



машина Тьюринга

Машина **Тьюринга** — это математическая модель вычислений, описывающая абстрактную машину^[1], которая манипулирует символами на полоске ленты в соответствии с таблицей правил.^[2] Несмотря на простоту модели, она способна реализовать любой компьютерный алгоритм.^[3]

Машина работает на бесконечной^[4] ленте памяти, разделенной на дискретные ячейки,^[5] каждая из которых может содержать один символ, взятый из конечного набора символов, называемого алфавитом машины. У нее есть «головка», которая в любой момент работы машины располагается над одной из этих ячеек, и «состояние», выбранное из конечного набора состояний. На каждом шаге своей работы головка считывает символ в своей ячейке. Затем, основываясь на символе и собственном текущем состоянии машины, машина записывает символ в ту же ячейку и перемещает головку на один шаг влево или вправо,^[6] или останавливает вычисление. Выбор того, какой заменяющий символ записать, в каком направлении переместить головку и следует ли останавливать, основан на конечной таблице, которая определяет, что делать для каждой комбинации текущего состояния и считываемого символа. Как и в случае с реальной компьютерной программой, машина Тьюринга может войти в бесконечный цикл, который никогда не остановится.

Машина Тьюринга была изобретена в 1936 году Аланом Тьюрингом,^{[7][8]} который назвал ее «a-machine» (автоматическая машина).^[9] Именно научный руководитель Тьюринга, Алонзо Чёрч, позже ввел термин «машина Тьюринга» в своем обзоре.^[10] С помощью этой модели Тьюринг смог ответить на два вопроса отрицательно:

- Существует ли машина, способная определить, является ли любая произвольная машина на ее ленте «циклической» (например, зависает или не может продолжить выполнение вычислительной задачи)?
- Существует ли машина, способная определить, печатает ли любая произвольная машина на своей ленте заданный символ?^{[11][12]}

Таким образом, предоставив математическое описание очень простого устройства, способного производить произвольные вычисления, он смог доказать свойства вычислений в целом и, в частности, невычислимость Entscheidungsproblem, или «проблемы принятия решений» (является ли каждое математическое утверждение доказуемым или опровергаемым).^[13]

Машины Тьюринга доказали существование фундаментальных ограничений мощности механических вычислений.^[14]

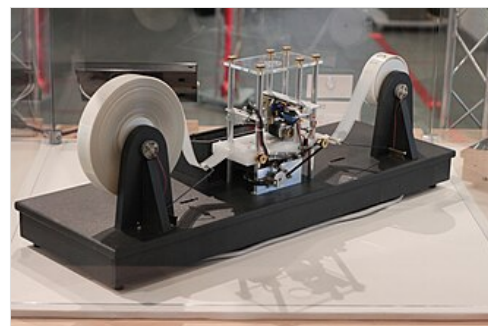
Хотя они могут выполнять произвольные вычисления, их минималистский дизайн делает их слишком медленными для практических вычислений: реальные компьютеры основаны на других конструкциях, которые, в отличие от машин Тьюринга, используют память с произвольным доступом.

Полнота по Тьюрингу — это способность вычислительной модели или системы инструкций имитировать машину Тьюринга. Язык программирования, полный по Тьюрингу, теоретически способен выражать все задачи, выполнимые компьютерами; почти все языки программирования являются полными по Тьюрингу, если игнорировать ограничения конечной памяти.

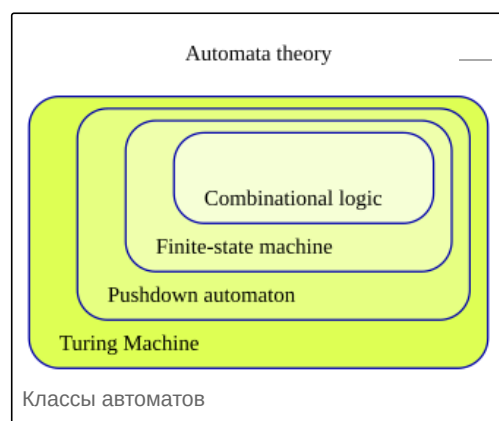
Обзор

Машина Тьюринга — это идеализированная модель центрального процессора (ЦП), который управляет всеми манипуляциями с данными, выполняемыми компьютером, при этом каноническая машина использует последовательную память для хранения данных. Обычно последовательная память представляется в виде ленты бесконечной длины, на которой машина может выполнять операции чтения и записи.

В контексте формальной теории языка машина Тьюринга (автомат) способна перечислять некоторое произвольное подмножество допустимых строк алфавита. Набор строк, который может быть перечислен таким образом, называется рекурсивно перечислимым языком. Машину Тьюринга можно эквивалентно определить как модель, которая распознает



Физическая модель машины Тьюринга, созданная Майком Дэйви. Настоящая машина Тьюринга имела бы ленту бесконечной длины; однако физические модели могут иметь только конечное количество ленты.



допустимые входные строки, а не перечисляет выходные строки.

При наличии машины Тьюринга M и произвольной строки s , как правило, невозможно решить, произведет ли M в конечном итоге s . Это связано с тем, что проблема остановки неразрешима, что имеет серьезные последствия для теоретических пределов вычислений.

Машина Тьюринга способна обрабатывать неограниченную грамматику, что также подразумевает, что она способна надежно оценивать логику первого порядка бесконечным числом способов. Это хорошо демонстрируется с помощью лямбда-исчисления.

Машина Тьюринга, способная имитировать любую другую машину Тьюринга, называется универсальной машиной Тьюринга (УТМ или просто универсальной машиной). Другой математический формализм, лямбда-исчисление, с похожей «универсальной» природой был введен Алонзо Чёрчем. Работы Чёрча переплелись с работами Тьюринга, чтобы сформировать основу тезиса Чёрча–Тьюринга. Этот тезис утверждает, что машины Тьюринга, лямбда-исчисление и другие подобные формализмы вычислений действительно охватывают неформальное понятие эффективных методов в логике и математике и, таким образом, предоставляют модель, с помощью которой можно рассуждать об алгоритме или «механической процедуре» математически точным способом, не привязываясь к какому-либо конкретному формализму. Изучение абстрактных свойств машин Тьюринга дало много идей в области компьютерной науки, теории вычислимости и теории сложности.

Физическое описание

В своем эссе 1948 года «Интеллектуальные машины» Тьюринг писал, что его машина состоит из:

...неограниченная емкость памяти, полученная в виде бесконечной ленты, размеченной на квадраты, на каждом из которых может быть напечатан символ. В любой момент в машине есть один символ; он называется сканируемым символом. Машина может изменять сканируемый символ, и ее поведение частично определяется этим символом, но символы на ленте в других местах не влияют на поведение машины. Однако ленту можно перемещать вперед и назад через машину, что является одной из элементарных операций машины. Поэтому любой символ на ленте может в конечном итоге иметь иннингс. ^[15]

— Тьюринг 1948, стр. 3 ^[16]

Описание

Машина Тьюринга математически моделирует машину, которая механически работает на ленте. На этой ленте находятся символы, которые машина может считывать и записывать по одному за раз с помощью головки ленты. Работа полностью определяется конечным набором элементарных инструкций, таких как «в состоянии 42, если видимый символ равен 0, записать 1; если видимый символ равен 1, перейти в состояние 17; в состоянии 17, если видимый символ равен 0, записать 1 и перейти в состояние 6» и т. д. В оригинальной статье («О вычислимых числах, с приложением к Entscheidungsproblem», см. также ссылки ниже) Тьюринг представляет себе не механизм, а человека, которого он называет «компьютером», который рабски выполняет эти детерминированные механические правила (или, как говорит Тьюринг, «бессистемно»).

Более конкретно, машина Тьюринга состоит из:

- Лента, разделенная на ячейки, расположенные одна за другой. Каждая ячейка содержит символ из некоторого конечного алфавита. Алфавит содержит специальный пустой символ (здесь записанный как «0») и один или несколько других символов. Предполагается, что лента может быть произвольно удлинена влево и вправо, так что машина Тьюринга всегда снабжена таким количеством ленты, которое ей необходимо для вычислений. Предполагается, что ячейки, которые ранее не были записаны, заполнены пустым символом. В некоторых моделях лента имеет левый конец, помеченный специальным символом; лента удлиняется или может быть удлинена до бесконечности вправо.
- Головка, которая может считывать и записывать символы на ленте и перемещать ленту влево и вправо на одну (и только одну) ячейку за раз. В некоторых моделях головка движется, а лента неподвижна.
- Регистр состояний, который хранит состояние машины Тьюринга, одно из конечного множества. Среди них есть особое *начальное состояние*, с которым инициализируется регистр состояний. Эти состояния, пишет Тьюринг, заменяют «состояние ума», в котором обычно находится человек, выполняющий вычисления.
- Конечная *таблица* ^[17] инструкций ^[18], которая, учитывая *состояние* (q_i), в котором в данный момент находится машина, и символ (a_j), который она считывает с ленты (символ, который в данный момент находится

q_4						
	s_1	s_1	s_3	s_1	s_0	s_1

Головка всегда находится над определенным квадратом ленты; показан только конечный отрезок квадратов. Состояние машины (q_4) показано над квадратом, который должен быть обработан. (Рисунок по Клини (1952) стр. 375.)

