо меняет *центр приложения силы*, фокальную точку всех сил, которые ветер создает в парусе, относительно *центра сопротивления дрейфу* — фокальной точки сил сопротивления, возникающих между доской и водой. При смещении паруса к корме, носу, правому борту или левому борту изменяется отношение между фокальными

точками сил ветра и сил сопротивления, и виндсерф поворачивает.

Взбираясь на доску после падения, вы обычно оказываетесь в неудобном положении. В идеале доска должна располагаться перпендикулярно ветру, а парус — по ветру. В таком случае его можно поднять без особого труда.

Если парус смотрит против ветра, то есть опасность, что гик ударит вас по лицу, когда вы поднимете парус и он наполнится ветром. Аналогичным образом, нос доски может повернуться против ветра, оказавшись в мертвой зоне.

Чтобы компенсировать эти нештатные ситуации, необходимо ногами повернуть доску так, чтобы она расположилась перпендикулярно парусу, независимо от направления ветра. Как только доска и парус расположатся перпендикулярно друг другу, можно медленно поднимать парус, позволяя ветру повернуть виндсерф в правильное положение. Когда парус станет по ветру, его можно поднять окончательно, а затем поворачивать в любом направлении [14].

После нескольких часов тренировок у меня начинает кое-что получаться. Поначалу не очень ловко, и мои повороты не назовешь изящными, но я уже могу вывести виндсерф на воду, доплыть до середины озера и вернуться приблизительно в то место, от которого отплыл. По сравнению с первым опытом это уже огромное достижение.

Всему свое время

В Колорадо, в районе Передового хребта, осень наступает внезапно и быстро. Обычно это происходит в конце сентября. Сегодня солнечно и тепло, а завтра уже минус один и вот-вот пойдет снег.

Каждый день я провожу на воде как можно больше времени, пытаюсь максимально использовать выпавший шанс, пока не испортилась погода. Температура воды остается постоянной, 20 градусов — не очень теплая, но без угрозы переохлаждения. При наличии гидрокостюма самым сильным является шок от первого падения — потом уже перестаешь обращать на это внимание.

В те дни, когда скорость ветра превышает 10 км/ч, я снаряжаю виндсерф. В безветренную погоду занимаюсь стоячей греблей.

Я чаще катаюсь на паддл борде, чем на виндсерфе: в отсутствие ветра трудно ходить под парусом. Это расстраивает, потому что мне хочется тренироваться как можно больше, но торговаться с матерью-природой бесполезно. Либо ветер достаточно сильный, и я могу выйти на воду, либо нет.

Однажды утром температура воздуха падает до 5 градусов. За три дня вода становится ледяной — всего 10 градусов. Мой гидрокостюм 5/4 помогает продержаться какое-то время, но риск переохладиться или утонуть меня совсем не радует. Похоже, сезон виндсерфинга для меня завершен.

Я суммирую время тренировок и выясняю, что не достиг своей цели: всего девять часов практики, значительно меньше тех двадцати, которые я собирался посвятить виндсерфингу. Гораздо больше времени я потратил на паддл бординг.

Важный урок: окружающая среда имеет значение. Конечно, я мог бы расстроиться, обнаружив, что не достиг поставленной цели, однако мое желание тренироваться не заставит ветер дуть сильнее. При всем при том проводить больше времени на воде было бы неразумно, поскольку усталость — один из основных факторов риска. В результате я мог тренироваться не больше 30 или 40 минут в день.

Несмотря на то что общее время занятий оказалось гораздо меньше 20 часов, я очень многому научился. Я уверенно снаряжаю парус, правильно его ставлю, знаю, как выходить на воду при разном ветре, а забираться на доску из воды тренировался столько раз, что невозможно даже сосчитать. Я умею поднимать парус, ложиться на выбранный курс и управлять доской на ходу, не опрокидываясь в воду. Я могу при необходимости повернуть виндсерф и вернуться к берегу примерно в том же месте, откуда выходил на воду. Это серьезный прогресс за такое короткое время.

А в качестве бонуса я приобрел еще один небольшой навык — паддл бординга. Занятие это гораздо более простое, чем виндсерфинг, но тоже увлекательное. Даже при сильном ветре я с удовольствием скольжу по воде на паддл борде, гоняя уток от одного конца озера до другого. Очень приятно кататься на доске с веслом на закате, любуясь пейзажем; это превосходный способ расслабиться.

