# Логика и алгоритмы, лекция 26

лектор: Кудинов Андрей Валерьевич

1 июня 2021 г.

# План лекции:

- Универсальная машина Тьюринга
- Главность универсальной МТ
- m-сводимость и m-полнота
- Теорема Клини о неподвижной точке
- Арифметика Пеано

# Кодирование машин Тьюринга

Машина $M = \langle Q, \Sigma, P, q_0, q_1 \rangle$ задаётся

- $Q = \{q_0, \dots, q_s\}$ — внутр. состояния;
- $\Sigma = \{a_0, \dots, a_r\}$ — рабочий алфавит;
- $P = \{p_0, \dots, p_{s(r+1)}\}$ — набор команд.

$q_1$ — нач., $q_0$ — кон., $a_0 = \#$ — пробел.

# Кодирование $Q$ и $\Sigma$

Алфавит программ есть $\Pi := \{\rightarrow, L, N, R, q, a, \mathbf{1}\}$.

Сопоставим элементам $Q$ и $\Sigma$ следующие коды в алфавите $\Pi$:
$q_i \longmapsto q\mathbf{1}^i; \quad a_j \longmapsto a\mathbf{1}^j.$

Слово $x \in \Sigma^*$ кодируется конкатенацией $Code(x)$ кодов всех его букв, например $Code(a_2a_0a_1) = a\mathbf{11}aa\mathbf{1}$.

# Коды команд

**Код команды** $q_i a_k \to q_j a_l \nu$, где $\nu \in \{L, N, R\}$, есть слово $q\mathbf{1}^i a\mathbf{1}^k \to q\mathbf{1}^j a\mathbf{1}^l \nu$ в алфавите $\Pi$.

Код команды $p \in P$ обозначим $Code(p)$.

# Коды машин

Код машины $M$ есть конкатенация кодов всех её команд, то есть $Code(M) := Code(p_0) \dots Code(p_{s(r+1)})$.

**Утверждение**

Отображение $M \longmapsto Code(M)$ инъективно.

В частности, по $Code(M)$ однозначно восстанавливаются рабочий алфавит, множество внутренних состояний, команды и т.д.

### Утверждение

Множество кодов всевозможных машин Тьюринга (выбранного нами формата) есть разрешимое подмножество $\Pi^*$.

# Функция, вычислимая машиной Тьюринга

Пусть $\Delta \subset \Sigma$ и $\# \notin \Delta$.

$M$ **чисто вычисляет** частичную функцию $f : \Delta^* \to \Delta^*$, если для каждого $x \in \Delta^*$

- если $x \in dom(f)$, то начав работу в конфигурации $q_1\#x$, машина $M$ останавливается в конфигурации $q_0\#f(x)$;
- если $x \notin dom(f)$, то машина $M$ не останавливается.

$M$ вычисляет частичную функцию $f : \Delta^* \to \Delta^*$, если для каждого $x \in \Delta^*$

- если $x \notin dom(f)$, то начав работу в конфигурации $q_1\#x$, машина $M$ не останавливается;
- если $x \in dom(f)$, то машина $M$ останавливается, на ленте написано слово $y = f(x)$, слева и справа от него стоят символы не из $\Delta^*$, а головка остановилась внутри или непосредственно перед $y$.

# Обозначения

$M_\Delta(x)$ есть результат работы $M$ на слове $x \in \Delta^*$.

$M_\Delta : \Delta^* \to \Delta^*$ — частичная функция, вычислимая $M$.

**Замечание 26.1**

$M_\Delta$ определена для любой машины $M$ с рабочим алфавитом $\Sigma \supset \Delta$.

## Утверждение

Для любой МТ $M$ и $\Delta$ можно указать машину $M'$ вычисляющую функцию $M_\Delta$ чисто.

- Преобразуем $M$ так, чтобы $M$ не печатала $\#$ (добавив «двойник» пробела).
- Добавим к программе $M$ инструкции, определяющие по завершении работы $M$ слово $M_\Delta(x)$ и удаляющие весь мусор слева и справа до символов $\#$.

# Универсальная машина Тьюринга

**Универсальная машина** $U_\Delta$ с рабочим алфавитом, содержащим $\Pi \cup \Delta \cup \{\$\}$, для любой МТ $M$ и слова $x \in \Delta^*$ (чисто) вычисляет результат работы машины $M$ на входе $x$, то есть частичную функцию

$$Code(M)\$x \longmapsto M_\Delta(x).$$

Другими словами:

- Если $U_\Delta$ начинает работу в конфигурации $q_1\#Code(M)\$x$ для $x \in \Delta^*$, то заключительная конфигурация $q_0\#M_\Delta(x)$;
- Иначе $U_\Delta$ зацикливается.

