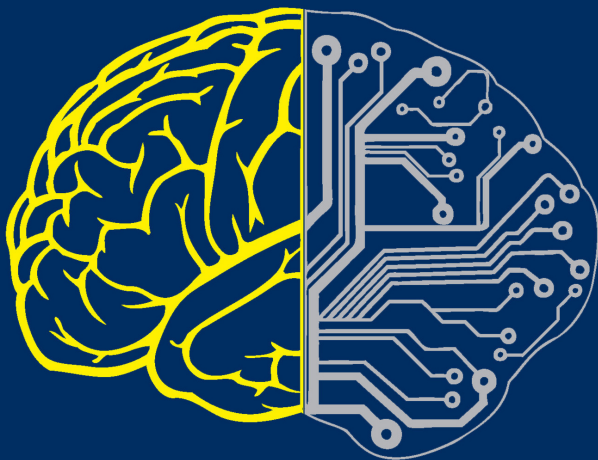


**ДЖОН ФОН  
НЕЙМАН**



**ВЫЧИСЛИТЕЛЬНАЯ  
МАШИНА  
И МОЗГ**

**Джон фон Нейман**  
**Вычислительная машина и мозг**  
**Серия «Наука: открытия**  
**и первооткрыватели»**

*[http://www.litres.ru/pages/biblio\\_book/?art=40071408](http://www.litres.ru/pages/biblio_book/?art=40071408)*  
*Вычислительная машина и мозг: АСТ; Москва; 2018*  
*ISBN 978-5-17-111540-1*

**Аннотация**

Информационные технологии кардинально изменили жизнь человека, затронув каждую ее сторону – от бизнеса и науки до политики и искусства. Искусственный интеллект перестал быть фантастикой и сделался реальностью. И именно эта всесторонняя компьютеризация общества поставила перед нами новый фундаментальный вопрос: какова разница между интеллектом искусственным и интеллектом человеческим и есть ли она вообще? Джон фон Нейман в своей поистине пророческой, выдержавшей испытание временем и до сих пор регулярно переиздающейся книге утверждает: такая разница минимальна. Несмотря на все различия в архитектуре и строительных блоках мозга и вычислительной машины, искусственный интеллект тем не менее способен имитировать работу мозга.

# Содержание

Силлимановские лекции	6
Предисловие к третьему изданию	7
Конец ознакомительного фрагмента.	20

# Джон фон Нейман

## Вычислительная машина и мозг



John Von Neumann

THE COMPUTER AND THE BRAIN

Third edition

Originally published by Yale University Press First edition 1958. Second edition published as a Yale Nota Bene book in 2000. Third edition 2012.

© Yale University Press, 1958

© Renewed by Marina V. N. Whitman, 1986

Foreword to the Second edition copyright © 2000 by Yale University

Foreword to the Third edition copyright © 2012 by Ray Kurzweil

Школа перевода В. Баканова, 2018

© Издание на русском языке AST Publishers, 2018

# Силлимановские лекции

На средства, пожертвованные в память о Хепсе Эли Силлимане, президент и сотрудники Йельского университета ежегодно представляют цикл лекций, призванных указать на присутствие Божие и промысел Божий в том виде, в коем проявляются они в мире природы и нравственности. По искреннему убеждению завещателя, всякое упорядоченное изложение явлений природы или фактов истории способствует достижению этой цели более успешно, нежели догматическое или полемическое богословие, которое посему должно быть исключено из материалов лекций. Темы преимущественно выбираются из областей естественных наук и истории; особое внимание уделено астрономии, химии, геологии и анатомии.

# Предисловие к третьему изданию

Информационные технологии кардинально изменили жизнь человека, затронув каждую ее сторону – от бизнеса до политики и искусства. Сегодня, в силу стремительного роста мощности компьютеров, с одной стороны, и одновременно-го улучшения соотношения «производительность – цена» – с другой, информационная эра неумолимо продолжает расширять сферу своего влияния. Учитывая данное обстоятельство, можно утверждать, что важнейшим информационным процессом, требующим всестороннего изучения, является сам человеческий интеллект. Вероятно, эта книга представляет собой первое серьезное исследование взаимосвязи между мышлением человека и машиной – исследование с точки зрения математика, сформулировавшего фундаментальную архитектуру компьютерной эры.