Применение метода

Давайте кратко опишем суть метода, который я использовал для обучения виндсерфингу.

- Я приобрел необходимое снаряжение: доску, парус, гидрокостюм, а также спасательный жилет, шлем и другую защитную экипировку.
- Я научился снаряжать доску и парус, а также разбирать, транспортировать, хранить снаряжение и ухаживать за ним.
- Я научился защищать себя от главных опасностей, в том числе утопления, контузии и переохлаждения.
- Я научился спускать доску на воду с берега, поднимать парус в нейтральное положение и увеличивать его тягу, чтобы привести доску в движение.
- Я научился поворачивать виндсерф и регулировать положение паруса,

чтобы передвигаться при разной силе ветра.

- Я научился избегать падения с доски, а также снова взбираться на доску и поднимать парус после падения.

Планы на будущее

К тому времени, когда вы прочтете эти строки, сезон виндсерфинга откроется снова. Как только температура воды и воздуха позволит заниматься без риска переохлаждения, я возобновлю занятия, чтобы восстановить навыки оснащения виндсерфа, балансировки и маневрирования.

Эти навыки восстановятся довольно быстро, поэтому я готовлюсь взять следующую высоту: заставить доску *гиссировать*, что значительно увеличит скорость. Глиссирование — единственный возможный способ передвижения при сильном ветре, но высокая скорость увеличивает риск падения и травмы, поэтому я должен твердо знать основы, прежде чем предпринять такую попытку.

Кроме того, я собираюсь купить больший по размерам парус. Ветер на озере обычно слабый (и переменный), и для таких условий лучше подходит большой парус. Освоив парус площадью 4,7 квадратных метра, я начну тренироваться с парусом 7,5. Эти два паруса будут дополнять друг друга: большой нужен при слабом ветре, а меньший по размерам — при сильном.

В целом виндсерфинг доставляет мне огромное удовольствие. Мне приятно проводить время на воде, особенно теперь, когда я падаю довольно редко. Я с нетерпением жду начала следующего сезона и не пропущу ни одного ветреного дня, когда бы они ни случились.

Я не властен над матерью-природой, но могу управлять своими тренировками, когда возникнут благоприятные условия. Этого достаточно.

Послесловие

Похоже, успех связан с действием. Успешные люди продолжают двигаться вперед. Они делают ошибки, но не сдаются.

Конрад Хилтон, основатель сети отелей Hilton

Меньше чем за год я овладел шестью сложными навыками.

Я не гений и не чудо природы. У меня нет никаких особых талантов. Я продолжал работать. Я не бросал все и не уезжал на другой конец света. И не забывал о своей семье.

Просто каждый день я выделял около часа на занятия избранным делом, но эти занятия были разумными. Навыки, поначалу казавшиеся абсолютной загадкой, через несколько дней или даже часов становились понятными. Каждый навык требовал небольшой теоретической подготовки, а также около 20 часов сфокусированной, намеренной практики.

Более того, занятия превратились в повседневность: теперь эти навыки стали частью моей жизни. Освоение самых главных элементов навыка на первом этапе значительно ускоряет прогресс. Когда вы будете читать эти строки, мои умения в каждой из освоенных областей еще больше укрепятся. *Насколько больше, зависит от интенсивности занятий.*

Для приобретения нового навыка нужна практика. Другого способа не существует.

Вы можете подготовиться. Вы можете изучить теорию. Вы можете устранить препятствия и изменить окружающую обстановку, чтобы облегчить занятия. Вы можете найти разумные методы, повышающие эффективность и результативность занятий. Но в конечном итоге от практики никуда не деться.

Это самый короткий путь, хотя, на первый взгляд, он кажется длинным. Обходного пути, исключаящего практику, не существует. Без практики невозможно приобрести навык. Проще не бывает.

Почему мы не тренируемся? Причины две — занятость и страх.

Но главное препятствие к быстрому приобретению навыка не физическое или интеллектуальное, а *психологическое*. Любое новое дело поначалу вызывает дискомфорт — в результате мы потратим массу времени и сил, размышляя о занятиях, вместо того чтобы заниматься.

