

# Содержание

<b>Введение.</b> Добро пожаловать в Мир-фильтр . . . . .	11
<b>Глава 1.</b> Взлет алгоритмических рекомендаций . . . . .	25
<b>Глава 2.</b> Разрушение личного вкуса . . . . .	68
<b>Глава 3.</b> Алгоритмическая глобализация . . . . .	130
<b>Глава 4.</b> Экономика инфлюэнсеров . . . . .	186
<b>Глава 5.</b> Регулирование Мира-фильтра . . . . .	259
<b>Глава 6.</b> В поисках человеческого курирования . . . . .	315
<b>Заключение</b> . . . . .	387
<i>Благодарности</i> . . . . .	395

# Глава 1

## Взлет алгоритмических рекомендаций

### Первые алгоритмы

*Алгоритм* как термин просто описывает некое уравнение: любую формулу или набор правил, которые выдают желаемый результат. Самые ранние примеры восходят еще к древнему Вавилону, располагавшемуся на территории современного Ирака. На клинописных табличках, датированных 1800–1600 годами до н. э., записаны алгоритмы для таких целей, как вычисление длины и ширины резервуара, если известны его глубина и объем вынутого грунта. По словам математика Дональда Кнута, вавилоняне “представляли каждую формулу в виде пошагового списка правил, то есть в виде алгоритма вычислений по этой формуле”. У них была специальная система записи вычислений, использующая, как пишет Кнут, “не символический язык, а представление формул на «машинном языке»”. Письменное объяснение каждого вавилонского алгоритма заканчивалось одной и той же фразой: “Таков способ”. Эта фраза подчеркивает неотъемлемое свойство алгоритмов: их можно повторять, они одинаково применимы и эффективны каждый раз, когда возникает определенная ситуация. Современный последователь Кремниевой долины назвал бы их адаптируемыми.

Алгоритмы занимают важное место в истории математики. Около 300 года до н. э. греческий философ Евклид

описал в своем труде “Начала” так называемый алгоритм Евклида<sup>1</sup> — способ нахождения наибольшего общего делителя двух чисел. И его, и “решето Эратосфена” — алгоритм III века до н. э., который находит все простые числа, не превосходящие определенного числа, — используют до сих пор, особенно в области криптографии. Однако само слово “алгоритм” происходит от имени определенного человека — или, по крайней мере, от места его рождения.

Персидский ученый Мухаммад ибн Муса аль-Хорезми родился около 780 года в Хорезме — государстве, располагавшемся на территории современных Туркменистана и Узбекистана. О его жизни мало что известно. Он добрался до Багдада, который стал интеллектуальным центром региона после того, как мусульманский халифат Аббасидов завоевал Персию в VII веке. Там ученый занимался астрологией, географией и математикой в Доме мудрости, известном также как Большая библиотека Багдада. Подобно своей предшественнице — египетской Александрийской библиотеке — Дом мудрости являлся междисциплинарным научным центром, где ценили исследования и переводили на арабский язык тексты, написанные на латыни, санскрите, греческом и персидском языках. Около 820 года аль-Хорезми завершил работу “Книга об индийском счете”, которая в конечном итоге привела к распространению в Европе современной системы записи чисел. Он также сочинил трактат о методах решения уравнений “Китаб аль-джебр ва-ль-мукабала” (“Краткая книга о восполнении и противопоставлении”). От слова “аль-джебр” из названия (которое означает “восстановление, восполнение”, то есть сокращение одинаковых членов в обеих частях уравнения) произошло слово “алгебра”. В труде аль-Хорезми описывались методы решения квадратных уравнений и вычисления площади и объема, указывались приближенные значения числа  $\pi$ .

1 Этот алгоритм был известен и до Евклида.

