

МАЙКЛ ЛИТТМАН

# ДРУЖЕЛЮБНЫЙ КОД

КАК ПРОГРАММИРОВАТЬ  
ЛЕГКО И ПРОСТО

УДК 004.4  
ББК 32.973.2-018  
Л64

Michael L. Littman  
CODE TO JOY:  
WHY EVERYONE SHOULD LEARN A LITTLE PROGRAMMING

©2023 Massachusetts Institute of Technology  
The rights to the Russian-language edition obtained via Igor Korzhenevskiy  
of Alexander Korzhenevskiy Agency (Moscow).

**Литтман, Майкл.**  
Л64 Дружелюбный код. Как программировать легко и просто / Майкл Литтман ; [перевод с английского Е. В. Жевлаковой]. — Москва : Эксмо, 2025. — 256 с.— (Мировой компьютерный бестселлер).

ISBN 978-5-04-193570-2

Стремительное развитие информационных технологий побуждает каждого так или иначе взаимодействовать с миром ИТ. Сейчас сложно представить, чтобы наша жизнь обходилась без множества простых и доступных функций, каждая из которых задана... обычным кодом. Многих пугают и отталкивают такие слова, как «код» и «языки проаммирования», но они могут быть простыми и понятными. Майкл Литтман интересным и доступным языком рассказывает о программировании и о том, как оно может быть использовано с пользой в современном мире. Автор научит не бояться искусственного интеллекта, а понять его. Кроме того, книга наполнена множеством уникальных и понятных примеров, которые сделают знакомство с кодом по-настоящему захватывающим.

УДК 004.4  
ББК 32.973.2-018

ISBN 978-5-04-193570-2

© Жевлакова Е.В., перевод на русский язык, 2025  
© Оформление. ООО «Издательство «Эксмо», 2025

# ОГЛАВЛЕНИЕ

---

---

ГЛАВА 1. ДАЕМ КОМПЬЮТЕРАМ ЗАДАНИЯ: Я ВАС ПОНЯЛ .....	9
ПОЧЕМУ МЫ ВЛАДЕЕМ КОМПЬЮТЕРАМИ .....	12
ПОЧЕМУ КОМПЬЮТЕРЫ ВЛАДЕЮТ НАМИ .....	15
ДАЛЕЕ СЛЕДУЮТ МАТЕРИАЛЫ ДЛЯ ВЗРОСЛЫХ .....	16
ПОДАВАЯ ПРИМЕР .....	22
ВЕЛИКИЕ СЦЕНАРИСТЫ .....	25
ГЛАВА 2. ЧТО ТАКОЕ ПРОГРАММИРОВАНИЕ:	
ПОЗВОЛЬТЕ ПРЕДСТАВИТЬСЯ .....	30
ТМ!, ИЛИ РАЗГРУЗИ МЕНЯ .....	33
УЧИТЬСЯ ПРОГРАММИРОВАТЬ И УЧИТЬ	
ПРОГРАММИРОВАНИЮ .....	39
ГЛАВА 3. ПОСЛЕДОВАТЕЛЬНОСТЬ КОМАНД:	
Я ГОТОВ ПРИНЯТЬ ВАШ ЗАКАЗ .....	45
МОЖНО МНЕ МАКРОС? .....	49
ПОД МАКРОСКОПОМ .....	52
ЭКСТРАОРДИНАРНЫЕ ВОПРОСЫ .....	58
«ПОКАЖИТЕ СИНЮЮ!» .....	60
СОСТАВЬ МНЕ ГОЛОВОЛОМКУ .....	66
НЕВЕЛИКА ПОТЕРЯ .....	69
МАШИНЫ УЧАТСЯ УЧИТЬСЯ .....	71
НАПОСЛЕДОК ПОИГРАЕМ В СЛОВА .....	74

