Логика и алгоритмы, лекция 22

лектор: Кудинов Андрей Валерьевич

11 мая 2021 г.

План

План лекции:

- Неформальное представление об алгоритмах.
- Вычислимые функции
- Вычислительные модели
- Тезис Чёрча–Тьюринга
- Машины Тьюринга

Неформальное представление об алгоритмах.

- Алгоритм есть

Неформальное представление об алгоритмах.

- Алгоритм есть строго определенное конечное предписание выполнить некоторую последовательность действий (может быть бесконечную).

Неформальное представление об алгоритмах.

- Алгоритм есть строго определенное конечное предписание выполнить некоторую последовательность действий (может быть бесконечную).
- Для данного алгоритма $\mathcal{A}$ определены:
  - область возможных исходных данных X;
  - область возможных значений Y.

  В качестве данных обычно рассматриваются слова $X = \Sigma^*$, где Σ — конечный алфавит, или числа $X = \mathbb{N}^n$.

Свойства алгоритма

- Процесс применения алгоритма $\mathcal{A}$ к данным $x \in X$ происходит по шагам.
- Процесс или заканчивается после конечного числа шагов с результатом $y \in Y$, или останавливается без результата или продолжается бесконечно.
- Таким образом, с алгоритмом $\mathcal{A}$ связывается частичная функция $f : X \to Y$.
  Мы будем говорить:

  «Алгоритм $\mathcal{A}$ вычисляет функцию f.»

Частичные функции

Определение

Частичной функцией $f : X \to Y$ называется подмножество $f \subseteq X \times Y$ такое, что из $\langle x, y_1 \rangle \in f$ и $\langle x, y_2 \rangle \in f$ следует $y_1 = y_2$.

Пишем $f(x) = y$ вместо $\langle x, y \rangle \in f$;

$!f(x)$ вместо $\exists y\, f(x) = y$.

Областью определения частичной функции f называется множество $dom(f) := \{x \in X : \exists y \in Y\ \langle x, y\rangle \in f\}$.

Областью значений частичной функции f называется множество $rng(f) := \{y \in Y : \exists x \in X\ \langle x, y\rangle \in f\}$.

Вычислимые функции

Частичная функция $f : X \to Y$ **вычислима**, если она вычисляется некоторым алгоритмом.

В частности, можно говорить о вычислимых функциях $f : \Sigma^* \to \Sigma^*$, $f : \mathbb{N}^k \to \mathbb{N}$ и т.д.

на входе $x \in X$

A заканчивает работу, если $\exists!\, f(x)$ и выдает $f(x)$

A незакан.. раб. (зацикливается), если $f(x)$ неопр. $x \notin dom(f)$

Вычислительные модели

- Машины Тьюринга (А. Тьюринг, Э. Пост)
- Частично рекурсивные функции (К. Гёдель, С. Клини)
- Лямбда-исчисление (А. Чёрч)
- Алгорифмы Маркова
- Машины с неограниченными регистрами
- Pascal, C, Java, Lisp, Python, ...

Эквивалентность вычислительных моделей

Теорема

Каждая из вышеперечисленных моделей определяет один и тот же класс вычислимых частичных функций $f : \Sigma^* \to \Sigma^*$.

Такие модели (языки программирования) называются полными по Тьюрингу.

Тезис Чёрча–Тьюринга

Тезис

Любая вычислимая в интуитивном смысле частичная функция $f : \Sigma^* \to \Sigma^*$ вычислима на машине Тьюринга.

Замечание

Это утверждение не является математическим, но говорит об адекватности математической модели (вычислимости по Тьюрингу) реальному явлению (вычислимости).

Тезис Чёрча–Тьюринга

Тезис

Любая вычислимая в интуитивном смысле частичная функция $f : \Sigma^* \to \Sigma^*$ вычислима на машине Тьюринга.

Замечание

Это утверждение не является математическим, но говорит об адекватности математической модели (вычислимости по Тьюрингу) реальному явлению (вычислимости).

Все попытки построения более общих вычислительных моделей неизбежно приводили к тому же самому классу вычислимых функций.