В середине XII века в Испании пересекались мусульманская, иудейская и христианская культуры — иногда мирно, иногда не очень, и между цивилизациями шел обмен идеями. Живший тогда в испанском городе Сеговия английский востоковед Роберт Честерский в 1145 году перевел “Книгу о восполнении и противопоставлении” на латынь. “*Аль-джебр*” превратилось в *algeber*, а “аль-Хорезми” — в *Algoritmi*. В то время слово *algorismus* относилось ко всем математическим операциям с использованием индийско-арабских цифр, а тех, кто занимался подобным искусством, называли алгоритмами. (Этим термином именуют себя визуальные художники, с 1960-х годов использующие алгоритмические процессы, однако он кажется подходящим для всех, кто работает над современной версией алгоритмов.) Такой длинный путь этимологии алгоритма показывает, что вычисления являются не только продуктом воспроизводимых научных законов, но и результатом человеческого искусства и труда.

## Изобретение компьютерного программирования

Работа компьютеров основана на многократно выполняемых операциях. Затем с результатами, закодированными нулями и единицами, производятся новые операции, результат которых выдается пользователю. В 1822 году британский изобретатель Чарльз Бэббидж изложил свою концепцию “применения машин для вычисления астрономических и математических таблиц” — метод автоматизации вычисления с помощью системы пронумерованных шестерней и валиков, названной разностной машиной. Это устройство так и не было построено целиком, однако его более поздние реализации выглядят как внутренности фортепиано, только вместо молоточков тянутся длинные ряды колес. Спроектированная Бэббиджем ма-

шина должна была иметь высоту восемь футов и весить четыре тонны. Затем изобретателю в голову пришла идея аналитической машины, которая могла (если бы ее удалось построить) получать команды, заданные с помощью перфокарт, и выполнять простые операции программирования (циклы и условные операторы). Она стала основой для гораздо более сложных современных вычислительных машин. Сын Бэббиджа Генри в 1888 году писал: “Это всего лишь вопрос карт и времени”.

Дочь лорда Байрона Ада Лавлейс сегодня считается первым программистом: она написала алгоритмы для машины в том виде, в каком ее спроектировал Бэббидж, включая процесс вычисления чисел Бернулли. Лавлейс также поняла, что повторяющиеся механические процессы, на которые способна машина, можно применять не только в математике. В 1843 году Лавлейс писала, что аналитическая машина “могла бы работать с другими вещами, кроме чисел, если найти объекты, взаимоотношения которых можно выразить с помощью абстрактной науки об операциях и которые также можно приспособить к рабочей нотации и механизму машины”. Иными словами, формально можно манипулировать всем, что удастся превратить в нечто похожее на данные — какой-то ряд чисел. Это может быть текст, музыка, искусство или даже игра — например, шахматы. Лавлейс представила себе одну из форм такой автоматизации: “Если предположить, например, что фундаментальные отношения звуков в науке гармонии и музыкальной композиции поддаются такому выражению и адаптации, то машина могла бы сочинять замысловатые научные музыкальные произведения любой степени сложности и объема”. Она предвосхитила нечто, подобное “генеративной музыке”; композитор Брайан Ино предложил этот термин в 1995 году для описания постоянно меняющихся синтезаторных композиций: программное обеспечение создает разные мелодии при каждом запуске программы. Лавлейс представляла, как новая технология может формировать и увековечивать куль-

туру — как это происходит сегодня при алгоритмическом наполнении лент.

Лавлейс одной из первых обнаружила, что манипулирование такими механическими командами может оказаться формой самовыражения. В 1990-х и 2000-х годах компьютерное программирование заняло место рядом с математикой и естественными науками, превратившись в навык, необходимый для полноценного образования ребенка. Я познакомился с ним примерно в 2002 году в компьютерном классе своей школы: мы играли в обучающие видеоигры, напоминающие языки программирования. Но по-настоящему я учился с помощью громоздких графических калькуляторов TI-83, которые нам приходилось приобретать для уроков математики. Эти калькуляторы можно было программировать на языке под названием TI-BASIC, включавшем простые операторы ветвления и функции. Сначала я составлял скромные программы для формул, которые использовались в моих тестах, но когда лучше разобрался в языке, стал создавать свои собственные версии игр “Крестики-нолики” и “Четыре в ряд”. Машина стала партнером в моем творчестве; это казалось настоящим волшебством.