К счастью, барьер фрустрации преодолеть несложно: приобретение навыка всегда *кажется* более трудным делом, чем это есть на самом деле. Выделив время для занятий, немного изучив теорию и признав неизбежность начального дискомфорта, вы обязательно добьетесь больших успехов за первые 10 или 20 часов занятий. Чтобы выиграть приз, необходимы лишь решимость,

настойчивость и немного терпения.

И не нужно выбирать сразу несколько навыков — достаточно одного. Выберите навык из списка «хочется» и заставьте себя попробовать. Выучите язык, освоите музыкальный инструмент, игру, осуществите проект, приготовьте кулинарное блюдо, напишите картину. Это легче, чем вам кажется.

Заранее приготовьтесь тратить на приобретение навыка около часа в день в течение следующего месяца. Приступив к занятиям, вы овладеете навыком быстрее, чем думали. Разбейте навык на элементы, найдите время для занятий, экспериментируйте — и ваш мозг начнет автоматически обучаться, именно так он устроен. Если вы растерялись или чувствуете, что дело не движется, попробуйте новый подход.

Помните: приступив к занятиям, нельзя останавливаться, пока вы не достигнете целевого уровня мастерства или 20-часовой отметки. Преодолевайте себя, если потребуется, но не отступайте. Стисните зубы и продолжайте идти вперед. Вы обязательно дойдете до цели — для этого нужна только практика.

И последнее: начать заниматься можно только *сегодня*.

Не завтра. Не на следующей неделе. Не в следующем месяце или в следующем году. Сегодня.

Когда вы просыпаетесь утром, у вас есть выбор. Можно инвестировать свое время в приобретение навыков, которые сделают жизнь более успешной, радостной и приносящей удовлетворение... или потратить время на что-то другое.

Итак, чем вы займетесь сегодня?

Благодарности

Келси и Леле: я вас люблю. Спасибо за все.

Дэйву, Шерри и Зулеме: спасибо, что присматривали за Лелой. Иначе эта книга не была бы написана.

Лесли Каминоффу, Дереку Сиверсу, Джиму Дугласу и Гэри Стоуну: спасибо за помощь. Я очень многому у вас научился.

Лорен Бейкер: спасибо за иллюстрации к этой книге.

Лизе Димона: я безмерно счастлив, что получил вашу помощь и поддержку.

Адриану Закхейму и Джоэлу Рикетту: viva Portfolio!

Эмили Эйнджел: вы в совершенстве владеете профессией редактора. Спасибо за проницательность и усердие.

Уиллу Вейссеру, Маргот Стамос, Ричарду Леннону и Эллисон Маклин: книгам нужны читатели. Спасибо, что рассказывали обо мне.

Джо Перецу и Дэну Донахью: в глазах тех, кто судит о книгах в первую очередь по обложкам, ваши авторы всегда выглядят великолепно. Я ценю ваш упорный труд по претворению этой идеи в жизнь.

Брии Сэнфорд, Саманте Лаблю, Саре Кэти Коэ, Томасу Дасселу и всей команде издательства Penguin, отвечавшей за выпуск и дистрибуцию: спасибо за ваши усилия, которые позволили познакомить с книгой читателей во всем мире.

Вам, дорогой читатель: я все это делаю ради вас. Надеюсь, книга вам понравилась, и информация, которую вы из нее почерпнули, оказалась полезной. Удачи!

Дополнительные материалы к этой книге можно найти в интернете по адресу: <http://first20hours.com.update>.

Картинки, видеоролики и комментарии к отдельным главам этой книги вы сможете найти в Сети:

к главе 4 — <http://first20hours.com/yoga>.

к главе 5 — <http://first20hours.com/programming>.

к главе 6 — <http://first20hours.com/typing>.

к главе 7 — <http://first20hours.com/go>.

к главе 8 — <http://first20hours.com/ukulele>.

к главе 9 — <http://first20hours.com/windserfing>.

Примечания

Глава 1

1. Ericsson, K. Anders, Neil Charness, Paul J. Feltovich, and Robert R. Hoffman, eds. *The Cambridge Handbook of Expertise and Expert Performance* (Cambridge: Cambridge University Press, 2006).

2. <http://www.chirunning.com/>.