Физический тезис Чёрча–Тьюринга

Текущему уровню знаний не противоречит и более сильный

Тезис

Всякая функция $f : \Sigma^* \to \Sigma^*$, вычислимая на (идеализированном) физически реализуемом устройстве, вычислима на машине Тьюринга.

Замечание

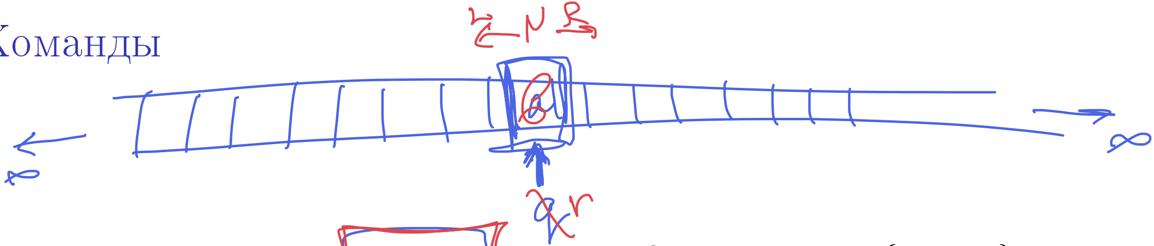
Физический тезис предполагает возможность аналогового вычисления, квантово–механические эффекты и т.д.

Машины Тьюринга

Машина Тьюринга задаётся конечными

- рабочим алфавитом Σ, содержащим символ $\#$ (пробел);
- множеством состояний Q, содержащим состояния q_1 (начальное) и q_0 (конечное);
- набором команд (программой) P.

Команды



- Команды имеют вид $\boxed{qa \to rb\nu}$, где $q, r \in Q$, $a, b \in \Sigma$ и $\nu \in \{L, N, R\}$.
  «прочтя символ a в состоянии q перейти в состояние r, заменить содержимое ячейки на b и сместиться влево (L), остаться на месте (N) или сместиться вправо (R) на одну ячейку, в зависимости от значения ν»

$q_1 \# \to q_2 0 R$

- Требуется, чтобы в программе P была ровно одна команда с левой частью qa для каждого $q \in Q \setminus \{q_0\}$ и $a \in \Sigma$.

Соглашение: команды вида $qa \to qaN$, приводящие к зацикливанию, можно не указывать.

Машина Тьюринга есть набор $M = \langle Q, \Sigma, P, q_0, q_1 \rangle$.

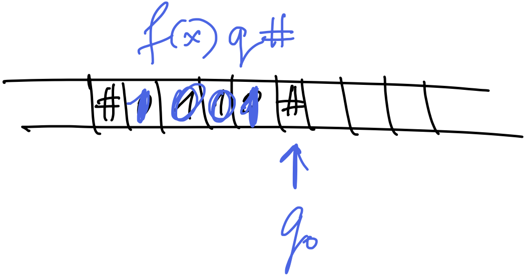
Пример машины Тьюринга

Пусть $\Sigma = \{\#, 0, 1\}$, $Q = \{q_0, q_1\}$, а P состоит из следующих команд:

$$\begin{array}{rcl} q_1\# & \mapsto & q_1\#R \\ q_1 0 & \mapsto & q_1 1R \\ q_1 1 & \mapsto & q_1 0R \end{array}$$

Что делает эта машина Тьюринга?

Модифицируем программу.



$$f : \{0,1\}^* \to \{0,1\}^*$$

Пример машины Тьюринга

Пусть $\Sigma = \{\#, 0, 1\}$, $Q = \{q_0, q_1, q_2\}$, а P состоит из следующих команд:

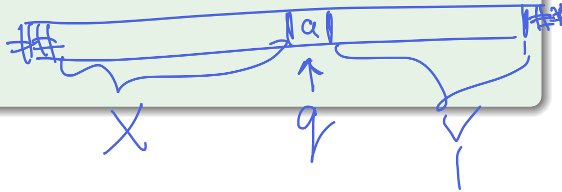
$$\begin{aligned}
q_1\# &\mapsto q_1\#R\\
q_10 &\mapsto q_21R\\
q_11 &\mapsto q_20R\\
q_20 &\mapsto q_21R\\
q_21 &\mapsto q_20R\\
q_2\# &\mapsto q_0\#N
\end{aligned}$$

Конфигурации

Предположение: лента содержит лишь конечное число символов, отличных от $\#$.

