Лекция 6

Машины Тьюринга. Определение конфигурации.

Опр Машина Тьюринга - это набор $(A; Q; Q_f; \delta; q_0, \Lambda)$, где:

- A - конечное множество, называемое алфавитом МТ
- Q - конечное множество, называемое множеством состояний МТ. $A \cap Q = \varnothing$
- Q_f - множество финальных состояний МТ. $Q_f \subseteq Q$
- δ - функция (таблица) переходов. $\delta: A \times Q \to A \times \{-1; 0; 1\} \times Q$
- q_0 - начальное состояние; $q_0 \in Q$
- Λ - пустой символ; $\Lambda \in A$

Пояснения, как это выглядит.

Есть память в виде бесконечной ленты с ячейками. В ячейках написаны символы из алфавита МТ. Над какой-то ячейкой находится считывающее устройство. Визуализация:

				q					
...	Λ	Λ	x_1	x_2	x_3	x_4	x_5	Λ	...

Машина Тьюринга получает на вход слово x (конкретную ленту с символами и положением считывающего устройства). Совершает действия над лентой и на выход даёт некое слово y.

Опр Конкретная лента с памятью + положение считывающего устройства называется конфигурацией Машины Тьюринга. (неформальное)

Опр Конфигурацией МТ $(A; Q; Q_F; \delta; q_0; \lambda)$ называется слово w в алфавите $A \cup Q$, которое содержит ровно один символ из множества Q, а первый и последний символы w не являются пустыми. (формальное)

Начальная конфигурация МТ имеет вид $q_0 u$, где $u \in (A \setminus \{\lambda\})^*$ — конечное слово, которое называется входом МТ.

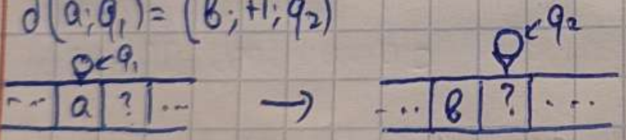
Пояснения, как это работает:

Любое действие МТ — отображение $Q \times A \to A \times \{-1; 0; 1\} \times Q$.

Сначала есть в ячейке, на которую указывает считывающее устройство, какой-то символ $a_1 \in A$ и состояние $q_1 \in Q$. За ход мы заменяем символ a_1 на какой символ $a_2 \in A$ (возможно $a_1 = a_2$), меняем состояние q_1 на q_2 (возможно $q_1 = q_2$) и делаем сдвиг считывающего устройства на 1 влево или вправо (-1 или 1), либо остаёмся в той же ячейке.

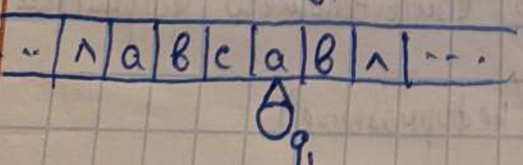
Пример хода:

$\delta(a; q_1) = (b; +1; q_2)$



Считаем, что если МТ приходит в одно из финальных состояний, то она сразу останавливает работу.

Пример конфигурации:



$\Leftrightarrow abcq_1ab$

Как, продвигаясь по ленте, понять, что слово закончилось, если в слове могут быть пустые символы?

Решение такое — введём символы ▶ и ◀, которые обозначают начало и конец слова соответственно.

<u>Пример программы на МТ</u>. Сложение двух чисел в 1-ричной системе счисления (записано просто количество „1"):

Вход: $\blacktriangleright \underbrace{111111}_{n} \# \underbrace{11111}_{m} \blacktriangleleft$

Выход: $\blacktriangleright \underbrace{11111111111}_{n+m} \blacktriangleleft$

Напишем программу без учёта ▶ и ◀. Зададим МТ полностью:

δ:

Q \ A	1	#	λ
q_0	$(1;+1;q_0)$	$(1;+1;q_0)$	$(\lambda;-1;q_1)$
q_1	$(\lambda;0;q_f)$	$(\#;0;q_{er})$	$(\#;0;q_{er})$
q_f	—	—	—

$A = \{\lambda; 1; \#\}$

$Q = \{q_0; q_1; q_f; q_{er}\}$

$Q_f = \{q_f; q_{er}\}$

q_f — финальное состояние при успешном выполнении

q_{er} — финальное состояние, выкидываемая ошибка

<u>Пример программы на МТ</u>. Вычитание из числа в бинарной записи 1.

Вход: ... | 1 | 0 | 1 | 1 | 0 | 1 | 0 | ...
q_0

q_f
Выход: ... | 1 | 0 | 1 | 1 | 0 | 0 | 1 | ...

(Пример для конкретных чисел, программа работает в общем случае)

δ_1: $Q \backslash A$	1	0	Λ
q_0	$(1;+1;q_0)$	$(0;+1;q_0)$	$(\Lambda;-1;q_1)$
q_1	$(0;0;q_f)$	$(1;-1;q_1)$	$(1;0;q_{er})$

$A = \{\Lambda; 0; 1\}$

$Q = \{q_0; q_1; q_f; q_{er}\}$

$Q_F = \{q_f; q_{er}\}$

<u>Утв</u> Множество всех Машин Тьюринга счётно.

<u>Опр</u> Расширенной конфигурацией МТ назовём двустороннюю бесконечную последовательность $\ldots \Lambda w \Lambda \ldots$, где w - конфигурация МТ, а на месте многоточий стоят пустые символы Λ.

Отображение на расширенных конфигурациях МТ:

$$d_M(uqv) = \begin{cases} ua'q'v' ; & \delta(a;q) = (a';+1;q') \\ uq'a'v' ; & \delta(a;q) = (a';0;q') \\ u'q'ba'v' ; & \delta(a;q) = (a';-1;q') \end{cases}$$

где $v = av'$; $u = u'b$

Отображение на конфигурациях формализует работу МТ за один такт.

Работа МТ M полностью задаётся начальной конфигурацией и отображением на конфигурациях и может быть описана конечной или бесконечной последовательностью конфигураций:

$w_0; w_1 \ldots w_n \ldots$, где $w_{i+1} = d_M(w_i)$

Опр Последовательность $w_0, w_1 \dots w_n \dots$ называется протоколом работы МТ.

Опр Результатом работы МТ является слово, которое получается из последней конфигурации вычёркиванием символа состояния головки.

Кодирование МТ:

Берём конечный алфавит: $A = \{a_1 \dots a_n\}$; $|A| \in \mathbb{N}$

Конечное множество состояний: $Q = \{q_0, q_1 \dots q_m\}$; $|Q| \in \mathbb{N}$

Закодируем МТ следующим образом:

$$\underbrace{q_0 q_1 \dots q_m}_{Q} \underbrace{\#}_{\text{разделитель}} \underbrace{a_1 a_2 \dots a_n}_{A} \# \underbrace{q_{f_1} q_{f_2} \dots q_{f_k}}_{Q_F} \# \underbrace{a q_0 b +1 q_1}_{\text{один переход из } \delta} \# \dots \# \dots$$

$\Downarrow$

У нас получилось задать МТ полностью на конечной ленте.