ГЛАВА 4. РАЗДЕЛЕНИЕ ПО УСЛОВИЯМ: ЕСЛИ ВЫ ТАК ХОТИТЕ! .....	78
ВЫБИРАЙТЕ УСЛОВИЯ САМИ .....	84
ЧЕТВЕРТАЯ R .....	89
ПАРНЫЕ УСЛОВНЫЕ КОНСТРУКЦИИ .....	100
ГЛАВА 5. ХРАНЕНИЕ В ПЕРЕМЕННЫХ: ТО, ЧТО Я ЗАМЕНЯЮ СОБОЙ .....	108
ОТ СПАМА НИКУДА НЕ ДЕТЬСЯ .....	113
ВЫ ИЗ КАКОЙ ОБЛАСТИ? .....	118
КАК НЕ СХЛОПОТАТЬ В ТАБЛИЦУ .....	121
«ОЙ, ОНИ ТАК ПОХОЖИ НА НАС!» .....	124
«ЭТОТ МАКАРОННЫЙ БЛИЦ-ОПРОС ОПРЕДЕЛИТ ВАШ ЯЗЫК ЛЮБВИ» .....	126
ЛИЦЕНЗИЯ НА «НЕ УБИЙ» .....	131
ГЛАВА 6. ОБЪЕДИНЕНИЕ В ЦИКЛЫ: ВАМ НЕ НУЖНО ПОВТОРЯТЬ МНЕ ДВАЖДЫ .....	137
СНОВА ДЕНЬ СУРКА .....	140
ВСЁ ПРОСТО И ЛЕГКО. НА ВЕКИ ВЕЧНЫЕ .....	146
ЦИКЛЫ ОБУЧЕНИЯ .....	153
КАК ВЫ ИЗМЕРЯЕТЕ ГОД? .....	156
ВЫГОДЫ И ЗАТРАТЫ ПРИ ОБУЧЕНИИ НА ОСНОВЕ ВЫГОД И ЗАТРАТ .....	162
МОИ БЕСКОНЕЧНЫЕ ЦИКЛЫ .....	166
ГЛАВА 7. ОПРЕДЕЛЕНИЕ ФУНКЦИЙ: НАЗОВИ МЕНЯ СВОИМ ИМЕНЕМ .....	170
ПО СКРИПТУ .....	171
ПУСТЬ С ВАМИ СЧИТАЮТ(СЯ) .....	176
РАЗГОВОР О РАЗГОВОРЕ О РЕКУРСИИ .....	180
ИСКУССТВО РЕКУРСИИ .....	184
НАЗВАТЬ НАЗВАНИЯ .....	187
РЕКУРСИВНЫЙ, НО НЕ КУРСИВНЫЙ .....	190
ДВОЙНОЙ СЕАНС С ЧУДОВИЩАМИ .....	192

ГЛАВА 8. СОЧЕТАНИЕ КОДА И ДАННЫХ:	
ДА, КОНЕЧНО, НО НЕ МОГЛИ БЫ ВЫ ПРИВЕСТИ ПРИМЕР? .....	197
ПЕРЕМОЕМ КОСТОЧКИ ПОСУДОМОЙКАМ .....	199
ГОВОРЯЩИЕ МАШИНЫ .....	205
МАШИНЫ ПИШУТ ПРОГРАММЫ .....	210
НЕ ЖЕЛАЕШЬ ЛИ СЛЕПИТЬ ПРОГРАММУ? .....	213
ЗАБЛУДИТЬСЯ С СОМРАС'ОМ .....	218
ПОГРУЖЕНИЕ НА ГЛУБИНУ СО SNORKEL'ЕМ .....	225
ГЛАВА 9. ПРОГРАММИРУЕМЫЙ МИР:	
ВСЁ КАК ВЫ ПОЖЕЛАЕТЕ .....	230
ЛАЙФХАКИ .....	231
ЗАВИСИМОСТЬ ОТ КОДА .....	240
БЛАГОДАРНОСТИ: СПАСИБО! .....	244
АЛФАВИТНЫЙ УКАЗАТЕЛЬ .....	248