Конфигурация машины M определяется содержимым ленты, состоянием и положением головки. Конфигурация записывается словом вида $XqaY$, где

- $XaY \in \Sigma^*$ есть содержимое ленты,
- $q \in Q$ есть состояние M,
- головка обозревает символ a.



Функция, вычислимая машиной Тьюринга

$\Sigma \setminus \Delta = \{\#, \ldots\}$

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

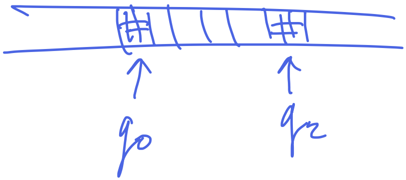
M **вычисляет** частичную функцию $f : \Delta^* \to \Delta^*$, если для каждого $x \in \Delta^*$

- 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. в конфиг. $q_1\#x$

f – вычислима, если $\exists M$ – маш. Тью, кот. её вычисляет.

Машина M из примера (почти) вычисляет функцию $\mathrm{neg} : \{0,1\}^* \to \{0,1\}^*$, заменяющую в данном слове 0 на 1 и 1 на 0. Чтобы вернуть головку в начало модифицируем M:

$$
\begin{array}{rcl}
q_1\# & \mapsto & q_1\#R \\
q_1 0 & \mapsto & q_2 1R \\
q_1 1 & \mapsto & q_2 0R \\
q_2 0 & \mapsto & q_2 1R \\
q_2 1 & \mapsto & q_2 0R \\
\hline
q_2\# & \mapsto & q_3\#L \\
q_3 0 & \mapsto & q_3 0L \\
q_3 1 & \mapsto & q_3 1L \\
q_3\# & \mapsto & q_0\#N
\end{array}
$$



Упражнения

Построить машины Тьюринга, вычисляющие следующие функции над алфавитом $\{0, 1\}$:

- $f(x) = xx$ (копирование слова)
- $g(x_1 \ldots x_n) = \sum_{i=1}^{n} x_i \mod 2$
  (сумма битов по модулю 2)

Логика и алгоритмы, лекция 25

лектор: Кудинов Андрей Валерьевич

25 мая 2021 г.

План лекции:

- Универсальная функция и неразрешимое множество
- Пара неотделимых перечислимых множеств
- Главные универсальные функции
- Теорема Райса–Успенского

Универсальная функция и неразрешимое множество

$\mathrm{Com}(\mathbb{N},\mathbb{N})$

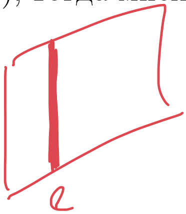
Пусть $\mathcal{F}$ — счётное семейство част. функций $f : X \to Y$.

Универсальной функцией для $\mathcal{F}$ называем такую функцию $F : \mathbb{N} \times X \to Y$, что

- Для любого $e \in \mathbb{N}$ функция $F_e(x) := F(e, x)$ принадлежит $\mathcal{F}$.
- $\forall f{\in}\mathcal{F}\ \exists e{\in}\mathbb{N}\ \forall x{\in}X\ \ f(x) \simeq F(e, x)$.

Пусть F — универсальная вычислима функция для $\mathrm{Com}(\mathbb{N}, \mathbb{N})$, тогда множество

$$K = \{x \in \mathbb{N} : !F(x, x)\}$$



является перечислимым и неразрешимым.

$f(x) = F(x,x)$

$K = \mathrm{dom}(f)$

Вопрос.

Что можно сказать про множество $\overline{K} = \mathbb{N} \setminus K$?

Вопрос.

Что можно сказать про множество $\overline{K} = \mathbb{N} \setminus K$?

Оно разрешимо?