В рамках грандиозного проекта по изучению механизмов работы мозга достигнуты значительные успехи: мы достаточно далеко продвинулись в обратном проектировании парадигм человеческого мышления и сегодня применяем их для создания более умных машин. Разумно ожидать, что искусственный интеллект (ИИ), построенный на базе данных принципов, в итоге воспарит над человеческим разумом. Однако, я полагаю, цель этого дерзкого предприятия состоит не в том, чтобы потеснить нас, а в том, чтобы рас-

ширить границы того, что уже сейчас можно смело назвать человеко-машинной цивилизацией. Именно это делает человеческий род поистине уникальным.

Итак, каковы же ключевые идеи, лежащие в основе информационной эры? По моему мнению, их пять. Джон фон Нейман выдвинул три из них, а также внес весомый вклад в четвертую. Клод Шеннон решил фундаментальную проблему надежности информации. Алан Тьюринг сформулировал идею универсальности машинных вычислений. Опираясь на Тьюринга и Шеннона, фон Нейман создал машину фон Неймана, которая была и остается фундаментальной архитектурой всех вычислительных машин, существующих на сегодняшний день.

В обманчиво скромном томике, который вы держите в руках, фон Нейман описывает предложенную им модель машинных вычислений и постулирует принципиальную эквивалентность живого мозга и искусственной машины. Он признает глубокие структурные различия, однако затем обращается к принципу эквивалентности всех вычислений, предложенному Тьюрингом, и предлагает рассматривать методы работы мозга как совокупность вычислений. Книга в высшей степени провидческая, особенно если учесть, что она была написана более полувека назад, когда нейронаука располагала лишь самыми примитивными инструментами. Фон Нейман предчувствовал быстрое развитие технологий и видел его неизбежные последствия в грядущей сингулярной транс-



формации человеческого существования.

Рассмотрим эти пять основных идей более подробно.

В 1940 году, услышав слово «computer», люди понимали, что речь идет об аналоговой вычислительной машине. В аналоговых машинах числа выражались разными уровнями напряжения, а специализированные компоненты могли выполнять такие арифметические функции, как сложение и умножение. Главный недостаток подобных машин заключался в весьма низкой степени точности. Числа могли быть представлены с точностью до одной сотой, а так как уровни напряжения, представляющие эти числа, обрабатывались большим количеством арифметических операторов, ошибки имели тенденцию накапливаться. При выполнении достаточно длинных вычислений результаты оказывались в высшей степени неточными, а то и вообще бессмысленными.

Это явление знакомо всякому, кто хоть раз записывал музыку на магнитную ленту. Первая копия была немного шумнее оригинала (шум представляет собой случайные неточности). На второй копии (т. е. копии копии) шумов было еще больше, а десятая содержала один только шум.

Это наводит на мысль, что с той же проблемой столкнется и мир цифровых вычислительных машин. Рассмотрим передачу цифровой информации по некоему каналу. Поскольку идеальных каналов не существует, любому из них изначально будет присущ определенный уровень ошибок. Допустим, мы выбрали канал с вероятностью правильной пере-

дачи каждого бита 0,9. Если я отправлю сообщение длиной в один бит, вероятность точной передачи его через этот канал будет равна 0,9. Допустим, я отправил 2 бита. Теперь точность составит  $0,9^2 = 0,81$ . А если я отправлю 1 байт (8 битов)? В таком случае вероятность, что я получу его правильно, окажется менее 0,5 (0,43, если быть точным). Вероятность точной отправки пяти байтов составит около 1 процента.