3. Как правило, продолжительность базовых программ колледжа составляет четыре года, и причина этого — традиция и материальная заинтересованность, поскольку за каждый год обучения колледж получает плату. Но это не значит, что невозможно завершить даже самый сложный курс за более короткое время: курс лекций по компьютерам, который читает в МТИ Скотт Х. Янг, занимает меньше года. См. <http://www.scotthyoung.com/blog/mit-challenge/>.

4. VanLehn, Kurt. “Cognitive Skill Acquisition.” *Annual Review of Psychology* 47, no. 1 (1996): 513–539.

Глава 2

1. Рекомендую прочитать книгу Маргарет Лобенстайн «*The Renaissance Soul: Life Design for People with Too Many Passions to Pick Just One*» (New York: Harmony, 2006). Эта книга помогла мне понять, что разнообразие интересов — это достоинство, а не недостаток и что структурированием усилий можно извлечь пользу из моего стремления научиться всему и сразу.

2. http://www.newyorker.com/reporting/2011/10/03/111003fa_fact_gawande.

3. Я использовал программируемый интервальный таймер Enso Pearl: <http://www.salubrion.com/products/ensopearl/>. Вы можете взять любой таймер, какой пожелаете, в том числе кухонный, но у Enso имеются уже запрограммированные интервалы и приятный звуковой сигнал, что, как оказалось, имеет значение. Если вы много занимаетесь, пронзительный писк быстро начинает раздражать.

4. Snoddy, George S. “Learning and Stability: A Psychophysiological Analysis of a Case of Motor Learning with Clinical Applications.” *Journal of Applied Psychology* 10, no. 1 (1926): 1.

5. Newell, Allen, and Paul S. Rosenbloom. “Mechanisms of Skill Acquisition and the Law of Practice.” *Cognitive Skills and Their Acquisition* (1981): 1–55.

6. Logan, Gordon D. “Toward an Instance Theory of Automatization.” *Psychological Review* 95, no. 4 (1988): 492.

Глава 3

1. <http://ankisrs.net/>.

2. <http://www.supermemo.com/>.

3. <http://smartr.be/>.

4. Эта книга не о методах обучения, но если вы прилежно читаете примечания, я могу предложить вам следующий метод: выберите какую-нибудь идею, возьмите лист бумаги и попытайтесь объяснить эту идею максимально простым языком, как будто разговариваете с новичком. Пробелы в ваших знаниях выявятся очень быстро, что облегчит возвращение к исходному материалу для усвоения пропущенного. Исследователь навыков обучения Скотт Х. Янг называет этот чрезвычайно эффективный метод «техникой Фейнмана» (в честь известного физика Ричарда Фейнмана). Более подробную информацию вы можете найти на сайте <http://www.scotthyoung.com/learnosteroids/grab/TranscriptFeynman.pdf>.

Глава 4

1. Это действительно очень, очень здорово — бежать изо всех сил к барьеру высотой до пояса и перепрыгивать через него, не сбиваясь с шага. Однажды я почувствовал себя почти Суперменом, когда во время забега моя нога попала в самый центр барьера. Деревянная планка разлетелась на куски, а я помчался дальше, не теряя скорости. Потрясающее ощущение.

2. http://www.youtube.com/watch?v=IMC1_RH_b3k.

3. <http://www.nytimes.com/2009/07/26/magazine/26FOB-consumed-t.html>.

4. <http://www.gilhedley.com/>.

5. Знаменитую «речь» Гила вы можете посмотреть здесь: <http://www.youtube.com/watch?v=FtSP-tkSug>. Имейте в виду, что видео содержит кадры с человеческим трупом; если у вас слабые нервы, эти моменты стоит пропустить.

6. Sarno, John E. *Healing Back Pain: The Mind-Body Connection* (New York: Grand Central Life & Style, 2010).

7. Я собираюсь в нескольких абзацах изложить несколько тысяч лет необычайно сложной истории йоги. Из книг на эту тему можно составить целую библиотеку, так что я прошу извинить меня за краткость. Если вы хотите подробнее ознакомиться с предметом, я рекомендую книгу Карен Армстронг «*The Great Transformation: The Beginning of Our Religious Traditions*» (New York: Anchor, 2007).