под заголовком), сообщает машине о необходимости выполнить следующие действия *в последовательности* (для моделей с 5 кортежами):

1. Либо сотрите, либо напишите символ (заменяя j на j_1).
2. Перемещение головы (которое описывается d_k и может иметь значения: «L» для одного шага влево *или* «R» для одного шага вправо *или* «N» для того, чтобы оставаться на том же месте).
3. Предположим то же самое или *новое состояние*, как предписано (перейти в состояние q_{j_1}).

В моделях 4-кортежа стирание или запись символа (a_{j_1}) и перемещение головки влево или вправо (d_k) задаются как отдельные инструкции. Таблица сообщает машине (ia) стереть или записать символ *или* (ib) переместить головку влево или вправо, *а затем* (ii) принять то же самое или новое состояние, как предписано, но не оба действия (ia) и (ib) в одной инструкции. В некоторых моделях, если в таблице нет записи для текущей комбинации символа и состояния, то машина остановится; другие модели требуют заполнения всех записей.

Каждая часть машины (то есть ее состояние, наборы символов и используемая лента в любой момент времени) и ее действия (такие как печать, стирание и движение ленты) являются *конечными*, *дискретными* и *различимыми*; именно неограниченный объем ленты и времени выполнения дает ей неограниченный объем дискового пространства.

Формальное определение

Следуя Хопкрофту и Ульману (1979, стр. 148), (одноленточную) машину Тьюринга можно формально определить как 7-кортежную $M = \langle Q, \Gamma, b, \Sigma, \delta, q_0, F \rangle$ где

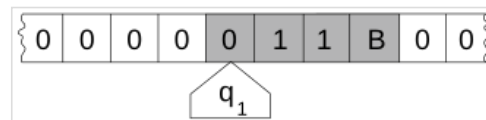
- Γ — конечный, непустой набор *символов ленточного алфавита*;
- $b \in \Gamma$ — *пустой символ* (единственный символ, которому разрешено появляться на ленте бесконечно часто на любом этапе вычислений);
- $\Sigma \subseteq \Gamma \setminus \{b\}$ — набор *входных символов*, то есть набор символов, которым разрешено появляться в исходном содержимом ленты;
- Q — конечное, непустое множество *состояний*;
- $q_0 \in Q$ — *начальное состояние*;
- $F \subseteq Q$ — это набор *конечных состояний* или *принимающих состояний*. Начальное содержимое ленты считается *принятым* M если он в конечном итоге остановится в состоянии от F .
- $\delta : (Q \setminus F) \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$ является *частичной функцией*, называемой *функцией перехода*, где L — сдвиг влево, R — сдвиг вправо. Если δ не определено для текущего состояния и текущего символа ленты, то машина останавливается; ^[19] интуитивно, функция перехода определяет следующее состояние, к которому переходит текущее состояние, какой символ следует перезаписать вместо текущего символа, на который указывает головка, и следующее движение головки.

Вариант допускает «без сдвига», скажем, N, как третий элемент набора направлений $\{L, R\}$.

Семикортежник для занятого бобра с тремя состояниями выглядит следующим образом (подробнее об этом занятом бобре см. в разделе Примеры машин Тьюринга):

- $Q = \{A, B, C, \text{HALT}\}$ (государства);
- $\Gamma = \{0, 1\}$ (символы алфавита на ленте);
- $b = 0$ (пустой символ);
- $\Sigma = \{1\}$ (входные символы);
- $q_0 = A$ (исходное состояние);
- $F = \{\text{HALT}\}$ (конечные состояния);
- δ = см. таблицу состояний ниже (функция перехода).

Изначально все ячейки ленты помечены знаком 0.



Здесь внутреннее состояние (q_1) показано внутри головки, а иллюстрация описывает ленту как бесконечную и предварительно заполненную "0", символом, служащим пробелом. Полное состояние системы (ее "полная конфигурация") состоит из внутреннего состояния, любых непустых символов на ленте (на этой иллюстрации "11B") и положения головки относительно этих символов, включая пробелы, т. е. "011B". (Рисунок по Мински (1967) стр. 121.)



Таблица состояний для 3-х позиционного, 2-х символьного занятого бобра

Символ ленты	Текущее состояние А			Текущее состояние В			Текущее состояние С		
	Написать символ	Переместить ленту	Следующий штат	Написать символ	Переместить ленту	Следующий штат	Написать символ	Переместить ленту	Следующий штат
0	1	Р	Б	1	Л	А	1	Л	Б
1	1	Л	С	1	Р	Б	1	Р	ОСТАНОВИТЬ

Дополнительные сведения, необходимые для визуализации или реализации машин Тьюринга

Например,

- Потребуется принять множество решений относительно того, как на самом деле будут выглядеть символы, а также разработать надежный способ чтения и записи символов на неопределенный срок.
- Операции сдвига влево и вправо могут смещать головку ленты по ленте, но при фактическом построении машины Тьюринга практичнее заставить ленту скользить вперед и назад под головкой.
- Лента может быть конечной и автоматически расширяться пробелами по мере необходимости (что ближе всего к математическому определению), но более распространено думать о ней как о бесконечно растягивающейся на одном или обоих концах и предварительно заполненной пробелами, за исключением явно заданного конечного фрагмента, на котором находится головка ленты (это, конечно, нереализуемо на практике). Лента *не может* быть фиксированной по длине, поскольку это не соответствовало бы данному определению и серьезно ограничило бы диапазон вычислений, которые может выполнять машина, до вычислений линейного

ограниченного автомата, если бы лента была пропорциональна размеру входных данных, или конечного автомата, если бы она была строго фиксированной длины.

Альтернативные определения

Определения в литературе иногда немного различаются, чтобы сделать аргументы или доказательства проще или понятнее, но это всегда делается таким образом, чтобы полученная машина имела ту же вычислительную мощность. Например, набор может быть изменен с $\{L, R\}$ на $\{L, R, N\}$, где N ("None" или "No-operation") позволит машине оставаться на той же ячейке ленты вместо того, чтобы двигаться влево или вправо. Это не увеличит вычислительную мощность машины.

Наиболее распространенное соглашение представляет каждую «инструкцию Тьюринга» в «таблице Тьюринга» одним из девяти кортежей по 5, согласно соглашению Тьюринга/Дэвиса (Тьюринг (1936) в «*Неразрешимом*», стр. 126–127 и Дэвис (2000) стр. 152):

(определение 1): $(q_i, S_j, Sk, E/N, L/R/N, q_m)$
 (текущее состояние q_i , отсканированный символ S_j , печать символа Sk , стирание E / нет N ,
 перемещение_ленты_на_один_квадрат влево L / вправо R / нет N , новое состояние q_m)

Другие авторы (Минский (1967) стр. 119, Хопкрофт и Ульман (1979) стр. 158, Стоун (1972) стр. 9) принимают иное соглашение, при котором новое состояние q_m указывается сразу после сканированного символа S_j :

(определение 2): $(q_i, S_j, q_m, Sk, E/N, L/R/N)$
 (текущее состояние q_i , отсканированный символ S_j , новое состояние q_m , печать символа Sk / стереть
 E / нет N , переместить_ленту_на_один_квадрат влево L / вправо R / нет N)

В оставшейся части статьи будет использоваться «определение 1» (конвенция Тьюринга/Дэвиса).

Пример: таблица состояний для 3-х состояний 2-символьного занятого бобра сокращена до 5-ти кортежей

Текущее состояние	Отсканированный символ	Символ печати	Переместить ленту	Конечное (т.е. следующее) состояние	5-кортежи
A	0	1	Р	Б	(A , 0, 1, Р, Б)
A	1	1	Л	С	(A , 1, 1, Л, С)
Б	0	1	Л	A	(Б , 0, 1, Л, A)
Б	1	1	Р	Б	(Б , 1, 1, Р, Б)
С	0	1	Л	Б	(С , 0, 1, Л, Б)
С	1	1	Н	ЧАС	(С , 1, 1, Н, ЧАС)

В следующей таблице исходная модель Тьюринга допускала только первые три строки, которые он называл N_1 , N_2 , N_3 (ср. Тьюринг в *The Undecidable*, стр. 126). Он допускал стирание «сканируемого квадрата», называя нулевой символ $S_0 =$ «стирание» или «пустой» и т. д. Однако он не допускал непечатаемость, поэтому каждая строка инструкции включает «печатный символ S_k » или «стирание» (ср. сноску 12 в Post (1947), *The Undecidable*, стр. 300). Сокращения принадлежат Тьюрингу (*The Undecidable*, стр. 119). После оригинальной статьи Тьюринга в 1936–1937 годах машинные модели допускали все девять возможных типов пятикортежей:

	Текущая m-конфигурация (состояние Тьюринга)	Символ ленты	Печать-операция	Лента-движение	Конечная m-конфигурация (состояние Тьюринга)	5-кортеж	5-кортеж комментариев	4-кортеж
H1	q_j	$C_{дж}$	Печать(S_k)	Левый L	$кв^M$	(q_i, S_j, Sk, L, q_m)	«Пусто» = S_0 , $1=S_1$ и т.д.	
H2	q_j	$C_{дж}$	Печать(S_k)	Правый R	$кв^M$	(q_i, S_j, Sk, R, q_m)	«Пусто» = S_0 , $1=S_1$ и т.д.	
H3	q_j	$C_{дж}$	Печать(S_k)	Ни один N	$кв^M$	(q_i, S_j, Sk, N, q_m)	«Пусто» = S_0 , $1=S_1$ и т.д.	(q_i, S_j, Sk, q_m)
4	q_j	$C_{дж}$	Ни один N	Левый L	$кв^M$	(q_i, S_j, N, L, q_m)		(q_i, S_j, L, q_m)
5	q_j	$C_{дж}$	Ни один N	Правый R	$кв^M$	(q_i, S_j, N, R, q_m)		(q_i, S_j, R, q_m)
6	q_j	$C_{дж}$	Ни один N	Ни один N	$кв^M$	(q_i, S_j, N, N, q_m)	Прямой «прыжок»	(q_i, S_j, N, q_m)
7	q_j	$C_{дж}$	Стереть	Левый L	$кв^M$	(q_i, S_j, E, L, q_m)		
8	q_j	$C_{дж}$	Стереть	Правый R	$кв^M$	(q_i, S_j, E, R, q_m)		
9	q_j	$C_{дж}$	Стереть	Ни один N	$кв^M$	(q_i, S_j, E, N, q_m)		(q_i, S_j, E, q_m)

Любая таблица Тьюринга (список инструкций) может быть составлена из девяти вышеуказанных 5-кортежей. По техническим причинам три непечатаемые или "N" инструкции (4, 5, 6) обычно можно опустить. Примеры см. в разделе [Примеры машин Тьюринга](#).