На первый взгляд наилучший способ обойти эту проблему – сделать канал более точным. Предположим, канал совершает всего одну ошибку на миллион битов. Если я отправлю файл размером в полмиллиона байт (примерно столько «весит» скромная программка или база данных), вероятность правильной его передачи составит менее 2 процентов, хотя изначально точность канала была весьма высока. Учитывая, что ошибка в один бит может испортить всю компьютерную программу, это не лучший выход. Вне зависимости от исходной точности канала вероятность ошибки стремительно возрастает с увеличением размера сообщения. Выходит, эта проблема неразрешима?

Аналоговые вычислительные машины тоже аккумулируют неточности, но если ограничиться небольшим набором вычислений, они оказываются весьма полезными. Цифровые машины, с другой стороны, требуют наличия постоянной связи не только между двумя разными машинами, но и между собственными составными частями. Так, память свя-

зана с центральным процессором, внутри которого происходит постоянный обмен данными между регистрами, а также между регистрами и арифметическим устройством. Внутри арифметического устройства информация передается от одного битового регистра к другому. Если считать, что частота ошибок стремительно возрастает с увеличением количества таких связей и что ошибка в один бит может нарушить весь процесс, то необходимо признать: цифровые машины обречены.

Таково было общее мнение, пока Шеннон не провозгласил первую ключевую идею информационной эры. Он продемонстрировал, что мы можем создавать произвольно точные сообщения, используя самые ненадежные каналы передачи информации. В своей знаковой статье «Математическая теория связи», опубликованной в журнале *Bell System Technical Journal* в июле и октябре 1948 года, Шеннон предложил теорему кодирования для каналов с шумами, которая гласила: если у вас есть доступный канал с любым коэффициентом ошибок (за исключением 50 процентов на бит, поскольку это означает, что канал передает чистый шум), вы можете передать сообщение с любой желаемой степенью точности. Другими словами, частота появления ошибок может составлять один бит из  $n$  бит, где  $n$  может быть сколь угодно большим. Например, если у вас есть канал, который правильно передает биты информации только в 51 проценте случаев (т. е. передает правильный бит чутьочку чаще, чем неправильный),

вы тем не менее можете передавать сообщения таким образом, что неправильным окажется только один бит из миллиона, один бит из триллиона или один бит из триллиона триллионов.

Как такое возможно? Ответ: через избыточность.

Сейчас подобное решение может показаться элементарным, но в то время оно было далеко не очевидным. Рассмотрим простой пример. Если я передам каждый бит три раза и применю «принцип большинства», то я значительно повышу надежность результата. Если этого будет недостаточно, я могу увеличить избыточность, пока не получу необходимую надежность. Повторение информации – самый простой способ достичь высокой точности в каналах низкой точности. Однако данный подход не самый эффективный. В своей статье Шеннон не только заложил основы теории информации, но и предложил оптимальные методы обнаружения и исправления ошибок. Эти методы позволяли добиться любой желаемой точности через любой неслучайный канал.

Читатели более старшего возраста наверняка помнят телефонные модемы, в которых постоянно что-то шипело и щелкало, так как они передавали информацию по аналоговым линиям с высоким уровнем помех. Однако те же модемы могли передавать цифровые данные с очень высокой точностью – благодаря теореме Шеннона для канала с шумами.

Аналогичная проблема и аналогичное решение существуют и для цифровой памяти. Вы когда-нибудь задумывались,

почему CD, DVD и программные диски продолжают работать даже после того, как упали на пол или были поцарапаны? Этим мы тоже обязаны Шеннону. Процесс вычислений состоит из трех элементов: связи (которая, как я уже говорил, имеет место как внутри, так и между машинами), памяти и логических вентилей (которые выполняют арифметические и логические функции). Точность логических вентилей можно сделать произвольно высокой с помощью особых кодов для обнаружения и исправления ошибок. Именно благодаря теореме Шеннона мы можем обрабатывать большие и сложные цифровые данные, используя для этого достаточно длинные алгоритмы.