Алгоритм работы машины $U_\Delta$:

- Читаем входное слово вплоть до первого пробела и проверяем, что оно имеет вид $Code(M)\$x$ для $x \in \Delta^*$. Если нет, зацикливаемся.
- Эмулируем работу $M$ на входе $x$, пользуясь частью ленты справа от \$ для записи кодов конфигураций $M$.

- В случае завершения работы $M$ на входе $x$ с результатом $y$ выделяем слово $Code(y)$ из кода заключительной конфигурации $M$.
- Преобразуем $Code(y)$ в $y$.

# Главность универсальной МТ

Пусть $\Delta = \{1\}$ и МТ $M$ вычисляет $g(e,x)$ в унарной записи, то есть $M_\Delta(\overline{c(e,x)}) \simeq \overline{g(e,x)}$.

Сопоставим МТ $M$ машину $M[n]$, которая для данного входа $\overline{x}$ вычисляет $\overline{c(n,x)}$, а далее работает как $M$. Преобразование $n \mapsto Code(M[n])$ является тотальной вычислимой функцией.

Пусть $\phi_\Pi : \mathbb{N} \to \Pi^*$ произвольная вычислимая тотальная биекция, такая что обратная биекция тоже вычислима.
Имеем
$$M_\Delta(\overline{c(e,x)}) \simeq M[e]_\Delta(\overline{x}) \simeq U_\Delta(Code(M[e])\$\overline{x}).$$
Вспомним, что универсальная функция $F(i,n) := |U_\Delta(\phi_\Pi(i)\$\overline{n})|$.

Отсюда $g(e,x) \simeq F(s(e),x)$, где
$$s(e) = \phi_\Pi^{-1}(Code(M[e])).$$

# m-сводимость

Говорят, что множество $A$ натуральных чисел $m$-сводится к другому множеству $B$ натуральных чисел, если существует всюду определённая вычислимая функция $f : \mathbb{N} \to \mathbb{N}$ с таким свойством:

$$x \in A \iff f(x) \in B$$

для всех $x \in \mathbb{N}$. Обозначение: $A \leqslant_m B$.

# m-сводимость

Говорят, что множество $A$ натуральных чисел $m$-сводится к другому множеству $B$ натуральных чисел, если существует всюду определённая вычислимая функция $f : \mathbb{N} \to \mathbb{N}$ с таким свойством:

$$x \in A \iff f(x) \in B$$

для всех $x \in \mathbb{N}$. Обозначение: $A \leqslant_m B$.

Свойства:

- $\leqslant_m$ — рефлексивно и транзитивно;
- $B$ — разрешима (перечислима) и $A \leqslant_m B \Rightarrow A$ — разрешима (перечислима);
- $B$ — неразреш. (неперечис.) и $A \leqslant_m B \Leftarrow A$ — неразреш. (неперечис.);
- $A \leqslant_m B \iff \mathbb{N} \setminus A \leqslant_m \mathbb{N} \setminus B$;
- $A$ — разрешима и $B \neq \varnothing, \mathbb{N} \Rightarrow A \leqslant_m B$.

Пусть $F$ — главная универсальная вычислима функция.
$A = \{e \mid F_e(0) = 0\}$. Что можно сказать про множество $A$?

# $m$-полные множества

Множество $A$ называется $m$-полным (в классе перечислимых множеств), если для любого перечислимого множества $B$ верно, что $B \leqslant_m A$.

## $m$-полные множества

Множество $A$ называется $m$-полным (в классе перечислимых множеств), если для любого перечислимого множества $B$ верно, что $B \leqslant_m A$.

**Теорема 26.2**

Для главной УВФ $F(e,x)$ множество $K = \{x \mid e \mid F(e,x) \text{ определено}\}$ является $m$-полным.

$K$ — перечислимо.