Реже встречается использование 4-кортежей: они представляют собой дальнейшее раздробление инструкций Тьюринга (ср. Post (1947), Boolos & Jeffrey (1974, 1999), Davis-Sigal-Weyuker (1994)); также см. больше на [Post–Turing machine](#).

«Государство»

Слово «состояние», используемое в контексте машин Тьюринга, может быть источником путаницы, поскольку оно может означать две вещи. Большинство комментаторов после Тьюринга использовали «состояние» для обозначения имени/обозначения текущей инструкции, которая должна быть выполнена, т. е. содержимого регистра состояния. Но Тьюринг (1936) провел четкое различие между записью того, что он назвал «m-конфигурацией» машины, и «состоянием прогресса» машины (или человека) через вычисление — текущим состоянием всей системы. То, что Тьюринг называл «формулой состояния», включает как текущую инструкцию, так и *все* символы на ленте:

Таким образом, состояние хода вычислений на любой стадии полностью определяется записью инструкций и символами на ленте. То есть, *состояние системы* может быть описано одним выражением (последовательностью символов), состоящим из символов на ленте, за которыми следует Δ (которая, как предполагается, не должна появляться в другом месте), а затем записью инструкций. Это выражение называется «формулой состояния».

— *Неразрешимое*, стр. 139–140, выделено автором

Ранее в своей статье Тьюринг пошел еще дальше: он приводит пример, в котором он поместил символ текущей «m-конфигурации» — метку инструкции — под сканируемым квадратом вместе со всеми символами на ленте (*The Undecidable*, стр. 121); это он называет «*полной конфигурацией*» (*The Undecidable*, стр. 118). Чтобы напечатать «полную конфигурацию» на одной строке, он помещает метку состояния/m-конфигурацию слева от сканируемого символа.

Вариант этого можно увидеть у Клини (1952), где Клини показывает, как записать число Гёделя «ситуации» машины: он помещает символ «m-конфигурации» q_4 над сканированным квадратом примерно в центре 6 непустых квадратов на ленте (см. рисунок ленты Тьюринга в этой статье) и помещает его справа от сканированного квадрата. Но Клини называет само « q_4 » «состоянием машины» (Клини, стр. 374–375). Хопкрофт и Ульман называют эту композицию «мгновенным описанием» и следуют соглашению Тьюринга о размещении «текущего состояния» (метка инструкции, m-конфигурация) слева от сканируемого символа (стр. 149), то есть мгновенное описание представляет собой композицию непустых символов слева, состояния машины, текущего символа, сканируемого головкой, и непустых символов справа.

Пример: общее состояние 3-х состояний 2-х символьного занятого бобра после 3-х «ходов» (взято из примера «бег» на рисунке ниже):

1 A 1

Это означает: после трех ходов на ленте ... 000110000 ..., головка сканирует самую правую 1, и состояние равно A. Пробелы (в данном случае представленные «0») могут быть частью общего состояния, как показано здесь: B 01; на ленте есть одна 1, но головка сканирует 0 («пробел») слева от нее, и состояние равно B.

«Состояние» в контексте машин Тьюринга следует уточнить, что именно описывается: текущая инструкция, или список символов на ленте вместе с текущей инструкцией, или список символов на ленте вместе с текущей инструкцией, размещенной слева от сканируемого символа или справа от сканируемого символа.

Биограф Тьюринга Эндрю Ходжес (1983: 107) отметил и проанализировал эту путаницу.

Диаграммы «состояний»

Таблица для 3-х состояний занятого бобра («P» = вывести/записать «1»)

Символ ленты	Текущее состояние A			Текущее состояние B			Текущее состояние C		
	Написать символ	Переместить ленту	Следующий штат	Написать символ	Переместить ленту	Следующий штат	Написать символ	Переместить ленту	Следующий штат
0	П	Р	Б	П	Л	А	П	Л	Б
1	П	Л	С	П	Р	Б	П	Р	ОСТАНОВИТЬ

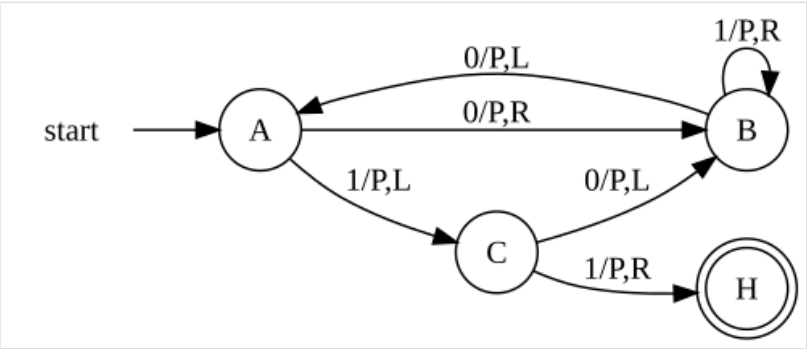
Справа: приведенная выше таблица, представленная в виде диаграммы «переходов состояний».

Обычно большие таблицы лучше оставить таблицами (Booth, стр. 74). Их легче моделировать на компьютере в табличной форме (Booth, стр. 74). Однако некоторые концепции — например, машины с «сбросом» состояний и машины с повторяющимися шаблонами (ср. Hill and Peterson, стр. 244ff) — легче увидеть, если рассматривать их как рисунок.

Читатель должен сам решать, представляет ли собой рисунок улучшение таблицы, исходя из конкретного контекста.

Читателю следует снова предупредить, что такие диаграммы представляют собой моментальный снимок их таблицы, замороженной во времени, а не ход («траекторию») вычисления во времени и пространстве. В то время как каждый раз, когда занятая машина-бобёр «работает», она всегда будет следовать одной и той же траектории состояния, это не относится к «копирующей» машине, которой можно предоставить переменные входные «параметры».

Диаграмма «прогресс вычисления» показывает прогресс «состояния» (инструкции) трехступенчатого занятого бобра через его вычисление от начала до конца. Справа находится «полная конфигурация» Тьюринга («ситуация» Клини, «мгновенное описание» Хопкрофта–Ульмана) на каждом шаге. Если бы машину остановили и очистили, чтобы очистить как «регистр



Машина Тьюринга «3-state busy beaver» в представлении с конечным числом состояний . Каждый круг представляет «состояние» таблицы — «m-конфигурацию» или «инструкцию». «Направление» перехода состояний показано стрелкой. Метка (например, 0/P,R) рядом с исходящим состоянием (в «хвосте» стрелки) указывает сканируемый символ, который вызывает определенный переход (например, 0), за которым следует косая черта / , за которой следуют последующие «поведения» машины, например, « P print », затем переместить ленту « R right ». Общепринятого формата не существует. Показанное соглашение соответствует McClusky (1965), Booth (1967), Hill и Peterson (1974).

состояния», так и всю ленту, эти «конфигурации» можно было бы использовать для возобновления вычисления в любом месте его хода (ср. Turing (1936) *The Undecidable* , стр. 139–140).

Эквивалентные модели

Многие машины, которые, как можно было бы подумать, обладают большей вычислительной мощностью, чем простая универсальная машина Тьюринга, могут оказаться не более мощными (Хопкрофт и Ульман, стр. 159, ср. Мински (1967)). Они могут вычислять быстрее, возможно, или использовать меньше памяти, или их набор инструкций может быть меньше, но они не могут вычислять более мощно (т. е. больше математических функций). (Тезис Чёрча-Тьюринга *предполагает*, что это верно для любого типа машины: что всё, что может быть «вычислено», может быть вычислено некоторой машиной Тьюринга.)

Машина Тьюринга эквивалентна автомату с одним стеком (PDA), который стал более гибким и лаконичным за счет ослабления требования LIFO (last-in-first-out) к его стеку. Кроме того, машина Тьюринга также эквивалентна PDA с двумя стеками со стандартной семантикой LIFO, используя один стек для моделирования ленты слева от головки, а другой стек — для ленты справа.

С другой стороны, некоторые очень простые модели оказываются эквивалентными Тьюрингу , то есть обладают той же вычислительной мощностью, что и модель машины Тьюринга.

Распространенными эквивалентными моделями являются многоленточная машина Тьюринга , многодорожечная машина Тьюринга , машины с входом и выходом и недетерминированная машина Тьюринга (NDTM) в отличие от детерминированной машины Тьюринга (DTM), для которой таблица действий имеет не более одной записи для каждой комбинации символа и состояния.