8. Armstrong, Karen. *The Great Transformation: The Beginning of Our Religious Traditions* (New York: Anchor, 2007).

9. Desikachar, T. V. K., and R. H. Cravens. *Heath, Healing, and Beyond: Yoga and the Living Tradition of Krishnamacharya* (New York: North Point Press, 1998). A short biography is also available at <http://www.yogajournal.com/wisdom/465>.

10. Специалисты до сих пор не пришли к единому мнению, в какой степени Кришнамачарья использовал внешние источники, такие как британская

гимнастика. Многие позы и движения, которые Кришнамачарья включил в свои комплексы, очень похожи на гимнастику и упражнения военной подготовки того времени.

11. http://www.nytimes.com/2012/01/08/magazine/how-yoga-can-wreck-you-body.html?_r=3&pagewanted=all.

12. <http://www.manduka.com/us/shop/categories/products/gear/manduka-pro-black-sage/>.

13. <http://orthoinfo.aaos.org/topic.cfm?topic=A00063>.

Глава 5

1. <http://personalmba.com/best-business-books/>.

2. <http://wordpress.org>.

3. Для тех, кому это интересно: мой стандартный пакет WordPress состоял из PHP5 с PHPFastCGI, NGINX, APC, MSMTP и WP-Supercache на Slicehost.com VPS, выполняющийся на Ubuntu 8.04 LTS, все с файлами конфигурации пользователя.

4. <http://jekyllrb.com>.

5. <http://github.com>.

6. По какой-то причине почти каждое руководство по программированию начинается с примера, как вывести на экран или распечатать сообщение «Hello, world!».

7. <http://stackoverflow.com>.

8. <http://news.ycombinator.com>.

9. <http://rubyonrails.org>.

10. <http://www.sinatrarb.com>.

11. <http://37signals.com>.

12. <http://rubysource.com/rails-or-sinatra-the-best-of-both-worlds/>.

13. <http://paulstamatiou.com/how-to-wordpress-to-jekyll>.

14. <https://github.com/sstephenson/rbenv>.

15. На практике «хакерство» совсем не похоже на то, что показывают в кино, и это меня очень разочаровало.

16. <https://toolbelt.heroku.com>.

17. <http://www.heroku.com>.

18. <http://git-scm.com>.

19. Версии Ruby старше 1.9.3-p125 требовали программу под названием GCC, чтобы завершить процесс установки. GCC доступна на <https://github.com/kennethreitz/osx-gcc-installer>.

20. Программисты придумали акроним «RTFM», который означает «читайте (чертову) инструкцию» как стандартный ответ на вопросы, освещенные в официальной документации.

21. <http://www.ruby-lang.org/en/documentation>.

22. <http://0xfe.muthanna.com/rubyrefresher>.
23. <https://code.google.com/p/ruby-security/wiki/Guide>.
24. Термин «метапрограммирование» обозначает продвинутую модификацию основных объектов, классов и методов Ruby. Я приобрел книгу Паоло Перотты под названием «Метапрограммирование на Ruby: Профессиональное программирование на Ruby» («Metaprogramming Ruby: Program Like the Ruby Pros, Paolo Perrotta») (Raleigh, NC: Pragmatic Bookshelf, 2010), но пока ничего в ней не понял.
25. <http://www.ruby-doc.org/core-1.9.3/index.html>.
26. <http://ruby.learncodethehardway.org>.
27. <https://devcenter.heroku.com/articles/keys>.
28. <https://devcenter.heroku.com/articles/ruby>.
29. <https://devcenter.heroku.com/articles/rack>.
30. <http://macromates.com>.
31. <https://devcenter.heroku.com/articles/bundler>.
32. <http://www.sinatrarb.com/intro>.
33. <http://backpackit.com>.
34. <http://tom.preston-werner.com/2010/08/23/readme-driven-development.html>.
35. <http://www.postgresql.org>.
36. <http://datamapper.org>.
37. <http://stackoverflow.com/questions/1152299/what-is-an-object-relational-mapping-framework>.
38. <http://en.wikipedia.org/wiki/SQL>.
39. <http://www.sqlite.org>.
40. <http://pow.cx>.
41. <https://github.com/rodreegez/powder>.
42. <http://twitter.github.com/bootstrap>.
43. Я понятия не имею, почему это так называется.
44. См. <http://www.regular-expressions.info/> for examples of common regular expressions.
45. <http://daringfireball.net/projects/markdown>.
46. <http://www.httpwatch.com/httpgallery/authentication>.
47. Использовать SSL на пользовательском домене сложнее: вы должны пройти через длинный процесс подтверждения и получить «сертификат», защищающий каждую сессию пользователя.
48. <https://github.com/SFEley/sinatra-flash>.