$\overline{K}$ – разреш $\Rightarrow$ K – разр

Вопрос.

Что можно сказать про множество $\overline{K} = \mathbb{N} \setminus K$?

Оно разрешимо?

Нет

Перечислимо?

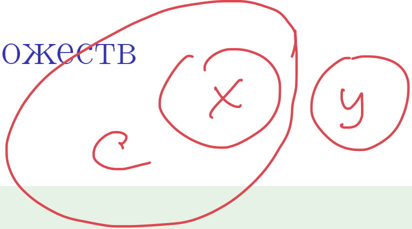
Нет

$\overline{K}$ – перечислимо $\xRightarrow{\text{т. Поста}}$ $\overline{K}$ разр. и K - разр.

Опр. A – наз. коперечислимым, если $\overline{A}$ – перечислимо

$\overline{K}$ – коперечислимо

Пара неотделимых перечислимых множеств



Пара множеств $X, Y \subseteq \mathbb{N}$ неотделима, если

- $X \cap Y = \varnothing$
- не существует разрешимого множества $C \subseteq \mathbb{N}$ такого,что $X \subseteq C$ и $Y \cap C = \varnothing$.

Теорема 25.1

Существует неотделимая пара перечислимых множеств.

Доказательство.

Пусть $f : \mathbb{N} \to \{0, 1\}$ — вычислимая функция без тотального вычислимого продолжения. Положим $X := \{x \in \mathbb{N} : f(x) = 0\}$ и $Y := \{x \in \mathbb{N} : f(x) = 1\}$.

$X \cap Y = \varnothing$

По теореме о графике X, Y перечислимы.

Если разрешимое C отделяет X и Y, то функция

$$g(x) := \begin{cases} 0, & \text{если } x \in C; \\ 1, & \text{иначе.} \end{cases}$$

$= 1 - \chi_C(x)$

продолжает f на всё $\mathbb{N}$.

Установленные факты

- Универсальная вычислимая функция $F(e, x)$.

Установленные факты

- Универсальная вычислимая функция $F(e,x)$.
- Частичная вычислимая $f:\mathbb{N}\to\{0,1\}$, не продолжаемая до тотальной вычислимой:

$$f(x) := \begin{cases} 1, & \text{если } F(x,x)=0; \\ 0, & \text{если } !F(x,x)\neq 0; \\ \text{неопр.}, & \text{иначе.} \end{cases}$$

Установленные факты

- Универсальная вычислимая функция $F(e, x)$.
- Частичная вычислимая $f : \mathbb{N} \to \{0, 1\}$, не продолжаемая до тотальной вычислимой:

$$f(x) := \begin{cases} 1, & \text{если } F(x, x) = 0; \\ 0, & \text{если } !F(x, x) \neq 0; \\ \text{неопр.}, & \text{иначе.} \end{cases}$$

- $K := \{x \in \mathbb{N} : !F(x, x)\}$ перечислимое, неразрешимое.

Пара неотделимых перечислимых множеств

Пара множеств $X, Y \subseteq \mathbb{N}$ неотделима, если

- $X \cap Y = \varnothing$
- не существует разрешимого множества $C \subseteq \mathbb{N}$ такого,что $X \subseteq C$ и $Y \cap C = \varnothing$.

Теорема 25.2

Существует неотделимая пара перечислимых множеств.

Доказательство. Пусть $f : \mathbb{N} \to \{0,1\}$ — вычислимая функция без тотального вычислимого продолжения. Положим $X :=\{x \in \mathbb{N} : f(x) = 0\}$ и $Y :=\{x \in \mathbb{N} : f(x) = 1\}$.

По теореме о графике X, Y перечислимы.

Если разрешимое C отделяет X и Y, то функция

$$g(x) := \begin{cases} 0, & \text{если } x \in C; \\ 1, & \text{иначе.} \end{cases}$$

продолжает f на всё $\mathbb{N}$.