Машины Тьюринга, предназначенные только для чтения и движущиеся вправо, эквивалентны DFA (а также NFA путем преобразования с использованием алгоритма преобразования NFA в DFA).

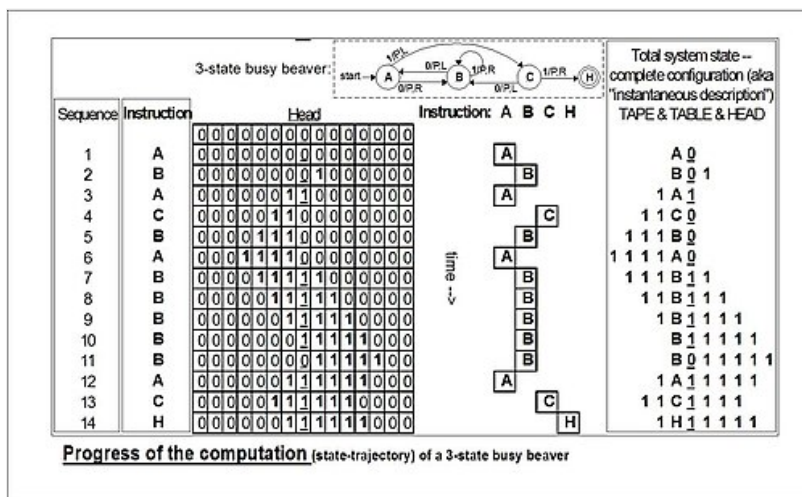
В практических и дидактических целях эквивалентную регистровую машину можно использовать как обычный язык программирования ассемблера .

Актуальным вопросом является то, является ли модель вычислений, представленная конкретными языками программирования, эквивалентной Тьюрингу. В то время как вычисления реального компьютера основаны на конечных состояниях и, таким образом, не способны имитировать машину Тьюринга, сами языки программирования не обязательно имеют это ограничение. Кирнер и др., 2009 показали, что среди языков программирования общего назначения некоторые являются полными по Тьюрингу, а другие — нет. Например, ANSI C не является полным по Тьюрингу, поскольку все экземпляры ANSI C (возможны различные экземпляры, поскольку стандарт намеренно оставляет определенное поведение неопределенным по причинам наследия) подразумевают память с конечным пространством. Это связано с тем, что размер типов данных ссылок на память, называемых указателями , доступен внутри языка. Однако другие языки программирования, такие как Pascal , не имеют этой функции, которая позволяет им быть полными по Тьюрингу в принципе. Он просто полон по Тьюрингу в принципе, поскольку выделение памяти в языке программирования может быть неудачным, что означает, что язык программирования может быть полным по Тьюрингу при игнорировании неудачных выделений памяти, но скомпилированные программы, исполняемые на реальном компьютере, не могут.

Выбор с-машин, оракул о-машин

В начале своей статьи (1936) Тьюринг проводит различие между «автоматической машиной» — ее «движением... полностью определяемым конфигурацией» — и «машиной выбора»:

...движение которого лишь частично определяется конфигурацией... Когда такая машина достигает одной из этих неоднозначных конфигураций, она не может продолжать работу, пока внешний оператор не сделает произвольный выбор. Это было бы так, если бы мы использовали машины для работы с аксиоматическими системами.



Эволюция вычислений трудолюбивого бобра начинается сверху и продолжается снизу.

Тьюринг (1936) не развивает эту тему более подробно, за исключением сноски, в которой он описывает, как использовать а-машину для «нахождения всех доказуемых формул исчисления [Гильберта]», а не использовать машину выбора. Он «предполагает, что выбор всегда находится между двумя возможностями 0 и 1. Каждое доказательство будет тогда определяться последовательностью выборов i_1, i_2, \dots, i_n ($i_1 = 0$ или $1, i_2 = 0$ или $1, \dots, i_n = 0$ или 1), и, следовательно, число $2^{i_1} + 2^{i_2} + \dots + 2^{i_n}$ полностью определяет доказательство. Автоматическая машина последовательно выполняет доказательство 1, доказательство 2, доказательство 3, ...» (сноска ‡, *The Undecidable* , стр. 138)

Это действительно метод, с помощью которого детерминированная (т. е. а-) машина Тьюринга может быть использована для имитации действия недетерминированной машины Тьюринга ; Тьюринг решил этот вопрос в сноске и, по-видимому, исключил его из дальнейшего рассмотрения.

Машина -оракул или о-машина — это а-машина Тьюринга, которая приостанавливает свои вычисления в состоянии «**o**», пока для завершения своих вычислений она «ждет решения» «оракула» — сущности, не указанной Тьюрингом, «за исключением того, что она не может быть машиной» (Тьюринг (1939), *Неразрешимое* , стр. 166–168).

Универсальные машины Тьюринга

Как писал Тьюринг в «*Неразрешимом*» , стр. 128 (курсив добавлен):

Можно изобрести одну машину , которую можно использовать для вычисления *любой* вычислимой последовательности. Если эта машина **U** снабжена лентой, в начале которой записана строка из пятёрок, разделённых точками с запятой некоторой вычислительной машины **M** , то **U** вычислит ту же последовательность, что и **M**.

Это открытие сейчас воспринимается как должное, но в то время (1936) оно считалось поразительным. Модель вычислений, которую Тьюринг назвал своей «универсальной машиной» — «**U**» для краткости — некоторые считают (ср. Дэвис (2000)) фундаментальным теоретическим прорывом, который привел к идее компьютера с хранимой программой .

Статья Тьюринга... по сути содержит изобретение современного компьютера и некоторых сопровождавших его методов программирования.

— Минский (1967), стр. 104

С точки зрения вычислительной сложности , многоленточная универсальная машина Тьюринга должна быть медленнее только на логарифмический множитель по сравнению с машинами, которые она моделирует. Этот результат был получен в 1966 году FC Hennie и RE Stearns . (Arora и Barak, 2009, теорема 1.9)

Сравнение с реальными машинами

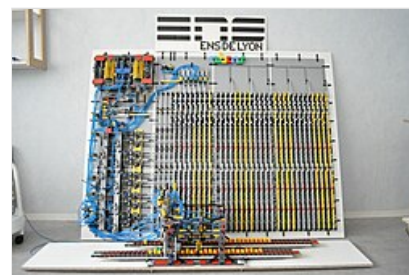
Машины Тьюринга более мощные, чем некоторые другие виды автоматов, такие как конечные автоматы и автоматы с магазинной памятью . Согласно тезису Чёрча-Тьюринга , они столь же мощные, как и настоящие машины, и способны выполнять любые операции, которые может выполнять настоящая программа. В этом утверждении упускается из виду, что, поскольку реальная машина может иметь только конечное число конфигураций , она является не чем иным, как конечным автоматом, тогда как машина Тьюринга имеет неограниченный объем памяти, доступной для ее вычислений.

Есть несколько способов объяснить, почему машины Тьюринга являются полезными моделями реальных компьютеров:

- Все, что может вычислить реальный компьютер, может вычислить и машина Тьюринга. Например: «Машина Тьюринга может имитировать любой тип подпрограммы, встречающийся в языках программирования, включая рекурсивные процедуры и любые известные механизмы передачи параметров» (Хопкрофт и Ульман, стр. 157).



Реализация машины Тьюринга



Реализация машины Тьюринга с использованием деталей Lego

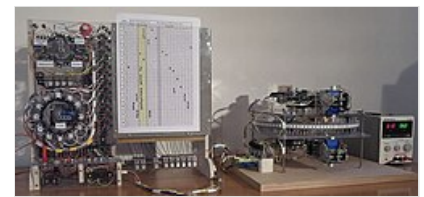
Достаточно большой FSA также может моделировать любой реальный компьютер, игнорируя ИО. Таким образом, утверждение об ограничениях машин Тьюринга будет применимо и к реальным компьютерам.

- Разница заключается только в способности машины Тьюринга манипулировать неограниченным количеством данных. Однако, учитывая конечное количество времени, машина Тьюринга (как и настоящая машина) может манипулировать только конечным количеством данных.
- Как и машина Тьюринга, реальная машина может по мере необходимости увеличивать объем памяти, приобретая больше дисков или других носителей информации.
- Описания реальных машинных программ с использованием более простых абстрактных моделей часто намного сложнее описаний с использованием машин Тьюринга. Например, машина Тьюринга, описывающая алгоритм, может иметь несколько сотен состояний, в то время как эквивалентный детерминированный конечный автомат (DFA) на данной реальной машине имеет квадриллионы. Это делает представление DFA невозможным для анализа.
- Машины Тьюринга описывают алгоритмы независимо от того, сколько памяти они используют. Существует предел памяти, которой обладает любая современная машина, но этот предел может произвольно увеличиваться со временем. Машины Тьюринга позволяют нам делать утверждения об алгоритмах, которые (теоретически) будут сохраняться вечно, независимо от достижений в архитектуре *обычных вычислительных машин*.
- Алгоритмы, работающие на абстрактных машинах, эквивалентных Тьюрингу, могут иметь доступные типы данных произвольной точности и никогда не сталкиваются с непредвиденными условиями (включая, помимо прочего, нехватку памяти).