# 1

## ДАЕМ КОМПЬЮТЕРАМ ЗАДАНИЯ: Я ВАС ПОНЯЛ

---

---

Если вы следите за новостями, то знаете, что компьютеры выполняют за нас работу. И шпионят за нами. Контролируют, что мы покупаем и за кого голосуем. Даже ущемляют нас в правах. Когда им надоест побеждать нас в наших же привычных забавах, возможно, они восстанут и убьют нас. В общем, отношения людей с машинами стали, откровенно говоря, нездоровыми.

Однако так было не всегда. На заре развития вычислительной техники мы более четко очерчивали границы дозволенного. Тогда с компьютерами взаимодействовали в основном те же люди, которые их программировали. Именно они создавали программное обеспечение (ПО), писали простые игры, выявляли проблемы в мире и придумывали, как ЭВМ могут принести пользу в их решении. Короче говоря, царила идиллия. К сожалению, те реалии исчезли точно так же, как домашние фильмы, снятые на 8-миллиметровую киноплёнку. Однако еще не поздно вернуть некоторые лучшие моменты. Это непросто и потребует времени, но нам под силу восстановить более естественную динамику отношений.

Первый шаг — напомнить себе: эти машины изобрели, чтобы повысить нашу эффективность. Второй шаг более масштабный: нам всем нужно стать программистами.

Нет, не удивляйтесь! Позвольте объяснить, что я подразумеваю под этим словом.

Конечно, я не имею в виду, что нужно научиться использовать языки программирования (ЯП) на уровне крутых разработчиков приложений. То есть нам всем не надо учиться на *кодера*, хотя, как профессионал и преподаватель в области компьютерных наук, я думаю, что это отличный навык, владеть которым порой очень даже приятно.

Я имею в виду следующее: каждый, кто пользуется компьютером, уже говорит ему, что от него хочет. Мы сообщаем машинам о своих желаниях различными способами — от нажатия кнопок до произнесения команд на естественном языке, например, когда просим Google, Siri или их собратьев «включить нам что-нибудь веселое» или интересуемся, «сколько ехать на автомобиле до Саскатуна» (читатели из Саскатуна могут смело подставить название другого города).

Однако мы сильно ограничены в том, что именно доносим до компьютера, и это не позволяет нам добиться того, чтобы он реагировал на *будущие* события так, как нам хотелось бы. Умение сделать это любым из существующих способов — вот что я называю программированием. Вот о чем эта книга. Поскольку вселенная программных интерфейсов по-настоящему огромна, я планирую вдаваться в подробности каких-либо конкретных приложений лишь в тех случаях, когда привожу примеры. Но даже тогда я не стану пытаться учить вас некому языку или пользованию определенным интерфейсом — скажем, тому, как писать формулы в Excel или устанавливать график температур на вашем термостате Nest.

Вместо этого мы рассмотрим то, что фактически представляют собой все конкретные языки и пользовательские интерфейсы: грамматику концепций, совместно формирующих фундаментальные подходы, с помощью которых мы заставляем компьютеры работать на нас. Эта тема особенно актуальна, поскольку методы искусственного интеллекта, в частности машинное обучение, прямо сейчас меняют то, как программируют любители и профессионалы. Мы рассмотрим и традиционные, и новые способы передачи наших намерений и желаний компьютерам.

Обещаю: экзамена по программированию в конце книги не будет.

\*\*\*

Одни люди считают, что программирование компьютеров — сложный навык, который нужно долго оттачивать. Другие думают иначе, но они просто-напросто ошибаются. Однако преимущества искусственного интеллекта и машинного обучения помогут машинам пойти нам навстречу. Благодаря этим достижениям компьютерное программирование станет все сильнее походить на обучение другого человека. Конечно, обучать людей тоже непросто, но на самом базовом уровне это один из наших врожденных навыков. Мы приходим в мир, уже обладая способностью учить. А компьютеры изначально созданы, чтобы учиться. Думаю, мы справимся с задачей.

