

## ***Тема 5.1. Понятие алгоритма. Машины Тьюринга***

**Аннотация:** Понятие алгоритма. Машины Тьюринга их применение к словам и конструирование машин Тьюринга.

### ***Развитие теории алгоритмов***

Теория алгоритмов — наука, находящаяся на стыке математики и информатики, изучающая общие свойства и закономерности алгоритмов и разнообразные формальные модели их представления. Вместе с математической логикой теория алгоритмов образует теоретическую основу вычислительных наук.

Развитие теории алгоритмов начинается с доказательства Куртом Гёделем теорем о неполноте формальных систем, включающих арифметику, первая из которых была доказана в 1931 году. Возникшее в связи с этими теоремами предположение о невозможности алгоритмического разрешения многих математических проблем (в частности, проблемы выводимости в исчислении предикатов) вызвало необходимость стандартизации понятия алгоритма. Первые стандартизованные варианты этого понятия были разработаны в 1930-е годы в работах Алана Тьюринга, Эмиля Поста и Алонзо Чёрча. Предложенные ими машина Тьюринга, машина Поста и лямбда-исчисление оказались эквивалентными друг другу. Основываясь на работах Гёделя, Стивен Клини ввёл понятие рекурсивной функции, также оказавшееся эквивалентным вышеперечисленным.

В 1950-е годы существенный вклад в теорию алгоритмов внесли работы Колмогорова и Маркова.

Во второй половине XX века оформились следующие направления в теории алгоритмов:

- **Классическая теория алгоритмов** (формулировка задач в терминах формальных языков, понятие задачи разрешения, введение сложностных классов, формулировка в 1965 году Эдмондсом проблемы  $P=NP$ , открытие класса  $NP$ -полных задач и его исследование);
- **Теория асимптотического анализа алгоритмов** (критерии оценки алгоритмов, методы получения асимптотических оценок, в частности для рекурсивных алгоритмов, асимптотический анализ трудоемкости или

времени выполнения), в развитие которой внесли существенный вклад Кнут, Ахо, Хопкрофт;

- **Теория практического анализа вычислительных алгоритмов** (получение явных функции трудоёмкости, интервальный анализ функций, практические критерии качества алгоритмов, методика выбора рациональных алгоритмов), основополагающей работой в этом направлении, очевидно, следует считать фундаментальный труд Д. Кнута «Искусство программирования для ЭВМ»

### *Основные определения*

Рассмотрим некоторые определения алгоритма.

**Алгоритм** - это заданное на некотором языке конечное предписание, задающее конечную последовательность выполнимых элементарных операций для решения задачи, общее для класса возможных исходных данных.

**Алгоритм** (Колмогоров) – это всякая система вычислений, выполняемых по строго определенным правилам, которая после какого-либо числа шагов заведомо приводит к решению поставленной задачи.

**Алгоритм** (Марков) – это точное предписание, определяющее вычислительный процесс, идущий от варьируемых исходных данных к искомому результату.

Различные рассмотренные определения алгоритма, объединяет то, что во всех этих определениях присутствует в явном или неявном виде следующий ряд требований:

- алгоритм должен содержать конечное количество элементарно выполнимых предписаний, т.е. удовлетворять требованию конечности записи;
- алгоритм должен выполнять конечное количество шагов при решении задачи, т.е. удовлетворять требованию конечности действий;
- алгоритм должен быть единым для всех допустимых исходных данных, т.е. удовлетворять требованию универсальности;
- алгоритм должен приводить к правильному по отношению к поставленной задаче решению, т.е. удовлетворять требованию правильности.

Другие формальные определения понятия алгоритма связаны с введением специальных математических конструкций (машина Тьюринга, рекурсивно-вычислимы функции Черча) и

постулированием тезиса об эквивалентности такого формализма и понятия «алгоритм».

### *Машина Тьюринга*

Введение понятия машины Тьюринга явилось одной из первых и весьма удачных попыток дать точный математический эквивалент для общего интуитивного представления об алгоритме. Это понятие названо по имени английского математика, сформулировавшего его в 1937 г., за 9 лет до появления первой электронно-вычислительной машины.

Машина Тьюринга есть математическая (воображаемая) машина, а не машина физическая. Она состоит из:

1) бесконечной в обе стороны ленты, разбитой на ячейки. В каждой ячейке записан один из символов алфавита:  $a_0, a_1, \dots, a_n$ . В алфавите всегда существует пустой символ (или пробел), обычно это  $a_0$  (считается, что этим символом лента заполнена в обе стороны до бесконечности, даже если эти символы не написаны).

2) набора  $q_0, q_1, \dots, q_m$  - внутренних состояний машины. Какое-то из внутренних состояний конечное (обычно это  $q_0$ ), какое-то – начальное (чаще всего  $q_1$ ). Для каждой задачи берётся свой набор внутренних состояний, но начальное и конечное состояния существуют всегда. Эти состояния могут и совпасть.