Спустя столетие после Лавлейс британский математик и информатик Алан Тьюринг во время Второй мировой войны работал на правительство — он помог взломать код немецкой шифровальной машины “Энигма”. В 1946 году, когда война закончилась, Тьюринг написал для Национальной физической библиотеки доклад, в котором предложил идею проекта “автоматической вычислительной машины”. Это было первое описание искусственного интеллекта как реальной возможности, а не теоретической концепции. Тьюринг писал, что вычислительные и сортировочные машины, предназначенные для выполнения конкретных задач, уже существуют, однако его предложение выходит и за эти рамки: “Вместо того чтобы постоянно использовать человеческий

труд для вывода материала из машины и ввода его обратно в нужный момент, обо всем будет заботиться сама машина”.

По мнению Тьюринга, подобное устройство могло бы выполнять любые вычисления в любом масштабе, не нуждаясь в перенастройке. У него имелся бы свой внутренний логический язык, который можно было бы приспособлять для разных целей, чтобы решать произвольные задачи. “Как можно ожидать, чтобы машина выполняла все эти многочисленные вещи? — писал Тьюринг. — Ответ заключается в следующем: мы должны считать, что машина делает нечто довольно простое, а именно — выполняет команды, отданные ей в стандартной форме, которую она способна понять”. Она будет следовать алгоритмам. Он предвосхитил то, как современные алгоритмы машинного обучения развиваются с течением времени, внося коррективы без участия человека.

Такая система будет выполнять вычисления гораздо быстрее и на более высоком уровне сложности, превосходя возможности человека. Тьюринг писал: “Скорость машины больше не ограничивается скоростью человека-оператора”. Однако он не считал, что такие машины обеспечат утопическое счастье. То, что они будут работать автоматически, не означает, что они всегда будут правы. “Фактор человеческой ошибки устраняется, однако в какой-то степени его, возможно, заменят механические ошибки”, — продолжал Тьюринг. В своей работе он спрогнозировал многие известные сегодня компоненты персональных компьютеров: память с возможностью стирания, механизмы ввода, преобразование двоичного языка и даже контроль температуры во избежание перегревов. Однако для Тьюринга слово “компьютер” относилось не к машине, а к человеку, выполняющему вычисления, что еще раз подчеркивает этот органический элемент.

Еще в 1936 году Тьюринг придумал то, что сейчас называется “машиной Тьюринга”. Он подробно описал это устройство в эссе 1948 года под названием “Интеллектуальная машина”.

Машина Тьюринга — это “бесконечная лента, разделенная на ячейки, в каждой из которых можно записать какой-то символ”. Лента проходит через управляющее устройство, которое проверяет одну ячейку за раз и выполняет операцию, которая определяется состоянием машины и наблюдаемым символом: записать в ячейку новый символ, перейти в новое состояние и переместиться к соседней ячейке вправо или влево. Машина Тьюринга может реализовать любой алгоритм в историческом смысле слова — алгоритм произвольной математической процедуры. Любая вычислительная система, которая способна делать все, что может машина Тьюринга, называется “полной по Тьюрингу” или “тьюринг-полной”. Например, все языки программирования являются полными по Тьюрингу, потому что могут смоделировать любой вид уравнений. (Даже программа для работы с электронными таблицами *Excel* в 2021 году стала полной по Тьюрингу.) Тьюринг пришел к верному выводу о том, что любая вычислительная машина сумеет выполнить работу любой другой, и даже аналитическая машина Чарльза Бэббиджа XIX века теоретически могла бы выполнять сложные задачи, обрабатываемые нашими ноутбуками, — если бы ей предоставили бесконечные ресурсы и время.