Цель книги — рассказать, как сообщать машинам, чего мы от них хотим, чтобы вы, читатели, более уверенно делали это самостоятельно. Каждая глава посвящена определенному элементу команд, а также содержит аналогии из повседневного взаимодействия с людьми. Также вы найдете описания того, как экспериментировать с этими понятиями в вычислительной среде, используя общедоступные системы. Тут есть приятный побочный эффект: это повысит производительность вашего труда. В каждой главе я также рассматриваю, как облегчить применение этих программных компонентов с помощью машинного обучения — технологии обработки данных для автоматического создания полезных приложений. Различные программные компоненты дают нам основу для «дорожной карты» перехода к новому миру, где программирование будет цениться наравне с грамотностью — станет тем, что каждый замотивирован освоить, тем, что принято знать в обществе. Я использую термин «программирование» или фразу «сообщать машинам, что делать», чтобы донести общую идею передачи задания, которое машине предстоит сделать за нас. К программированию также относится написание кода в традиционном значении этого термина, как частный случай создания набора инструкций, которым следует компьютер. Но программирование — это также и любой способ, которым мы передаем машине, как ей себя вести, чтобы выполнить наше поручение. Написание кода — один из наиболее формализованных способов выполнения работы. Однако настоящие и будущие воображаемые системы черпают вдохновение в том, как люди учат друг друга что-то делать. Давайте посмотрим, как это происходит.

Уильям Артур Уорд, автор афоризмов для «*Ридерз Дайджест*», сказал в 1965 году:

*Посредственный учитель рассказывает. Хороший учитель объясняет.*

*Выдающийся учитель показывает. Великий учитель вдохновляет.*

Неплохо. Хотя, как у профессионального преподавателя, у меня есть несколько вопросов к этой цитате. Во-первых, учитель, который ничего не рассказывает, вряд ли вдохновит. Я склонен думать, что Уорд подразумевал, что великий учитель опирается на *все* эти элементы, но, будучи великим автором афоризмов, выразил суть идеи как можно короче.

Так или иначе, полагаю, в словах Уорда есть нечто ценное, поскольку эти четыре стиля преподавания — рассказ, объяснение, демонстрация и вдохновение — могут служить концептуальной схемой для четырех главных методик, которые мы сегодня используем, чтобы отдавать

машинам распоряжения. Далее я расскажу вам (и объясню, и покажу, и, возможно, даже вдохновлю вас!) о каждой из этих моделей, доказывая, что их надлежит использовать в симбиозе, чтобы сделать программирование более доступным. Если написанием кода сможет заниматься каждый, если получится создать нечто вроде всеобщей грамотности в этой области, человечество пойдет по самому легкому пути к восстановлению взаимовыгодных отношений между людьми и машинами.

## ПОЧЕМУ МЫ ВЛАДЕЕМ КОМПЬЮТЕРАМИ

Компьютеры автоматизируют: мы даем им задачу по обработке информации, и они выполняют ее за нас. В этом смысле машины похожи на человеческий мозг. По сути, совсем недавно, в 1970-х, английским словом *computer* («расчетчик», «выкладчик») называли должность, на которой работали люди.

И все же, если человек и так обладает хорошими вычислительными навыками, зачем нам понадобились электронные счетные машины? Почему мы не придерживаемся старых проверенных методов, просто поручая выполнение задач по обработке информации людям? Ведь человека, например, гораздо проще сделать. Еще никто из нас, напившись в баре и завалившись с кем-нибудь в туалет, не собирал там случайно ноутбук с графическим сопроцессором. Кроме того, довольно сложно объяснить роботу, управляемому компьютером, как выполнить несложное поручение — например, купить в конце улицы замороженный йогурт, убедившись, что сверху есть фрукты (ну, вы понимаете, так полезнее).