Глава 6

1. Любопытный факт: Тайгер Вудс — единственный профессиональный гольфист в истории, выигрывавший турнир U. S. Masters тремя разными

свингами. Тайгер переучивался после победы в 1997 году, снова победил на турнире в 2001 и 2002 годах, еще раз поменял удар в 2002 году, затем победил в 2005 году. Недавно, в 2011 году, Тайгер снова стал переучивать удар, и это будет его четвертый свинг за профессиональную карьеру.

2. Изображение клавиатуры с сайта <http://wowvectors.com/object/mac-keyboard-vector/>. Распространяется по лицензии Creative Commons Attribution 3.0 Unported.

3. Очень интересные факты о разработке пишущей машинки QWERTY можно найти в статье «The Fable of the Keys» С. Дж. Лейбовица и Стивена И. Марголиса на <http://www.utdallas.edu/~liebowitz/keys1.html>.

4. <http://www.google.com/patents?id=qSVdAAAAEBAJ>.

5. <http://mkweb.bcgsc.ca/carpalx>.

6. <http://mkweb.bcgsc.ca/carpalx/?colemak>.

7. <http://colemak.com>.

8. Для других операционных систем требуется установка небольшого пакета программного обеспечения, поддерживающего раскладку. Пакеты для большинства популярных систем можно найти на <http://colemak.com>.

9. <http://www.typematrix.com/2030/features.php>.

10. На клавиатуре TypeMatrix вы можете активировать режим Colemak нажатием Fn+F5.

11. Этот метод подходит не для всех клавиатур: в некоторых моделях клавиши верхнего, среднего и нижнего ряда имеют разный размер. В таком случае, наверное, проще купить новую клавиатуру или заказать наклейки на клавиши.

12. <http://www.typeonline.co.uk/typingspeed.php>.

13. Материал теста берется из общедоступных книг, бесплатно распространяемых в сети посредством проекта Gutenberg. См. <http://www.gutenberg.org/>.

14. <http://www.mavisbeacon.com/>.

15. <http://typingtrainer.sourceforge.net/>.

16. <http://github.com/wwwtyro/keyzen>.

17. Luft, Andreas R., and Manuel M. Buitrago. “Stages of Motor Skill Learning.”

Molecular Neurobiology 32, no. 3 (2005): 205–216.

18. Walker, Matthew P., and Robert Stickgold. “It’s Practice, with Sleep, That Makes Perfect: Implications of Sleep-Dependent Learning and Plasticity for Skill Performance.” *Clinics in Sports Medicine* 24, no. 2 (2005): 301–317.

19. <http://www.daskeyboard.com/model-s-ultimate-silent/>.

20. <http://type-fu.com>.

21. Milton, James. *Measuring Second Language Vocabulary Acquisition* (Bristol, UK: Multilingual Matters, 2009).

22. <http://code.google.com/p/amphetype/>.

23. <http://norvig.com/ngrams/>.

Глава 7

1. На мой взгляд, самый главный недостаток реального мира — отсутствие магии: с учетом того, сколько книг я прочел, я был бы уже волшебником *как минимум* 80-го уровня. Увы, я не способен силой мысли изменять материальный мир.

2. Еще один любопытный факт: если тщательно перетасовать карточную колоду, то окажется, что получившаяся последовательность карт никогда не встречалась за всю историю Вселенной. «52 факториал» — это очень большое число: 8,065 Ч 1067 возможных комбинаций.

3. Среди других знаменитых партий — «партия кровавой рвоты», когда один из мастеров умер после матча, и «партия атомной бомбы», прерванная взрывом атомной бомбы над Хиросимой. После взрыва бомбы игроки сделали перерыв на обед, заменили камни и продолжили игру. Информацию о других знаменитых партиях можно найти на сайте <http://senseis.xmp.net/?FamousGoGames>.

4. <http://www.ymimports.com>.

5. <http://senseis.xmp.net/>.

6. Да, именно от этого термина произошло название игровой приставки Atari.