Ограничения

Теория сложности вычислений

Ограничением машин Тьюринга является то, что они не моделируют сильные стороны конкретной компоновки. Например, современные компьютеры с хранимой программой на самом деле являются примерами более конкретной формы абстрактной машины, известной как машина с произвольной выборкой или модель машины RASP. Как и универсальная машина Тьюринга, RASP хранит свою «программу» в «памяти», внешней по отношению к «инструкциям» ее конечной машины. В отличие от универсальной машины Тьюринга, RASP имеет бесконечное количество различных, пронумерованных, но неограниченных «регистров» — «ячеек» памяти, которые могут содержать любое целое число (ср. Elgot and Robinson (1964), Hartmanis (1971) и, в частности, Cook-Rechow (1973); ссылки на машину с произвольной выборкой). Конечная машина RASP оснащена возможностью косвенной адресации (например, содержимое одного регистра может использоваться в качестве адреса для указания другого регистра); Таким образом, «программа» RASP может адресовать любой регистр в последовательности регистров. Результатом этого различия является то, что существуют вычислительные оптимизации, которые могут быть выполнены на основе индексов памяти, что невозможно в обычной машине Тьюринга; таким образом, когда машины Тьюринга используются в качестве основы для ограничения времени выполнения, может быть доказана «ложная нижняя граница» для времени выполнения определенных алгоритмов (из-за ложного упрощающего предположения машины Тьюринга). Примером этого является двоичный поиск , алгоритм, который, как можно показать, выполняется быстрее при использовании модели вычислений RASP, а не модели машины Тьюринга.



Еще одна реализация машины Тьюринга

Взаимодействие

На заре вычислений использование компьютеров обычно ограничивалось пакетной обработкой , т. е. неинтерактивными задачами, каждая из которых производила выходные данные из заданных входных данных. Теория вычислимости, которая изучает вычислимость функций от входов до выходов, и для которой были изобретены машины Тьюринга, отражает эту практику.

С 1970-х годов интерактивное использование компьютеров стало гораздо более распространенным. В принципе, это можно смоделировать, имея внешнего агента, считывающего с ленты и записывающего на нее одновременно с машиной Тьюринга, но это редко соответствует тому, как на самом деле происходит взаимодействие; поэтому при описании интерактивности обычно предпочитают альтернативы, такие как автоматы ввода-вывода .

Сравнение с арифметической моделью вычисления

Арифметическая модель вычислений отличается от модели Тьюринга в двух аспектах: [20] : 32

- В арифметической модели каждое действительное число требует одной ячейки памяти, тогда как в модели Тьюринга размер хранения действительного числа зависит от количества бит, необходимых для его представления.
- В арифметической модели каждая базовая арифметическая операция с действительными числами (сложение, вычитание, умножение и деление) может быть выполнена за один шаг, тогда как в модели Тьюринга время

выполнения каждой арифметической операции зависит от длины операндов.

Некоторые алгоритмы работают за полиномиальное время в одной модели, но не в другой. Например:

- Алгоритм Евклида выполняется за полиномиальное время в модели Тьюринга, но не в арифметической модели.
- Алгоритм, который считывает n чисел, а затем вычисляет 2^{2^n} путем повторного возведения в квадрат за полиномиальное время в арифметической модели, но не в модели Тьюринга. Это происходит потому, что количество бит, требуемых для представления результата, экспоненциально зависит от размера входных данных.

Однако, если алгоритм работает за полиномиальное время в арифметической модели, и, кроме того, двоичная длина всех задействованных чисел является полиномиальной по длине входных данных, то он всегда является полиномиальным по времени в модели Тьюринга. Говорят, что такой алгоритм работает за строго полиномиальное время.

История

Исторический обзор: вычислительная техника

Робин Ганди (1919–1995) — ученик Алана Тьюринга (1912–1954) и его друг на всю жизнь — прослеживает происхождение понятия «вычислительная машина» до Чарльза Бэббиджа (около 1834 года) и фактически выдвигает «Тезис Бэббиджа»:

Что все разработки и операции анализа теперь могут быть выполнены машинами .

— (курсив в Бэббидже, цитируемом Ганди, стр. 54)

Анализ Ганди аналитической машины Бэббиджа описывает следующие пять операций (ср. стр. 52–53):

1. Арифметические функции $+$, $-$, \times , где $-$ обозначает «правильное» вычитание: $x - y = 0$, если $y \geq x$.
2. Любая последовательность операций является операцией.
3. Итерация операции (повторение n раз операции P).
4. Условная итерация (повторение n раз операции P при условии «успешности» теста T).
5. Условный перенос (т.е. условный «goto»).

Ганди утверждает, что «функции, которые могут быть вычислены с помощью (1), (2) и (4), являются именно теми, которые являются вычислимыми по Тьюрингу». (стр. 53). Он цитирует другие предложения относительно «универсальных вычислительных машин», включая предложения Перси Ладгейта (1909), Леонардо Торреса Кеведо (1914), [21] [22] Мориса д'Оканя (1922), Луи Куффиньяля (1933), Ванневары Буша (1936), Говарда Эйкена (1937). Однако:

... акцент делается на программировании фиксированной итерируемой последовательности арифметических операций. Фундаментальная важность условной итерации и условного переноса для общей теории вычислительных машин не осознается...

— Ганди стр. 55

Entscheidungsproblem («проблема принятия решения»): десятый вопрос Гильберта 1900 года

Что касается проблем Гильберта, поставленных известным математиком Дэвидом Гильбертом в 1900 году, аспект проблемы № 10 витал в воздухе почти 30 лет, прежде чем был сформулирован точно. Первоначальное выражение Гильберта для № 10 выглядит следующим образом:

10. Определение разрешимости диофантова уравнения . Дано диофантово уравнение с любым числом неизвестных величин и с рациональными целыми коэффициентами: Разработать процесс, согласно которому можно определить за конечное число операций, разрешимо ли уравнение в рациональных целых числах. Entscheidungsproblem [задача принятия решений для логики первого порядка] решена, когда мы знаем процедуру, которая позволяет для любого заданного логического выражения решить за конечное число операций его действительность или выполнимость... Entscheidungsproblem следует считать основной проблемой математической логики.

— цитируется с этим переводом и оригинальным немецким текстом в Dershowitz and Gurevich, 2008

К 1922 году это понятие «Entscheidungsproblem» несколько развилось, и Х. Бехманн заявил, что

... наиболее общая форма Entscheidungsproblem [является] следующей:

Требуется совершенно определенное общеприменимое предписание, которое позволит за конечное число шагов решить истинность или ложность данного чисто логического утверждения...

— Ганди, стр. 57, цитируя Бехмана

Бехман замечает, что... общая проблема эквивалентна проблеме определения того, какие математические предложения истинны.

— *там же*.

Если бы удалось решить Entscheidungsproblem, то появилась бы «процедура решения многих (или даже всех) математических задач».

— *там же* , стр. 92

К международному конгрессу математиков 1928 года Гильберт «сформулировал свои вопросы достаточно точно. Во-первых, была ли математика полной ... Во-вторых, была ли математика последовательной ... И, в-третьих, была ли математика разрешимой ?» (Ходжес, стр. 91, Хокинг, стр. 1121). На первые два вопроса ответил в 1930 году Курт Гёдель на том же самом собрании, где Гильберт произнес свою речь об уходе на пенсию (к большому огорчению Гильберта); третий — Entscheidungsproblem — пришлось ждать до середины 1930-х годов.

Проблема заключалась в том, что ответ сначала требовал точного определения «определенного общего применимого предписания», которое профессор Принстона Алонзо Чёрч назовёт «эффективной вычислимостью», а в 1928 году такого определения не существовало. Но в течение следующих 6–7 лет Эмиль Пост разработал своё определение рабочего, перемещающегося из комнаты в комнату, пишущего и стирающего отметки по списку инструкций (Post 1936), как и Чёрч и его два студента Стивен Клини и Дж. Б. Россер , используя лямбда-исчисление Чёрча и теорию рекурсии Гёделя (1934). Статья Чёрча (опубликованная 15 апреля 1936 года) показала, что Entscheidungsproblem действительно «неразрешима» ^[23] и опередила Тьюринга почти на год (статья Тьюринга была представлена 28 мая 1936 года, опубликована в январе 1937 года). Тем временем Эмиль Пост представил краткую статью осенью 1936 года, так что Тьюринг, по крайней мере, имел приоритет над Постом. Пока Чёрч рецензировал статью Тьюринга, Тьюринг успел изучить статью Чёрча и добавить Приложение, в котором он набросал доказательство того, что лямбда-исчисление Чёрча и его машины будут вычислять одни и те же функции.

Но то, что сделал Черч, было чем-то совершенно иным и в определенном смысле более слабым. ... конструкция Тьюринга была более прямой и давала аргумент, исходящий из первых принципов, закрывая пробел в доказательствах Черча.

— Ходжес, стр. 112

А Пост лишь предложил определение исчисляемости и раскритиковал «определение» Черча, но ничего не доказал.

Машина Алана Тьюринга

Весной 1935 года Тьюринг, будучи молодым студентом магистратуры в Королевском колледже в Кембридже , принял вызов; его вдохновили лекции логика М. Х. А. Ньюмена «и он узнал из них о работах Гёделя и Entscheidungsproblem... Ньюмен использовал слово «механический»... В некрологе Тьюринга 1955 года Ньюмен пишет:

На вопрос «что такое «механический» процесс?» Тьюринг дал характерный ответ: «Нечто, что может быть сделано машиной», и приступил к весьма близкой ему по духу задаче анализа общего понятия вычислительной машины.