Существует несколько причин, по которым мы поручаем компьютерам выполнять работу за нас. Лучше всего компьютерам подходят те задачи, при выполнении которых люди (1) справляются слишком медленно, (2) слишком быстро устают, или (3) если их работа обходится слишком дорого.

Например, всеми планетоходами на Марсе управляют компьютеры. Причина проста: время, необходимое для передачи соответствующих данных водителю на Земле, а затем для получения обратного сигнала фактического управления марсоходом, составляет от 10 до 40 минут. Управлять такой машиной с Земли — все равно что играть в самую нудную на свете вариацию *Mario Kart*\*. Чтобы операторы марсохода

---

\* *Mario Kart* — серия компьютерных игр в жанре гонок, разработанных и изданных японской компанией Nintendo в качестве спин-оффа к серии игр *Super Mario*. — Здесь и далее прим. науч. ред., если не указано иного.

меньше раздражались, выполняя миссию, ракетостроители установили на его борту компьютеры, которые сами анализируют изображения с камер и направляют небольшой аппарат к каменюкам, представляющим научный интерес (полагаю, уворачиваясь при этом от «синих ракушек»<sup>\*</sup>).

Иногда планетоход под управлением программы застревает, и тогда вмешивается специалист миссии. Возможно, лучше, чтобы аппаратом непосредственно рулил *марстронавт*? Вообще говоря, да, так гораздо лучше. Однако люди еще не придумали, как выжить на Красной планете, из-за чего этот план выглядит менее привлекательно. Как пел сэр Элтон Джон в *Rocket Man*, Марс — не то место, где можно растить детей. И так, там работают компьютеры, потому что люди слишком медленные: отдавать команды достаточно быстро не получится, так как операторы очень далеко. Помимо того, мы слишком легко устаем, а управление роботом, который должен пройти полтора километра со скоростью 0,0032 км/час, кого угодно сведет с ума. Очень, очень медленно сведет. К тому же труд людей слишком дорого стоит: по оценкам, отправка водителя на Марс обошлась бы в миллиард долларов, а столько с вас не сдерут даже за такси из аэропорта в период высокого спроса.

Исследование поверхности Марса — это пример использования вычислительной техники «где-то там». Однако более обыденное применение машин тоже подпадает под перечисленные мною три категории задач. Для отправки сообщений вы снова обращаетесь к компьютерам, поскольку иные способы рассылки фотографий очаровательных миниатюрных лам всем участникам группового чата отняли бы слишком много времени. На компьютере вы ищете магазин, в котором продается ваш любимый сорт кофе, поскольку вам не хватает терпения, чтобы поочередно обойти все торговые точки в радиусе восьмидесяти километров. Помимо того, макет делового письма вы тоже создаете на компьютере, поскольку нанять художника-полиграфиста — слишком дорогая затея.

Заметьте: между этими тремя категориями нет четкого деления. Дефицит в одной из них можно восполнить избытком в другой. Если вы устаете не так быстро, то сумеете работать дольше и тем самым компенсировать недостаток средств на выполнение какой-либо задачи. Если же вы работаете достаточно споро, то уже не имеет значения, что вы легко устаете, поскольку вы справитесь с заданием еще до того, как заскучаете. И так далее. Однако случается, что нам не хватает времени, сил

---

<sup>\*</sup> *Синяя ракушка* (англ. Blue Shell) — элемент игры, представляющий собой помеху для игрока в серии видеоигр Mario Kart.

и денег — всего сразу. Вот тогда компьютеры действительно могут «заработать» себе на жизнь. Благодаря им мы становимся молниеносными, неутомимыми и неприлично богатыми. История еще не знала случая, когда людям предоставлялся столь массовый доступ к таким большим возможностям. В Средние века приложений не существовало. Отправлять послания могли позволить себе только феодалы.