В жизни Тьюринга тоже происходило своеобразное столкновение между механическими правилами и человеческими действиями. В 1952 году математика обвинили в совершении грубой непристойности — в “выражении гомосексуальности” (юридический оборот, означающий секс с другим мужчиной) — в ходе путаного судебного разбирательства, которое он сам инициировал после ограбления собственного дома. Гомосексуальный секс между взрослыми по обоюдному согласию оставался незаконным в Англии вплоть до 1967 года — этот закон является своего рода алгоритмом, выносящим приговор на основе безжалостного набора правил. В итоге Тьюринга признали виновным. Ему предоставили выбор между тюремным заключением и химической кастрацией, и он предпочел

второй вариант. В июне 1954 года домработница нашла 41-летнего Тьюринга мертвым. Причиной смерти стало отравление цианидом, а предполагаемым методом самоубийства являлось надкушенное яблоко на прикроватной тумбочке<sup>1</sup>.

Когда мы говорим об “алгоритме”, часто возникает ощущение, что эта сила появилась совсем недавно, в эпоху социальных сетей. Однако на самом деле речь идет о технологии, имеющей богатую историю и наследие, которые медленно формировались на протяжении целых веков до появления интернета. Восстановление этой более общей картины может помочь нам лучше понять силу, которой обладают современные алгоритмы. И все же, каким бы сложным ни был алгоритм, по своей сути он остается неким уравнением: методом, позволяющим получить желаемый результат — будь то шумерская схема, позволяющая разделить некоторое количество зерна поровну между несколькими мужчинами, или лента Фейсбука, определяющая, какое сообщение показывается первым, когда вы заходите на сайт. Все алгоритмы — это двигатели автоматизации, и, как прогнозировала Ада Лавлейс, автоматизация теперь вошла во многие сферы нашей жизни, не ограничиваясь чистой математикой.

## Алгоритмическое принятие решений

В 1971 году в одном из офисных зданий в центре города Сантьяго (Чили) появилась шестиугольная комната, спроектированная как своеобразное диспетчерское помещение для всей страны. На обшитых деревянными панелями стенах висели мониторы и экраны, на которые выводились различные показатели — например, поставки сырья и уровень занятости населения. В центре комнаты по кругу стояли семь белых

<sup>1</sup> Несмотря на распространенность этой теории, в действительности яблоко не проверяли на наличие яда.

стеклопластиковых кресел, напоминавших место капитана космического крейсера из научной фантастики. С правой стороны от каждого располагалась панель управления экранами; имелись также пепельница и держатель для стакана — возможно, для виски. Этот диспетчерский центр — часть масштабного проекта “Киберсин”<sup>1</sup> — был спроектирован при социалистическом президенте Чили Сальвадоре Альенде под руководством Стаффорда Бира — британского специалиста по исследованию операций, который у себя в стране применял методы “кибернетики” для управления бизнесом. Бир описывал кибернетику как “науку управления”. Она включает в себя анализ сложных систем, будь то корпорации или биология, и изучение их работы для моделирования или создания подобных интеллектуальных самокорректирующихся систем. (В США аналогичную практику анализа систем впервые применила корпорация RAND в 1950-х годах.) Предполагалось, что проект “Киберсин” станет идеальной моделью, помогающей чилийскому правительству принимать решения в режиме реального времени, пока они сидят в помещении, курят сигареты и пьют виски — очередная встреча холодного технологического и хаотичного человеческого. Из этого центра люди наблюдали за алгоритмами, которые управляли государством.

Физический дизайн проекта “Киберсин”, разработанный под руководством немецкого консультанта Ги Бонсипе, создавал образ утопии в духе модерна середины века. Провода, соединявшие мониторы на стенах и сиденья, были скрыты от глаз. Сами гладкие одинаковые кресла отличались искривленной плавностью и отсутствием видимых швов. Комната символически сводила управление страной к манипулированию данными — словно к победе в какой-то видеоигре. Проект “Киберсин” обещал заменить человеческое руководство техно-

1 Слово *Cybersyn* образовано от *cybernetics* (“кибернетика”) и *synergy* (“синергия”, то есть усиление суммарного воздействия по сравнению с воздействием отдельных факторов).

логическим надзором, а скромный ряд экранов предоставлял любую требуемую информацию. Можно было просто усесться в одно из кресел и наблюдать за всем, что происходит в стране.