7. <http://senseis.xmp.net/?TheTenGoldenRulesList>.

8. http://diiq.org/five_stone_questions.html.

Глава 8

1. <http://cdp.sagepub.com/content/14/6/317.short>.

2. <http://www.crowhillguitars.com>.

3. <http://www.daddario.com/DADProductDetail.Page?ActiveID=3769&productid=26/>.

4. <http://www.axisofawesome.net>.

5. <http://www.ukuleles.com/Technology/strings.html>.

6. <http://www.kiwiukulele.co.uz/Kiwi-Ukulele-Chord-Chart.pdf>.

Глава 9

1. <http://jimbodouglass.blogspot.com/>.

2. Если вы хотите понять, что такое мчаться на доске Formula, посмотрите видео, снятое камерой Джима, прикрепленной к гику: <http://jimbodouglass.blogspot.com/2010/01/formula-windsurfing-boom-mount-video.html>.

3. Бойскауты называют такой прием «работой в паре», и это главное

правило безопасности на воде.

4. «Мокрый» гидрокостюм удерживает у поверхности тела слой воды, который нагревается и служит в качестве термоизоляции. При температуре воды ниже 10 градусов Цельсия лучше использовать «сухой» гидрокостюм, который не пропускает воду к коже.

5. <http://www.dbw.ca.gov/Pubs/Windsurf/index.htm>.

6. <http://jimbodouglass.blogspot.com/2008/02/top-16-windsurfing-questionsanswered.html>.

7. <http://jimbodouglass.blogspot.com/2010/11/updated-windsurf-calculatoronline.html>.

8. <http://www.isthmussailboards.com/>.

9. <http://www.wunderground.com/>.

10. <http://www.windfinder.com/>.

11. <http://www.iwindsurf.com/>.

12. <http://www.kestrelmeters.com/products/kestrel-3000-wind-meter>.

13. http://www.isthmussailboards.com/info_technical_help.asp.

14. Подробную инструкцию, снабженную рисунками, вы можете найти на <http://jimbodouglass.blogspot.com/2012/10/beginner-windsurfing-how-to-reorient.html>.

Примечания

1

Ad infinitum — до бесконечности (лат.). — *Прим. пер.*

2

Гладуэлл М. Гении и аутсайдеры. Почему одним всё, а другим ничего? — М.: Манн, Иванов и Фербер, 2013.

3

Шелли П. Б. Озимандия / Пер. К. Бальмонта. — *Прим. пер.*

4

Привет, как дела, отлично (исп.). — *Прим. пер.*

5

Кауфман Дж. Сам себе MBA. Самообразование на 100 %. — М.: Манн, Иванов и Фербер, 2013.

6

Холмс Ч. Совершенная машина продаж. 12 проверенных стратегий эффективности бизнеса. — М.: Альпина Паблишер, 2013.

7

Аллен Д. Как привести дела в порядок. Искусство продуктивности без стресса. — М.: Манн, Иванов и Фербер, 2013.

8

Бейлс Д., Орланд Т. Искусство и страх. Гид по выживанию для современного художника. — СПб.: Питер, 2011.

9

Перевод на русский язык издательства «Питер».

10

Лоэр Дж., Шварц Т. Жизнь на полной мощности! Управление энергией — ключ к высокой эффективности, здоровью и счастью. — М.: Манн, Иванов и Фербер, 2009.

11

Каминофф Л., Мэтьюз Э. Анатомия йоги. — М.: Попурри, 2012.

12

Дж. Мартин «Песнь Льда и Пламени», перевод Н. Виленской.

13

Гильбо К. Стартап за \$100. Создай новое будущее, делая то, что ты любишь. — М.: Манн, Иванов и Фербер, 2013.

14

Крис Гильбо. Искусство нонконформизма. — М.: Попурри, 2012.

**После ознакомления с книгой не забудьте купить ее
бумажную версию!**

Все книги на сайте представлены исключительно в ознакомительных целях.

После скачивания книги и ознакомления с ее содержанием Вы должны незамедлительно ее удалить.

Копируя и сохраняя текст книги, Вы принимаете на себя всю ответственность, согласно действующему законодательству об авторских и смежных правах.

Администрация сайта призывает своих посетителей приобретать книги только легальным путем.