3) управляющей головки,двигающейся в обе стороны вдоль ленты в дискретные моменты времени. Головка читает символы алфавита, анализирует их, записывает новые символы и двигается по ленте по программе. В зависимости от внутреннего состояния в данный момент и от символа на воспринимаемой ячейке, в следующий момент машина переходит в новое состояние, записывает в ячейку новый символ. Попад в конечное состояние, машина прекращает работу.

Конфигурация на ленте состоит из:

- 1) последовательности символов алфавита на ленте.
- 2) текущего внутреннего состояния машины.
- 3) номера воспринимаемой ячейки (позиции ячейки).

Если, в данный момент машина находится во внутреннем состоянии  $q_i$  и читается ячейка  $a_j$ , в следующий момент машина переходит в состояние  $q_k$ , а символ на ленте заменяется на  $a_m$ , то такая **команда** записывается следующим образом:  $q_i a_j \rightarrow q_k a_m S$ , где

$S=\{L, R, C\}$  (влево, вправо, стоп), команда стоп часто опускается. При этом машина одну конфигурацию переводит в другую.

**Программа** – это совокупность всех команд, выполняемых машиной. Часто записывается в виде двумерной таблицы.

**Пример:** Дана машина Тьюринга с внешним алфавитом  $A = \{0,1\}$  (здесь 0 — символ пустой ячейки), алфавитом внутренних состояний  $Q = \{q_0, q_1, q_2\}$  и со следующей программой:

	$q_1$	$q_2$
0	$q_2 0P$	$q_0 1$
1	$q_1 1P$	$q_2 1P$

Посмотрим, в какое слово переработает эта машина слово 101, если изначально машина находится в состоянии  $q_1$  и просматривается крайняя правая ячейка.

Будем последовательно выписывать конфигурации машины при переработке ею этого слова. Изначально:

$q_1$								
	1	0	1					

На первом шаге по таблице находим, что должна примениться команда  $q_1 1 \rightarrow q_1 1P$ , т.е. мы последовательно делаем следующее: заменяем символ на 1 (у нас не меняется), заменяем состояние машины на  $q_1$  (опять не меняется) и сдвигаемся по ленте вправо. В результате получим следующую конфигурацию (с учетом того, что пустая ячейка у нас содержит 0):

$q_1$								
	1	0	1	0				

Теперь будет действовать команда:  $q_1 0 \rightarrow q_2 0P$  (0 заменяем на 0, состояние  $q_1$  - на состояние  $q_2$  и сдвигаемся вправо) и на машине создается конфигурация:

$q_2$								
	1	0	1	0	0			

На третьем шаге получим по таблице команду:  $q_2 0 \rightarrow q_0 1$ . В результате чего создается конфигурация:

$q_0$								

	1	0	1	0	1			
--	---	---	---	---	---	--	--	--

Эта конфигурация является заключительной, потому что машина оказалась в состоянии остановки  $q_0$ .

Таким образом, исходное слово 101 переработано машиной в слово 10101.

Создание (синтез) машин Тьюринга (т.е. написание соответствующих программ) является задачей значительно более сложной, нежели процесс применения данной машины к данным словам и обычно рассматривается в предмете «Математическая логика и теория алгоритмов».

**Тезис Тьюринга (основная гипотеза теории алгоритмов):** Для нахождения значений функции, заданной в некотором алфавите, тогда и только тогда существует какой-нибудь алгоритм, когда функция является вычислимой по Тьюрингу, т.е. когда она может вычисляться на подходящей машине Тьюринга.

Это означает, что строго математическое понятие вычислимой (по Тьюрингу) функции является по существу идеальной моделью взятого из опыта понятия алгоритма. Данный тезис есть не что иное, как аксиома, постулат, выдвигаемый нами, о взаимосвязях нашего опыта с той математической теорией, которую мы под этот опыт хотим подвести. Конечно же данный тезис в принципе не может быть доказан методами математики, потому что он не имеет внутриматематического характера (одна сторона в тезисе — понятие алгоритма — не является точным математическим понятием). Он выдвинут исходя из опыта, и именно опыт подтверждает его состоятельность. Точно так же, например, не могут быть доказаны и математические законы механики; они открыты Ньютоном и многократно подтверждены опытом.

Впрочем, не исключается принципиальная возможность того, что тезис Тьюринга будет опровергнут. Для этого должна быть указана функция, которая вычислима с помощью какого-нибудь алгоритма, но невычислима ни на какой машине Тьюринга. Но такая возможность представляется маловероятной (в этом одно из значений гипотезы): всякий алгоритм, который будет открыт, может быть реализован на машине Тьюринга.

Вопросы для самоконтроля:

1. Понятие алгоритма. Основные определения.
2. Машины Тьюринга их применение к словам.

### 3. Основы конструирования машин Тьюринга.