Вторая важная идея, на которую опирается наш информационный век, – универсальность машинных вычислений. В 1936 году Алан Тьюринг описал «машину Тьюринга» – абстрактную вычислительную машину, которая состоит из бесконечно длинной ленты, разделенной на клетки с цифрами 1 или 0. Машина считывает одну клетку за другой и содержит набор правил в виде пронумерованных состояний, фактически представляющих собой хранимую в памяти программу. Каждое правило предписывает машине совершить одно действие, если в считываемой клетке стоит 0, и другое действие, если в считываемой клетке стоит 1. Возможные действия включают запись 0 или 1, перемещение ленты на одну клетку вправо или влево, остановку ленты. Каждое состояние содержит номер следующего состояния, в которое долж-

на перейти машина. Завершив алгоритм, машина останавливается; выход процесса остается на ленте. Хотя лента теоретически бесконечна, любая программа (которая не подразумевает бесконечный цикл) использует конечную часть ленты; следовательно, если мы ограничимся конечной памятью, машина по-прежнему сможет решать широкий круг задач.

В машине Тьюринга нет ничего сложного, верно? На самом деле именно этого и добивался ученый. Он хотел, чтобы его машина была максимально простой (но не проще, перефразируя Эйнштейна). Позже Тьюринг и его бывший учитель, Алонзо Черч сформулировали тезис Черча – Тьюринга, согласно которому задача, которая не может быть решена машиной Тьюринга, не может быть решена никакой *другой* машиной. Хотя собственно машина Тьюринга способна выполнять крайне ограниченное количество команд и одновременно обрабатывает всего один бит, она может вычислить все, что может вычислить любая вычислительная машина.

Строгие интерпретации тезиса Черча – Тьюринга предполагают принципиальную эквивалентность того, что человек может думать или знать, и того, что может быть вычислено машиной. Основная идея заключается в том, что человеческий мозг подчиняется естественным законам; следовательно, его способность обрабатывать информацию не может превосходить таковую у машины (и соответственно у машины Тьюринга).

В своей статье Тьюринг заложил теоретические основы

машинных вычислений. Хотя это целиком и полностью его заслуга, важно отметить, что большое влияние на него оказала лекция, прочитанная Джоном фон Нейманом в 1935 году в Кембридже (Англия). Лекция была посвящена идее программы, которую можно хранить в памяти – концепция, позднее воплощенная в машине Тьюринга. На фон Неймана в свою очередь произвела глубочайшее впечатление статья Тьюринга 1936 года, где были изложены принципы машинных вычислений и которую в конце 1930-х – начале 1940-х годов он включил в список обязательной литературы, составленный для своих коллег.

В той же работе Тьюринг сообщает о другом неожиданном открытии, а именно – о проблеме неразрешимых задач. Неразрешимые задачи – это хорошо описанные задачи с однозначным ответом, который, однако, не может быть вычислен на машине Тьюринга (т. е. на *любой* машине). Это противоречит постулату XIX века, гласящему, что все задачи, которые могут быть описаны, в конечном счете будут решены. Тьюринг показал, что неразрешимых задач столько же, сколько и разрешимых. В своей «Теореме о неполноте» 1931 года Курт Гедель приходит к аналогичному выводу. Таким образом мы оказываемся в странной ситуации: с одной стороны, мы можем описать задачу и доказать, что однозначный ответ существует, а с другой – знаем, что ответ никогда не будет найден.

Гораздо больше можно сказать о философском значении

работ Тьюринга, Черча и Геделя, однако в рамках данного предисловия достаточно отметить следующее. Тьюринг показал, что в основе всех машинных вычислений лежит очень простой механизм. Поскольку машина Тьюринга (и, следовательно, любая вычислительная машина) способна определять дальнейший образ действий, исходя из результатов предыдущих операций, она способна принимать решения и моделировать произвольно сложные иерархии данных.