Но если компьютеры делают нас такими всесильными, почему так неудобно ими пользоваться? Наши отношения с компьютерами, скажем так, сложные. Никто не устраивает вечеринку, когда его ПК успешно излечивается от вирусов и начинает работать быстрее. Хотя, возможно, нам стоило бы. Мощность компьютеров удваивалась почти двадцать раз (то есть они стали в  $2^{20}$  раз мощнее) со времен выхода на рынок Apple Macintosh в 1984 году. Следовательно, теперь ПК могут вместо нас обрабатывать в миллионы раз больше информации. Ну а мы беспрерывно богатеем, справляемся с делами все быстрее и... даже не знаю — хорошеем? Загвоздка лишь одна: нам трудно сказать компьютеру, как ему *распорядиться* всей этой мощью.



**Рис. 1.1.** Правда же, миниатюрные ламы довольно симпатичны? (Фотограф — Лукас Андийон Монтан. <https://www.flickr.com/photos/andion/7249354502>)

## ПОЧЕМУ КОМПЬЮТЕРЫ ВЛАДЕЮТ НАМИ

В результате многие люди бóльшую часть времени используют вычислительные устройства для решения обычных задач, а самую сложную работу — указать компьютерам, что делать, — уже выполнил кто-то другой. Например, отправляя электронное письмо, мы можем сказать ПК, в какой «ящик» доставить сообщение. Но нам не удастся изменить содержание письма в зависимости от времени его доставки, ведь почтовый клиент не дает такой возможности, а как сделать это самостоятельно, мы не знаем. Получается, что мы полностью зависим от отдельных личностей и корпораций, умеющих давать команды компьютерам. У нас с машинами сложились не совсем те отношения, которых нам хотелось бы. Однако для «людей наверху» это хороший вариант. Вот *они* и решили, что пусть лучше компьютеры управляют нами, чем наоборот. Вы можете включить робот-пылесос, чтобы тот начал уборку дома, но не знаете, как помешать ему передавать своему создателю данные о вас, получаемые в ходе работы.

На первых порах большинство таких компаний не собирались тайно проникать в нашу личную жизнь. В 1984 году, когда объявили о выпуске Apple Macintosh, в эпической рекламе, снятой Ридли Скоттом, говорилось:

*24 января Apple Computer представит вашему вниманию Macintosh.  
И вы поймете, почему 1984-й не будет похож на «1984»\*.*

Ох. У меня каждый раз мурашки по коже.

Видите ли, в то время компьютеры еще только становились частью повседневной жизни. Они были новшеством для нас, казались загадочными и чуждыми. (Кстати, «Чужой» — более ранний режиссерский проект Ридли Скотта.) Мы не разбирались в этих устройствах, поэтому никак не могли применять их эффективно. Без программ компьютеры бесполезны, а доступными программами не так-то легко было воспользоваться. В лучшем случае мы получали строго определенный список действий, следуя которому заставляли компьютер трудиться на пользу нам. Это работало,

---

\* «1984» — роман-антиутопия Джорджа Оруэлла, изданный в 1949 году. Главный герой романа работает в Министерстве правды, занимающемся непрерывной фальсификацией различной исторической информации (статистических данных, исторических фактов) на всех уровнях информирования населения: в СМИ, книгах, образовании, искусстве, спорте.

но превращало нас в винтики большой бездушной машины. Образ «1984» казался излишне уместным.

Компания Apple перебросила мост над пропастью между программным обеспечением и людьми, которые хотели запускать программы, сделав так, чтобы этот процесс меньше напоминал написание кода. Программировать — непростое занятие, поэтому, если мы сумеем замаскировать эту сложность, компьютеры станут менее пугающими, и тогда преимуществами технологий воспользуются больше людей. Это сработало! В какой-то мере. Apple продемонстрировала всей индустрии, как сделать компьютеры более удобными, поставив разработчиков между нами и программами. Благодаря такому ходу разработчикам удастся устроить так, что при использовании компьютера вам не кажется, будто вы попали в «1984». С другой стороны, по той же причине разработчиков непреодолимо тянет сделать все как в «1984». Посредничая между нами и компьютерами, они получают власть, а власть развращает.