Предположим, что $A$ — перечислимо. Рассмотрим функцию

$$g(n,x) = \begin{cases} \text{неопред.,} & \text{если } n \in A; \\ x, & \text{если } n \notin A; \end{cases}$$

По главность $F$ найдется тотальная функция $f : \mathbb{N} \to \mathbb{N}$, т.ч.

$$g(n,x) \simeq F(f(n),x).$$

$$g(n,x) = \begin{cases} \text{неопред.,} & \text{если } n \in A; \\ 1, & \text{если } n \notin A; \end{cases}$$

$$g(n,x) \simeq F(f(n),x).$$

Покажем, что

$$x \in A \iff f(x) \in K$$

# Теорема Клини о неподвижной точке

**Теорема 26.3 (Клини)**

Пусть $F$ — главная УВФ для класса $\mathrm{Com}(\mathbb{N}, \mathbb{N})$, а $h$ — всюду определённая вычислимая функция одного аргумента. Тогда существует такое число $m$, что $F_n = F_{h(n)}$, то есть $n$ и $h(n)$ — номера одной функции.

# Теорема Клини о неподвижной точке

## Теорема 26.3 (Клини)

Пусть $F$ — главная УВФ для класса $\mathrm{Com}(\mathbb{N}, \mathbb{N})$, а $h$ — всюду определённая вычислимая функция одного аргумента. Тогда существует такое число $m$, что $F_n = F_{h(n)}$, то есть $n$ и $h(n)$ — номера одной функции.

# Программа печатающая свой номер (текст)

**Следствие 26.4**

Существует $n$, такой что $F(n, x) = n$ при любом $x$.

# Программа печатающая свой номер (текст)

## Следствие 26.4

Существует $n$, такой что $F(n, x) = n$ при любом $x$.

# Арифметика Пеано PA

Сигнатура: $0$, $S$, $+$, $\cdot$, $\mathrm{Exp}$, $\leq$, $=$

Стандартная модель: $(\mathbb{N}; 0, S, +, \cdot, \mathrm{Exp}, \leq, =)$, где $S(x) = x + 1$ и $\mathrm{Exp}(x) = 2^x$.

# Аксиомы PA

1. $\neg S(a) = 0, \quad S(a) = S(b) \to a = b,$
2. $a + 0 = a, \quad a + S(b) = S(a + b),$
3. $a \cdot 0 = 0, \quad a \cdot S(b) = a \cdot b + a,$
4. $\mathrm{Exp}(0) = S(0), \quad \mathrm{Exp}(S(a)) = \mathrm{Exp}(a) + \mathrm{Exp}(a),$
5. $a \leq 0 \leftrightarrow a = 0,$
6. $a \leq S(b) \leftrightarrow (a \leq b \vee a = S(b)),$
7. ( Схема аксиом индукции)

   $A[a/0] \wedge \forall x\, (A[a/x] \to A[a/S(x)]) \to \forall x\, A[a/x],$

   для любой формулы $A$.

# Арифметика Робинсона

Теория $Q$ получается из PA заменой схемы индукции единственной аксиомой:

$$a \leq b \vee b \leq a.$$

## Упражнение 26.1

Показать, что $\mathsf{PA} \vdash Q$.

# Решение

(1) Сначала покажем индукцией по $x$, что $\forall x\,(a \leq x \leftrightarrow a = x \vee S(a) \leq x)$.

(2) Затем покажем индукцией по $x$, что $\forall x\,(a \leq x \vee x \leq a)$.

Заметим, что из (1) следует $a \leq a$ и $a \leq S(a)$.

# Вывод (1)

Базис: $a \leq 0 \leftrightarrow a = 0 \vee S(a) \leq 0$. Поскольку $S(a) \leq 0 \to S(a) = 0$, имеем $\neg S(a) \leq 0$.

# Вывод (1)

Базис: $a \leq 0 \leftrightarrow a = 0 \vee S(a) \leq 0$. Поскольку $S(a) \leq 0 \to S(a) = 0$, имеем $\neg S(a) \leq 0$.

Шаг: эквивалентно преобразуем

1. $a \leq S(x)$
2. $a \leq x \vee a = S(x)$ (аксиома)
3. $(a = x \vee S(a) \leq x) \vee a = S(x)$ (пр. инд.)
4. $S(a) = S(x) \vee S(a) \leq x \vee a = S(x)$ (аксиома)
5. $S(a) \leq S(x) \vee a = S(x)$

# Вывод (2)

Базис: $a \leq 0 \vee 0 \leq a$ поскольку $0 \leq a$.

Шаг:

1. $a \leq x \vee x \leq a \quad$ (пр. инд.)
2. $x \leq a \to (a = x \vee S(x) \leq a)$ (1)
3. $a \leq x \vee a = x \vee S(x) \leq a$
4. $a \leq x \to a \leq S(x) \quad$ (аксиома)
5. $a = x \to a \leq S(x) \quad$ (из (1))
6. $a \leq S(x) \vee S(x) \leq a$