Однако технология проекта “Киберсин” являлась лишь фасадом, своеобразным фантастическим дизайн-концептом — интерактивной иллюзией возможного. Компьютерные сети того времени не позволяли реализовать его потенциал. Вывод данных осуществлялся вручную, а не автоматически. Центр использовал единственный компьютер, получавший информацию от чилийских заводов с помощью телексных аппаратов через телефонные линии. И, наконец, хотя центр и закончили, его так и не запустили<sup>1</sup>. Одиннадцатого сентября 1973 года правительство Альенде было свергнуто при поддержке ЦРУ, и к власти пришел Аугусто Пиночет.

Фотографии проекта “Киберсин” по-прежнему обладают неоспоримой притягательностью. Они снова и снова мелькают на дизайнерских досках настроения, демонстрируя эстетику, которая даже спустя много десятилетий все еще выглядит футуристично. Возможно, эти изображения впечатляют по той причине, что мы сохраняем мечту обработки реальных исходных данных и переработки их в цифровые схемы, после оценивания которых определяется правильный путь действий. Проект “Киберсин” излучал ощущение непогрешимости, хотя изобретатели уровня Тьюринга понимали, что компьютеры не могут работать настолько идеально. Как утверждал пионер кибернетики Стаффорд Бир, мы стремимся использовать машины для автоматизации уже существующих структур и процессов, которые изначально являлись творениями человека. Указывая на этот парадокс, Бир писал в своей книге 1968 года “Наука управления”: “Мы закрепляем в стали, стекле и полупроводниках те самые ограничения руки, глаза и мозга, для преодоления которых и был

<sup>1</sup> Система “Киберсин” все же успела поработать, но после прихода к власти Пиночета ее забросили.

изобретен компьютер”. Как и в случае с механическим турком, внутри машины остается человек.

Сегодня мы уже располагаем вариантами алгоритмического правительства и алгоритмической жизни: банки прибегают к машинам, чтобы определить, кто получит кредит; *Spotify* использует данные о ваших прошлых действиях, чтобы рекомендовать те песни, которые лучше всего подходят вашей душевной организации. Однако технология, которая все это обеспечивает, не похожа на проект “Киберсин”. У нее нет шестиугольных комнат и кресел с подлокотниками. Алгоритмы стали одновременно невидимыми и вездесущими, они содержатся в приложениях, которые мы носим с собой на телефонах, хотя их данные физически хранятся где-то далеко, на огромных серверных фермах с градирнями, расположенных в малозаметных местах на природе. Если в проекте “Киберсин” предполагалось, что мир, управляемый данными, может быть последовательным и понятным и его можно заключить в рамки помещения диспетчерского центра, то теперь мы знаем, что он абстрактный и диффузный, он везде и нигде одновременно. Нас побуждают забыть о наличии алгоритмов.

Новые технологии неизбежно порождают новые формы поведения, но это поведение редко совпадает с тем, которого ожидают изобретатели. Технология обладает собственным смыслом, который в конечном итоге выходит на первый план. Маршалл Маклюэн написал знаменитый афоризм “Средство коммуникации есть сообщение”<sup>1</sup> в своей книге 1964 года “Понимание медиа: внешние расширения человека”. Он имел в виду, что структура нового средства передачи информации — электрического света, телефона, телевидения — важнее содержания, которое через него передается. Сама способность телефона соединять людей превосходит любой конкретный разговор. Маклюэн писал: “Ибо «сообщением» любого сред-

<sup>1</sup> Здесь и далее цитаты даны по изданию: МАКЛЮЭН М. *Понимание медиа: внешние расширения человека*. М.: Кучково поле, 2007. Перевод В. Николаева.

ства коммуникации, или технологии, является то изменение масштаба, скорости или формы, которое приносится им в человеческие дела”. В нашем случае средством передачи является алгоритмическая лента; она масштабирует и ускоряет взаимосвязь людей по всему миру до невообразимой степени. Смысл ее функции заключается в том, что на каком-то уровне наши коллективные потребительские привычки, переведенные в данные, приводят к одинаковости.