Я не утверждаю, что все кодеры — плохие. Среди моих лучших друзей есть разработчики программного обеспечения. Вообще говоря, я не думаю, что у них есть возможность выбирать: в сложившейся ситуации они вынуждены стоять между нами и машинами. Ведь люди, которые не умеют программировать, не смогут пользоваться программами, если не скрыть, как они написаны. Очевидно, нам нужно больше информации о действиях компьютеров. Значит, мы должны совершить условный шаг им навстречу. Открыться идее программирования.

Начав думать как программисты, мы сможем требовать, чтобы разработчики делали машины более программируемыми. Однако создавать в своих творениях «точки перехвата», которые дадут нам больше контроля, — это дополнительный труд для разработчиков. Они не возьмутся за дело, пока мы (1) не покажем им, что готовы, и (2) не примемся настаивать. Только тогда ситуация изменится. Это произойдет не сразу. Но если мы сможем постепенно расширить свое участие в процессе управления машинами, то начнем возрождать надлежащую расстановку сил — ту, при которой наши компьютеры служат нам. Революция начинается сегодня!

## **ДАЛЕЕ СЛЕДУЮТ МАТЕРИАЛЫ ДЛЯ ВЗРОСЛЫХ**

Что нужно, чтобы компьютеры понимали наши команды и затем выполняли их? Следует иметь в виду, что ПК — это французские таксисты машинного мира. Если вам требуется, чтобы они довезли вас до

популярных мест вроде Эйфелевой башни или Лувра, проблем не возникнет. Они делают это десятки раз в день, и от вас потребуется только показать картинку нужной достопримечательности. Однако придется немного подучить французский, если вы хотите сказать, что вам надо на Коко-де-Мер, но перед этим следует подбросить вас к отелю и подождать, пока вы сбегаете за курткой, а затем неторопливо проехаться вдоль Сены, чтобы не прибыть на свидание слишком рано — ведь вы не хотите показаться чересчур нетерпеливым. Чем сложнее просьба, тем лучше нужно знать французский.

Вот о чем эта книга — о французском языке. Ой, нет! О том, как получить максимальную пользу от компьютеров, эффективно передавая им свои пожелания. Также мы поговорим о человеческой стороне недавних достижений в области искусственного интеллекта и машинного обучения, предоставивших машинам возможность пойти нам навстречу и более активно участвовать в интерпретации наших намерений. Программирование не получит широкого распространения, если будет выглядеть как написание кода и требовать от нас мыслить как счетные устройства. Оно скорее должно походить на то, как мы объясняем друг другу, куда пойти и что сделать.

**Таблица 1.1.** В этой таблице сопряженности размером  $2 \times 2$  кратко показано, как мы указываем машинам, что делать

	<i>инструкция</i>	<i>стимул</i>
<i>напрямую</i>	рассказывать	объяснять
<i>через пример</i>	показывать	вдохновлять

Существует четыре категории подходов, с помощью которых мы сообщаем машинам о своих желаниях. Они прекрасно сочетаются с четырьмя глаголами из афоризма Уильяма Артура Уорда: рассказывать, объяснять, показывать и вдохновлять. Каждый из этих подходов по-своему успешен, но также имеет существенные ограничения. Мы увидим, что лучший вариант, который позволит всем нам умело применять вычислительную мощь, — создание в будущем компьютерных систем, включающих все четыре метода. Но даже в краткосрочной перспективе эта книга даст вам информацию, благодаря которой вы лучше поймете вездесущих электронных помощников и сможете эффективно работать с ними.

Я расположил эти четыре подхода в таблице  $2 \times 2$ . Вы сразу же понимаете, что видите полную картину, потому что заполнены все четыре ячейки. (Вот так пишут книги настоящие мастера!)