— Ганди, стр. 74

Ганди утверждает, что:

Я предполагаю, но не знаю, что Тьюринг с самого начала своей работы ставил своей целью доказательство неразрешимости Entscheidungsproblem. Он сказал мне, что «главная идея» статьи пришла к нему, когда он лежал на лугах Гранчестера летом 1935 года. «Главной идеей» мог быть либо его анализ вычислений, либо его понимание того, что существует универсальная машина, и, следовательно, диагональный аргумент для доказательства неразрешимости.

— *там же* , стр. 76

Хотя Ганди считал, что приведенное выше утверждение Ньюмана «вводит в заблуждение», это мнение разделяют не все. Тьюринг всю жизнь интересовался машинами: «Алан мечтал изобрести пишущие машинки, когда был мальчиком; у [его матери] миссис Тьюринг была пишущая машинка; и он вполне мог начать с того, чтобы спросить себя, что имелось в виду, когда называл пишущую машинку «механической»» (Ходжес, стр. 96). Во время работы в Принстоне, работая над докторской диссертацией, Тьюринг построил булев-логический множитель (см. ниже). Его докторская диссертация под названием «Системы логики, основанные на ординалах», содержит следующее определение «вычислимой функции»:

Выше было сказано, что «функция эффективно вычислима, если ее значения могут быть найдены некоторым чисто механическим процессом». Мы можем понимать это утверждение буквально, понимая под чисто механическим процессом тот, который может быть выполнен машиной. Можно дать математическое описание в определенной нормальной форме структур этих машин. Развитие этих идей приводит к авторскому определению вычислимой функции и к отождествлению вычислимости с эффективной вычислимостью. Несложно, хотя и несколько трудоемко, доказать, что эти три определения [третье — λ -исчисление] эквивалентны.

— Тьюринг (1939) в «*Неразрешимом*», стр. 160

Алан Тьюринг изобрел «а-машину» (автоматическую машину) в 1936 году.^[7] Тьюринг представил свою статью 31 мая 1936 года в Лондонское математическое общество для его *трудов* (ср. Hodges 1983:112), но она была опубликована в начале 1937 года, а оттиски стали доступны в феврале 1937 года (ср. Hodges 1983:129). Именно научный руководитель Тьюринга, Алонзо Чёрч, позже ввёл термин «машина Тьюринга» в обзор.^[10] С помощью этой модели Тьюринг смог ответить на два вопроса отрицательно:

- Существует ли машина, способная определить, является ли любая произвольная машина на ее ленте «циклической» (например, зависит или не может продолжить выполнение вычислительной задачи)?
- Существует ли машина, способная определить, печатает ли любая произвольная машина на своей ленте заданный символ?^[11]^[12]

Таким образом, предоставив математическое описание очень простого устройства, способного производить произвольные вычисления, он смог доказать свойства вычислений в целом и, в частности, невыводимость Entscheidungsproblem («проблемы принятия решений»)^[13].

Когда Тьюринг вернулся в Великобританию, он в конечном итоге стал соавтором взлома немецких секретных кодов, созданных шифровальными машинами под названием «Энигма»; он также принял участие в разработке ACE (автоматической вычислительной машины), «[Тьюринг] предложение ACE было фактически самодостаточным, и его корни лежали не в EDVAC [инициативе США], а в его собственной универсальной машине» (Ходжес, стр. 318). Все еще продолжаются споры относительно происхождения и природы того, что было названо Клини (1952) «тезисом Тьюринга». Но то, что Тьюринг доказал с помощью своей модели вычислительной машины, отражено в его статье «О вычисляемых числах с приложением к проблеме Entscheidungsproblem» (1937):

[что] проблема Гильберта Entscheidungsproblem не может иметь решения... Поэтому я предлагаю показать, что не может быть общего процесса для определения того, доказуема ли данная формула U функционального исчисления K , т. е. что не может быть машины, которая, имея любую одну из этих формул U , в конечном итоге скажет, доказуема ли U .

— из статьи Тьюринга, перепечатанной в *The Undecidable*, стр. 145

Пример Тьюринга (его второе доказательство): Если кто-то попросит общую процедуру, которая скажет нам: «Печатает ли эта машина когда-нибудь 0», то вопрос «неразрешим».

1937–1970: «Цифровой компьютер», рождение «компьютерной науки»

В 1937 году, работая в Принстоне над своей докторской диссертацией, Тьюринг построил цифровой (булево-логический) умножитель с нуля, создав собственные электромеханические реле (Ходжес, стр. 138). «Задачей Алана было воплотить логическую конструкцию машины Тьюринга в сети переключателей, управляемых реле...» (Ходжес, стр. 138). Хотя Тьюринг, возможно, изначально был просто любопытным и экспериментировал, вполне серьезная работа в том же направлении велась в Германии (Конрад Цузе (1938)), а также в Соединенных Штатах (Говард Эйкен) и Джордж Штибиц (1937); плоды их трудов использовались как армиями стран Оси, так и союзниками во Второй мировой войне (ср. Ходжес, стр. 298–299). В начале-середине 1950-х годов Хао Ван и Марвин Мински свели машину Тьюринга к более простой форме (предшественнику пост-Тьюринговой машины Мартина Дэвиса); одновременно европейские исследователи сводили новомодный электронный компьютер к похожему на компьютер теоретическому объекту, эквивалентному тому, что теперь называлось «машиной Тьюринга». В конце 1950-х и начале 1960-х годов параллельно идущие разработки Мелзака и Ламбека (1961), Мински (1961) и Шепердсона и Стерджиса (1961) продвинули европейскую работу дальше и свели машину Тьюринга к более удобной, похожей на компьютер абстрактной модели, называемой счетной машиной ; Элгот и Робинсон (1964), Хартманис (1971), Кук и Рекхов

(1973) продвинули эту работу еще дальше с моделями регистровой машины и машины с произвольным доступом — но в основном все они представляют собой просто многоленточные машины Тьюринга с набором инструкций, подобным арифметике.

1970—настоящее время: как модель вычислений

Сегодня счетчики, регистры и машины с произвольным доступом, а также их предок машина Тьюринга продолжают оставаться моделями выбора для теоретиков, исследующих вопросы теории вычислений. В частности, теория вычислительной сложности использует машину Тьюринга:

В зависимости от объектов, которыми мы хотим манипулировать в вычислениях (такие числа, как неотрицательные целые числа или буквенно-цифровые строки), в теории сложности машин доминирующее положение заняли две модели:

автономная многоленточная машина Тьюринга ..., представляющая собой стандартную модель для строковых вычислений, и *машина с произвольным доступом (RAM)*, представленная Куком и Рекховом..., которая моделирует идеализированный компьютер в стиле фон Неймана.

— ван Эмде Боас 1990:4

Лишь в смежной области анализа алгоритмов эту роль берет на себя модель RAM.

— ван Эмде Боас 1990:16

Смотрите также

- Арифметическая иерархия
- Граница Бекенштейна, показывающая невозможность машин Тьюринга с бесконечной лентой, конечным размером и ограниченной энергией
- Блуп и Флуп
- Константа Хайтина или Омега (компьютерная наука) для информации, касающейся проблемы остановки
- Рационализатор исчисления
- Китайская комната
- Игра Конвея «Жизнь», полный по Тьюрингу клеточный автомат
- Цифровая бесконечность
- Новый разум императора*
- Перечислитель (в теоретической информатике)
- Генетикс
- «Гёдель, Эшер, Бах: вечная золотая коса»* — известная книга, в которой, помимо прочего, обсуждается тезис Чёрча–Тьюринга.
- Проблема остановки, для получения дополнительных ссылок
- Гарвардская архитектура
- Императивное программирование
- Муравей Лэнгтона и Тьюрмиты, простые двумерные аналоги машины Тьюринга
- Список вещей, названных в честь Алана Тьюринга
- Модифицированная архитектура Гарварда
- Квантовая машина Тьюринга
- Клод Шеннон, еще один ведущий мыслитель в области теории информации
- Примеры машин Тьюринга
- Turing tarpit — любая вычислительная система или язык, которые, несмотря на полноту по Тьюрингу, обычно считаются бесполезными для практических вычислений.
- Неорганизованная машина, для самых ранних идей Тьюринга о нейронных сетях
- Архитектура фон Неймана

Примечания

- Минский 1967:107 «В своей статье 1936 года А. М. Тьюринг определил класс абстрактных машин, которые теперь носят его имя. Машина Тьюринга — это конечный автомат, связанный с особым видом среды — ее лентой, — в которой она может хранить (и позднее восстанавливать) последовательности символов», также Стоун 1972:8, где слово «машина» заключено в кавычки.
- Stone 1972:8 утверждает: «Эта «машина» является абстрактной математической моделью», также см. Sipser 2006:137ff, который описывает «модель машины Тьюринга». Rogers 1987 (1967):13 ссылается на «характеристику Тьюринга», Boolos Burgess and Jeffrey 2002:25 ссылается на «определенный вид идеализированной машины».
- Sipser 2006:137 «Машина Тьюринга может делать все, что может делать настоящий компьютер».
- Cf. Sipser 2002:137. Также, Rogers 1987 (1967):13 описывает «бумажную ленту бесконечной длины в обоих направлениях». Minsky 1967:118 утверждает: «Лента считается бесконечной в обоих направлениях». Boolos Burgess and Jeffrey 2002:25 включают возможность того, что «на каждом конце находится кто-то, кто добавляет дополнительные пустые квадраты по мере необходимости».