Главные универсальные функции

Вычислимая универсальная функция $F : \mathbb{N}^2 \to \mathbb{N}$ называется главной, если для любой вычислимой $g : \mathbb{N}^2 \to \mathbb{N}$ найдётся тотальная вычислимая функция $s : \mathbb{N} \to \mathbb{N}$ такая, что

$$\forall e, x \; g(e, x) \simeq F(s(e), x).$$

Главные универсальные функции

Вычислимая универсальная функция $F : \mathbb{N}^2 \to \mathbb{N}$ называется главной, если для любой вычислимой $g : \mathbb{N}^2 \to \mathbb{N}$ найдётся тотальная вычислимая функция $s : \mathbb{N} \to \mathbb{N}$ такая, что

$$\forall e, x \; g(e, x) \simeq F(s(e), x).$$

Теорема 25.3

Главная вычислимая универсальная функция $F : \mathbb{N}^2 \to \mathbb{N}$ существует.

Главные универсальные функции

Вычислимая универсальная функция $F : \mathbb{N}^2 \to \mathbb{N}$ называется **главной**, если для любой вычислимой $g : \mathbb{N}^2 \to \mathbb{N}$ найдётся тотальная вычислимая функция $s : \mathbb{N} \to \mathbb{N}$ такая, что

$$\forall e, x \; g(e, x) \simeq F(\underline{\underline{s(e)}}, x).$$

Теорема 25.3

Главная вычислимая универсальная функция $F : \mathbb{N}^2 \to \mathbb{N}$ существует.

На самом деле, универсальная МТ задает главную унив. функцию.

M – вычисляет g s на вход даже $\underline{\underline{e}}$

Замечание

Вычислимую функцию $g(e, x)$ можно понимать как (возможно, не универсальный) язык программирования, где e — программа вычисления функции $x \mapsto g(e, x)$.

Функция s есть интерпретатор, сопоставляющий программе e языка g машину Тьюринга $s(e)$, вычисляющую ту же функцию.

(

Вариации Шелл

$F(x,y)$ – ун. выч ф-ция

$\downarrow$

$T(x,y,z)$ – ун выч ф-ия где $Com(\mathbb{N}^3\mathbb{N})$

$\downarrow$

F

Теорема Райса–Успенского

Какие свойства вычислимых функций распознаваемы по программе?

Примеры практически интересных свойств частичных функций f:

- $\forall x \, !f(x)$ (тотальность); +
- $f(x_0) = y_0$, где x_0, y_0 фиксированы; +
- $f = g_0$, где функция g_0 фиксирована; +
- «вычисление $f(x)$ на некотором x приводит к стиранию всех данных на HD компьютера». +

Пусть фиксирована универсальная вычислимая функция F. Обозначим через F_e частичную функцию с индексом e, т.е. $F_e(x) \simeq F(e, x)$.

Нетривиальным свойством вычислимых функций называем любое подмножество $\mathcal{C} \subset \mathrm{Com}(\mathbb{N}, \mathbb{N})$ такое, что $\mathcal{C} \neq \varnothing$ и $\mathcal{C} \neq \mathrm{Com}(\mathbb{N}, \mathbb{N})$.

$I_C = \phi$ $I_C = \mathbb{N}$

С каждым свойством $\mathcal{C}$ вычислимых функций связывается множество всех программ, вычисляющих функции со свойством $\mathcal{C}$, то есть множество $I_{\mathcal{C}} := \{e \in \mathbb{N} : F_e \in \mathcal{C}\}$.

Теорема 25.4

Если $\mathcal{C}$ — нетривиальное свойство вычислимых функций, то множество $\{e \in \mathbb{N} : F_e \in \mathcal{C}\}$ неразрешимо.

$I_c =$

Доказательство.

- Можно считать, что нигде не определённая функция ζ не обладает свойством $\mathcal{C}$ — иначе заменим $\mathcal{C}$ на его дополнение.
- Т.к. $\mathcal{C} \neq \varnothing$, фиксируем вычислимую функцию $f_0 \in \mathcal{C}$.

Доказательство.