К декабрю 1943 года Тьюринг спроектировал и построил «Колосс», который часто называют первым компьютером в истории электронных вычислительных машин. «Колосс» предназначался для выполнения одной-единственной задачи – декодирования радиосообщений, зашифрованных нацистской машиной Enigma – и не мог быть перепрограммирован для выполнения другой задачи. Зато с дешифровкой он справлялся блестяще: считается, что именно благодаря ему союзники смогли взять верх над немецкими «люфтваффе» и выиграть решающую битву за Британию.

Все вышеизложенное послужило своеобразным фундаментом, на котором Джон фон Нейман создал архитектуру современного компьютера – машину фон Неймана, составляющую основу практически каждого автомата, изобретенного за последние шестьдесят шесть лет: от микроконтроллера стиральных машин до самых больших суперкомпьютеров. Это третья ключевая идея информационной эры.



В статье «Первый проект отчета о EDVAC»,<sup>1</sup> датированной 30 июня 1945 года, фон Нейман изложил революционные принципы в области машинных вычислений. Модель фон Неймана состоит из центрального процессора, в котором выполняются арифметические и логические операции, блока памяти, в котором хранятся программа и данные, устройства массовой памяти, счетчика команд и каналов ввода-вывода. Данная концепция описана в первой части этой книги. Хотя статья фон Неймана, по сути, представляла собой внутренний проектный документ, она не только стала библией для конструкторов вычислительных машин 1940-х и 1950-х годов, но и значимо повлияла на архитектуру всех компьютеров, построенных с тех пор.

Машина Тьюринга не была рассчитана на практическое применение. Теоремы Тьюринга не имели никакого отношения к эффективности решения задач; они лишь позволяли определить круг тех задач, которые могли быть решены посредством машинных вычислений. Фон Нейман, наоборот, ставил своей целью формулирование практических принципов вычислительной машины. Так, в машине фон Неймана однокбитовые вычисления Тьюринга заменены многобитовыми словами (как правило, количество битов кратно восьми). Если машина Тьюринга тратит огромное количество време-

---

<sup>1</sup> EDVAC (Electronic Discrete Variable Automatic Computer) – одна из первых электронных вычислительных машин на двоичной основе. – *Здесь и далее примеч. ред.*

ни на перемещение ленты вперед и назад, чтобы сохранять и извлекать промежуточные результаты, то машина фон Неймана, напротив, снабжена памятью с произвольным доступом, поэтому любой элемент данных может быть извлечен немедленно.

Одним из важнейших принципов, предложенных фон Нейманом, является принцип хранимой программы, который он сформулировал десятью годами ранее: программа хранится в той же памяти, что и данные. Это позволяет перепрограммировать вычислительную машину для выполнения различного рода задач, а также использовать самомодифицирующийся код, допускающий применение некоторых форм рекурсии. До того времени практически все вычислительные машины, включая «Колосс» Тьюринга, могли решать только те задачи, для решения которых были предназначены изначально. Хранимая программа придала вычислительной машине поистине универсальный характер. Так, идея Тьюринга об универсальной машине получила реальное воплощение.

Другой ключевой принцип фон Неймана заключается в том, что каждая инструкция должна содержать код арифметической или логической операции, которую необходимо выполнить, а также адрес операнда в памяти. Данная формула впервые была предложена фон Нейманом в рамках его публикации, посвященной совместному с Дж. Преспером Эккертом и Джоном Мокли проекту EDVAC. Сам EDVAC

был запущен только в 1951 году; к тому времени уже существовали другие вычислительные машины с хранимыми программами, в частности Манчестерская малая экспериментальная машина, ENIAC,<sup>2</sup>

---

<sup>2</sup> ENIAC (русс. ЭНИАК) – электронный числовой интегратор и вычислитель, сокр. от Electronic Numerical Integrator and Computer, первый электронный цифровой вычислитель общего назначения, который можно было перепрограммировать для решения широкого спектра задач.

# Конец ознакомительного фрагмента.

Текст предоставлен ООО «ЛитРес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на ЛитРес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.