5. См. Rogers 1987 (1967): 13. Другие авторы используют слово «квадрат», например Boolos Burgess Jeffrey 2002: 35, Minsky 1967: 117, Penrose 1989: 37.
6. Boolos Burgess Jeffrey 2002:25 иллюстрирует машину, движущуюся по ленте. Penrose 1989:36-37 описывает себя как «чувствующего себя некомфортно» с бесконечной лентой, отмечая, что «ее может быть трудно сдвинуть!»; он «предпочитает думать о ленте как о представляющей некоторую внешнюю среду, через которую может двигаться наше конечное устройство», и после замечания, что «„движение“ — это удобный способ изображения вещей», а затем предполагает, что «устройство получает все свои входные данные из этой среды. Некоторые вариации модели машины Тьюринга также позволяют головке оставаться в том же положении, а не двигаться или останавливаться.
7. Ходжес, Эндрю(2012).Алан Тьюринг: Энигма(ред. The Centenary).Princeton University Press.ISBN 978-0-691-15564-7.
8. Идея пришла к нему в середине 1935 года (возможно, см. больше в разделе «История») после вопроса, заданного М. Х. А. Ньюманом в его лекциях: «Существовал ли определенный метод или, как выразился Ньюман, «механический процесс», который можно было бы применить к математическому утверждению и который дал бы ответ на вопрос, доказуемо ли оно» (Hodges 1983:93). Тьюринг представил свою статью 31 мая 1936 года в Лондонское математическое общество для его *трудов* (ср. Hodges 1983:112), но она была опубликована в начале 1937 года, а оттиски были доступны в феврале 1937 года (ср. Hodges 1983:129).
9. См. сноску в Davis 2000:151.
10. см. примечание в предисловии к «Собранию сочинений Алонзо Чёрча» (ред.Бердж, Тайлер; Эндертон, Герберт (2019-04-23)). «Собрание сочинений Алонзо Чёрча» (<https://mitpress.mit.edu/books/collected-works-alonzo-church>) . Кембридж, Массачусетс, США: MIT Press.ISBN (<https://mitpress.mit.edu/books/collected-works-alonzo-church>) 978-0-262-02564-5.)
11. Тьюринг 1936 в*The Undecidable*1965:132-134; Определение Тьюринга «кругового» можно найти на странице 119.
12. Тьюринг, Алан Матисон (1937). «О вычислимых числах с приложением к проблеме*Entscheidungsproblem*». *Труды Лондонского математического общества*. Серия 2.**42**(1):230–265.doi:10.1112/plms/s2-42.1.230 (<https://doi.org/10.1112/plms/s2-42.1.230>).S2CID73712 (<https://api.semanticscholar.org/CorpusID:73712>). (<https://doi.org/10.1112/plms/s2-42.1.230>) (<https://api.semanticscholar.org/CorpusID:73712>)
13. Тьюринг 1936 в*Неразрешимое*1965:145
14. Sipser 2006:137 отмечает, что «Машина Тьюринга может делать все, что может делать настоящий компьютер. Тем не менее, даже машина Тьюринга не может решить некоторые проблемы. В самом реальном смысле эти проблемы находятся за пределами теоретических пределов вычислений».
15. См. определение термина «иннингс» в Викисловаре.
16. AM Turing (июль 1948 г.). Интеллектуальные машины (http://www.alanturing.net/Turing_archive/archive/I/I32/L32-004.html) (отчет). Национальная физическая лаборатория. (http://www.alanturing.net/Turing_archive/archive/I/I32/L32-004.html)Здесь: стр.3-4
17. Иногда называется *таблицей действий* или *функцией перехода* .
18. Обычно пятерки [кортежи по 5]: $q_i a_j \rightarrow q_{i1} a_{j1} d_k$, но иногда четверки [кортежи по 4].
19. стр.149; в частности, Хопкрофт и Ульман предполагают, что δ не определено во всех состояниях от F
20. Гретшель, Мартин ; Ловас, Ласло ; Шрийвер, Александр (1993), *Геометрические алгоритмы и комбинаторная оптимизация* (<https://books.google.com/books?id=hWvmCAAAQBAJ&pg=PA281>) , Алгоритмы и комбинаторика, том. 2 (2-е изд.), Springer-Verlag, Берлин, номер номера : 10.1007/978-3-642-78240-4 (<https://doi.org/10.1007/978-3-642-78240-4>) , ISBN. (<https://books.google.com/books?id=hWvmCAAAQBAJ&pg=PA281>) (<https://doi.org/10.1007/978-3-642-78240-4>) 978-3-642-78242-8, г-н 1261419 (<https://mathscinet.ams.org/mathscinet-getitem?mr=1261419>)
21. Л. Торрес Кеведо. *Ensayos sobre Automática – ваше определение. Extension teórica de sus aplicaciones*, Revista de la Academia de Ciencias Exacta, Revista 12, стр. 391–418, 1914.
22. Торрес Кеведо. Л. (1915). «Essais sur l'Automatique - Sa définition. Etendue theorique de ses application» (<https://dicc.an.com/dicoport/Torres.htm>) , *Revue Générale des Sciences Pures et Appliquées* , vol. 2, стр. 601–611.
23. Более узкий вопрос, поставленный в десятой проблеме Гильберта , о диофантовых уравнениях , оставался нерешенным до 1970 года, когда связь между рекурсивно перечислимыми множествами и диофантовыми множествами была окончательно раскрыта.

Ссылки

Первичная литература, переиздания и компиляции

- B. Jack Copeland ed. (2004), *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*, Clarendon Press (Oxford University Press), Oxford UK, ISBN 0-19-825079-7 . Содержит статьи Тьюринга, а также черновик письма Эмилю Посту относительно его критики «конвенции Тьюринга» и исправлений Дональда У. Дэвиса к универсальной вычислительной машине Тьюринга.
- Мартин Дэвис (ред.) (1965), «Неразрешимое» , Raven Press, Хьюлетт, Нью-Йорк.
- Эмиль Пост (1936), «Конечные комбинаторные процессы — формулировка 1», *Журнал символической логики* , 1, 103–105, 1936. Перепечатано в *The Undecidable* , стр. 289 и далее.

- Эмиль Пост (1947), "Рекурсивная неразрешимость проблемы Туэ", *Журнал символической логики*, т. 12, стр. 1–11. Перепечатано в *The Undecidable*, стр. 293 и далее. В Приложении к этой статье Пост комментирует и дает исправления к статье Тьюринга 1936–1937 годов. В частности, см. сноску 11 с исправлениями к универсальному машинному кодированию и сноску 14 с комментариями к первому и второму доказательствам Тьюринга.
- Тьюринг, АМ (1936). «О вычислимых числах с приложением к Entscheidungsproblem» (<https://www.cs.ox.ac.uk/activities/ieg/e-library/sources/tp2-ie.pdf>) (PDF). *Труды Лондонского математического общества*. 2. **42** (опубликовано в 1937 г.): 230–265. doi : 10.1112/plms/s2-42.1.230 ([https://doi.org/10.1112/plms/s2-42.1.230](https://doi.org/10.1112%2Fplms%2Fs2-42.1.230)) . S2CID 73712 (<https://api.semanticscholar.org/CorpusID:73712>) .
- Тьюринг, АМ (1938). «О вычислимых числах с приложением к Entscheidungsproblem: исправление». *Труды Лондонского математического общества*. 2. **43** (6) (опубликовано в 1937 г.): 544–6. doi : 10.1112/plms/s2-43.6.544 ([https://doi.org/10.1112/plms/s2-43.6.544](https://doi.org/10.1112%2Fplms%2Fs2-43.6.544)) . Перепечатано в *The Undecidable*, стр. 115–154.
- Алан Тьюринг, 1948, «Интеллектуальные машины». Перепечатано в «Кибернетика: ключевые статьи». Ред. CR Evans и ADJ Robertson. Балтимор: University Park Press, 1968. стр. 31. Перепечатано в Turing, AM (1996). «Интеллектуальные машины, еретическая теория» (<https://doi.org/10.1093%2Fphilmat%2F4.3.256>) . *Philosophia Mathematica*. 4 (3): 256–260. doi : 10.1093/philmat/4.3.256 (<https://doi.org/10.1093%2Fphilmat%2F4.3.256>) . (<https://doi.org/10.1093%2Fphilmat%2F4.3.256>) (<https://doi.org/10.1093%2Fphilmat%2F4.3.256>)
- FC Hennie и RE Stearns . *Двухленточное моделирование многоленточных машин Тьюринга* . *JACM* , 13(4):533–546, 1966.