- Можно считать, что нигде не определённая функция ζ не обладает свойством $\mathcal{C}$ — иначе заменим $\mathcal{C}$ на его дополнение.
- Т.к. $\mathcal{C} \neq \varnothing$, фиксируем вычислимую функцию $f_0 \in \mathcal{C}$.
- Построим тотальную вычислимую функцию $s : \mathbb{N} \to \mathbb{N}$ такую, что для всех $x \in \mathbb{N}$
$$x \in K \iff s(x) \in I_{\mathcal{C}}.$$
- Если бы $I_{\mathcal{C}} := \{e \in \mathbb{N} : F_e \in \mathcal{C}\}$ было разрешимо, то мы получили бы следующий разрешающий алгоритм для K: для данного x вычислить $y = s(x)$ и проверить $y \in I_{\mathcal{C}}$.

Вычисляем $g(e, x)$ в соответствии со следующим алгоритмом:

- вычислить $F_e(e)$;
- если $!F_e(e)$, очистить ленту, а затем вычислить $f_0(x)$. $\in$ C

По свойству главности получаем тотальную вычислимую функцию s такую, что

$$\forall e, x\ F_{s(e)}(x) \simeq g(e, x).$$

Тогда имеем:

- Если $e \in K$, то $F_{s(e)}(x) \simeq f_0(x)$; $\in \mathcal{C}$
- Если $e \notin K$, то $\boxed{F_{s(e)} = \zeta}$.

$s(e) \in I_{\mathcal{C}}$

$\zeta \notin \mathcal{C}$

$s(e) \notin I_{\mathcal{C}}$

Отсюда $e \in K \iff F_{s(e)} \in \mathcal{C} \iff s(e) \in I_{\mathcal{C}}$.

Следствие 25.5

Следующие свойства вычислимых функций не распознаваемы по программе:

- тотальность,
- ограниченность,
- конечность области определения, и т.д.

Замечание

Такие свойства как

- «вычисление $f(0)$ завершается менее, чем за 100 шагов»;
- «программа f содержит менее 100 символов» (при фиксированном алфавите)

являются разрешимыми свойствами программ. Они не соответствуют никакому классу частичных функций.

Следствие

$f \in Com(\mathbb{N}, \mathbb{N})$ $\quad \{e \mid F_e(x) \simeq f(x)\}$

— неразр. и след. бескон.

$\boxed{F(e,x) \text{ — ун. выч. ф-ция}}$

$\{e \mid F_e$ — нигде неопр. ф-ция$\} = \{0\}$

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: $A \leqslant_m B$.

Св-ва. 1) если B — разрешимо $\Rightarrow$ A — разрешимо

2) если B — перечислимо $\Rightarrow$ A — перечислимо

3) $\leqslant_m$ — рефлекс. и транзитивно

$y \in B \Leftrightarrow g(y) \in C$

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

$x \in A \Leftrightarrow g(f(x)) \in C$

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:

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

$\mathbb{N}\setminus A$ $\quad$ $\mathbb{N}\setminus B$

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

4) A – разрешима и $B \neq \varnothing, \mathbb{N}$, то

$A \leq_m B$

5) $A \leq_m B \Rightarrow \mathbb{N} \setminus A \leq_m \mathbb{N} \setminus B$

тоже f.

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

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

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

Кодирование Q и Σ

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

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

Слово $x \in \Sigma^*$ кодируется конкатенацией $Code(x)$ кодов всех его букв, например $Code(a_2 a_0 a_1) = a\mathbf{11}aa\mathbf{1}$.

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 П.

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 Π^*.

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 Δ^*, 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_Δ определена для любой машины M с рабочим алфавитом $\Sigma \supset \Delta$.

Утверждение

Для любой МТ M и Δ можно указать машину M' вычисляющую функцию M_Δ 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_Δ 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_Δ начинает работу в конфигурации $q_1\#Code(M)\$x$ для $x \in \Delta^*$, то заключительная конфигурация $q_0\#M_\Delta(x)$;
- Иначе U_Δ зацикливается.

Алгоритм работы машины U_Δ:

- 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 $K = \{x \,|\, e \mid F(e,x)$ 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, $+$, $\cdot$, 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: $a \leq 0 \lor 0 \leq a$ поскольку $0 \leq a$.

Шаг:

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