Теория вычислимости

- Булос, Джордж; Ричард Джеффри (1999) [1989]. *Вычислимость и логика* (https://archive.org/details/computabilitylog0000bool_r8u9) (3-е изд.). Кембридж, Великобритания: Cambridge University Press. ISBN 0-521-20402-X.
- Булос, Джордж; Джон Берджесс; Ричард Джеффри (2002). *Вычислимость и логика* (4-е изд.). Кембридж, Великобритания: Cambridge University Press. ISBN 0-521-00758-5. Некоторые части были значительно переписаны Берджессом. Представление машин Тьюринга в контексте «счётных машин» Ламбека (ср. Регистровая машина) и рекурсивных функций , показывающее их эквивалентность.
- Тейлор Л. Бут (1967), *Последовательные машины и теория автоматов* , John Wiley and Sons, Inc., Нью-Йорк. Инженерный текст для выпускников; охватывает широкий спектр тем, Глава IX *Машины Тьюринга* включает некоторую теорию рекурсии.
- Мартин Дэвис (1958). *Вычислимость и неразрешимость* . McGraw-Hill Book Company, Inc., Нью-Йорк.. На страницах 12–20 он приводит примеры таблиц из 5 кортежей для сложения, функции следования, вычитания ($x \geq y$), правильного вычитания (0 , если $x < y$), функции тождества и различных функций тождества, а также умножения.
- Дэвис, Мартин; Рон Сигал; Элейн Дж. Вейуер (1994). *Вычислимость, сложность, языки и логика: основы теоретической информатики* (2-е изд.). Сан-Диего: Academic Press, Harcourt, Brace & Company. ISBN 0-12-206382-1.
- Хенни, Фредрик (1977). *Введение в вычислимость* . Эддисон–Уэсли, Реддинг, Массачусетс. QA248.5H4 1977. На страницах 90–103 Хенни обсуждает UTM с примерами и блок-схемами, но без реального «кода».
- Хопкрофт, Джон ; Ульман, Джеффри (1979). *Введение в теорию автоматов, языки и вычисления* (1-е изд.). Addison–Wesley, Reading Mass. ISBN 0-201-02988-X. Сосредоточен вокруг вопросов машинной интерпретации «языков», NP-полноты и т. д.
- Хопкрофт, Джон Э.; Раджив Мотвани; Джеффри Д. Ульман (2001). *Введение в теорию автоматов, языки и вычисления* (2-е изд.). Массовое чтение: Эддисон–Уэсли. ISBN 0-201-44124-1.
- Стивен Клини (1952), *Введение в метаматематику* , North–Holland Publishing Company, Амстердам, Нидерланды, 10-е издание (с исправлениями 6-го переиздания 1971 г.). Текст для аспирантов; большая часть главы XIII *Вычислимые функции* посвящена доказательствам вычислимости рекурсивных функций с помощью машины Тьюринга и т. д.
- Кнут, Дональд Э. (1973). *Том 1/Фундаментальные алгоритмы: Искусство программирования* (2-е изд.). Реддинг, Массачусетс: Addison–Wesley Publishing Company.. Относительно роли машин Тьюринга в развитии вычислений (как аппаратных, так и программных) см. 1.4.5 *История и библиография*, стр. 225 и далее, и 2.6 *История и библиография*, стр. 456 и далее.
- Зохар Манна , 1974, *Математическая теория вычислений* . Переиздано, Дувр, 2003. ISBN 978-0-486-43238-0
- Марвин Мински , *Вычисления: конечные и бесконечные машины* , Prentice–Hall, Inc., NJ, 1967. См. главу 8, раздел 8.2 «Неразрешимость проблемы остановки».
- Христос Пападимитриу (1993). *Computational Complexity* (1-е изд.). Эддисон Уэсли. ISBN 0-201-53082-1. Глава 2: Машины Тьюринга, стр. 19–56.
- Хартли Роджерс-младший , *Теория рекурсивных функций и эффективная вычислимость* , The MIT Press, Кембридж, Массачусетс, издание в мягкой обложке 1987 г., оригинальное издание McGraw-Hill 1967 г., ISBN 0-262-68052-1 (pbk.)
- Майкл Сипсер (1997). *Введение в теорию вычислений* (<https://archive.org/details/introductiontoth00sips>) . PWS Publishing. ISBN 0-534-94728-X. Глава 3: Тезис Чёрча–Тьюринга, стр. 125–149.
- Стоун, Гарольд С. (1972). *Введение в организацию компьютеров и структуры данных* (1-е изд.). Нью-Йорк: McGraw–Hill Book Company. ISBN 0-07-061726-0.
- Питер ван Эмде Боас 1990, *Модели машин и симуляции* , стр. 3–66, в издании Яна ван Леувена , *Справочник по теоретической информатике, Том А: Алгоритмы и сложность* , Издательство MIT/Elsevier, [место?], ISBN 0-

444-88071-2 (Том А). QA76.H279 1990.

Тезис Чёрча

- Нахум Дершовиц; Юрий Гуревич (сентябрь 2008 г.). "Естественная аксиоматизация вычислимости и доказательство тезиса Чёрча" (<http://research.microsoft.com/en-us/um/people/gurevich/Opera/188.pdf>) (PDF) . *Бюллетень символической логики* . **14** (3) . Получено 15 октября 2008 г. .
- Роджер Пенроуз (1990) [1989]. *Новый разум императора* (2-е изд.). Oxford University Press, Нью-Йорк. ISBN 0-19-851973-7.

Малые машины Тьюринга

- Рогожин, Юрий, 1998, « Универсальная машина Тьюринга с 22 состояниями и 2 символами (https://web.archive.org/web/20050308141040/http://www.imt.ro/Romjist/Volum1/Vol1_3/turing.htm) », *Румынский журнал информационной науки и технологий* , 1(3), 259–265, 1998. (обзор известных результатов о малых универсальных машинах Тьюринга)
- Стивен Вольфрам , 2002, *Новый вид науки* (<http://www.wolframscience.com/nksonline/page-707>) , Wolfram Media, ISBN 1-57955-008-8
- Брунфилд, Джефф, Студент получает премию по математике (<http://www.nature.com/news/2007/071024/full/news.2007.190.html>) , *Nature* , 24 октября 2007 г.
- Джим Джайлс (2007), Простейший «универсальный компьютер» выигрывает для студентов 25 000 долларов (<http://www.newscientist.com/article/dn12826-simplest-universal-computer-wins-student-25000.html>) , New Scientist, 24 октября 2007 г.
- Алекс Смит, Универсальность машины Тьюринга Вольфрама 2, 3 (<http://www.wolframscience.com/prizes/tm23/TM23Proof.pdf>) , Заявка на соискание премии Wolfram 2, 3 за исследования машин Тьюринга.
- Воан Пратт, 2007, « Простые машины Тьюринга, универсальность, кодировки и т. д. (<http://cs.nyu.edu/pipermail/fom/2007-October/012156.html>) », список рассылки FOM. 29 октября 2007 г.
- Мартин Дэвис, 2007, « Наименьшая универсальная машина (<http://cs.nyu.edu/pipermail/fom/2007-October/012132.html>) » и «Определение универсальной машины Тьюринга», (<http://cs.nyu.edu/pipermail/fom/2007-October/012145.html>) рассылка FOM. 26–27 октября 2007 г.
- Аласдер Уркухарт, 2007 г. « Самая маленькая универсальная машина (<http://cs.nyu.edu/pipermail/fom/2007-October/012140.html>) », список рассылки FOM. 26 октября 2007 г.
- Гектор Зенил (Wolfram Research), 2007 г. « Наименьшая универсальная машина (<http://cs.nyu.edu/pipermail/fom/2007-October/012163.html>) », список рассылки FOM. 29 октября 2007 г.
- Тодд Роулэнд, 2007, « Путаница в FOM (<http://forum.wolframscience.com/showthread.php?s=&threadid=1472>) », форум Wolfram Science, 30 октября 2007 г.
- Оливье и Марк РЕЙНО, 2014 г. Программируемый прототип для создания машин Тьюринга. (http://www.machinedeturing.com/ang_default.asp) Архивировано (https://web.archive.org/web/20160114140003/http://machinedeturing.com/ang_default.asp) 14 января 2016 г. в лаборатории *Wayback Machine* LIMOS Университета Блеза Паскаля (Клермон-Ферран, Франция).

Другой

- Мартин Дэвис (2000). *Двигатели логики: математики и происхождение компьютера* (1-е изд.). WW Norton & Company, Нью-Йорк. ISBN 978-0-393-32229-3.
- Робин Ганди , «Слияние идей в 1936 году», стр. 51–102 в книге Рольфа Херкена , см. ниже.
- Стивен Хокинг (редактор), 2005, «Бог создал целые числа: математические прорывы, изменившие историю» , Running Press, Филадельфия, ISBN 978-0-7624-1922-7 . Включает статью Тьюринга 1936–1937 годов с краткими комментариями и биографией Тьюринга, написанной Хокингом.
- Рольф Херкен (1995). *Универсальная машина Тьюринга — обзор за полвека* . Springer Verlag. ISBN 978-3-211-82637-9.
- Эндрю Ходжес , *Алан Тьюринг: Энигма* , Саймон и Шустер , Нью-Йорк. См. главу «Дух истины» для истории, ведущей к его доказательству, и его обсуждения.
- Иварс Петерсон (1988). *Математический турист: снимки современной математики* (<https://archive.org/details/mathematicaltour00pete>) (1-е изд.). WH Freeman and Company, Нью-Йорк. ISBN 978-0-7167-2064-5.
- Роджер Пенроуз , *Новый разум императора: о компьютерах, разуме и законах физики* , Oxford University Press , Оксфорд и Нью-Йорк, 1989 (исправления 1990 года), ISBN 0-19-851973-7 .
- Пол Стратерн (1997). *Тьюринг и компьютер — Большая идея* (<https://archive.org/details/turingcomputer00stra>) . Anchor Books/Doubleday. ISBN 978-0-385-49243-0.
- Хао Ван , «Вариант теории вычислительных машин Тьюринга», *Журнал Ассоциации вычислительной техники* (JACM) 4, 63–92 (1957).
- Чарльз Петцольд , *Аннотированный Тьюринг* (<http://www.theannotatedturing.com/>) , John Wiley & Sons, Inc., ISBN 0-470-22905-5