Теория алгоритмов

Лекция №3

Конспект Вячеслава Копейцева, КН-301
Читает Юрий Окуловский

28.09.2015

На прошлой лекции было закончено обсуждение машины Тьюринга (далее МТ) и дано определение машины Минского (далее ММ)

1 Вычислительная мощность ММ

Докажем, что ММ по вычислительным возможностям равномощна МТ т.е.:

1. $MM \subseteq MT$
2. $MT \subset MM$

Доказательство:

1. Возьмём МТ с $\Sigma\{1, 0\}$
2. Договоримся, что δ полностью определена

$q_i, 0 \rightarrow q_{ni}, s_{ni}, p_i^2$
$q_i, 1 \rightarrow q_{yi}, s_{yi}, p_i^3$
Лента МТ:

$$\begin{array}{l} \qquad\qquad\quad \nabla \\ \underbrace{1000}_{\text{R1}}\,\underbrace{1}_{\text{R2}}\,\underbrace{10}_{\text{R3}} \leftarrow \text{Регистры} \\ \qquad\qquad\quad \uparrow \end{array}$$

Число, записанное в обратном порядке
ММ:
$p_i : R_2 + +; p_i'$
$p_i' : R_2 - -, p_i^2 : p_i^3$ - в зависимости от сравнения с 0
Рассмотрим состояния p_i^2:
Нужно напечатать что-то на ленте
p_i^2: если $S_{ni} = 0$, то ничего;
$S_{ni} = 1$, то R_2++;

В самой ММ условного перехода нет, т.к. всё уже определено
если $t_{ni} = \bullet$, то ничего;
если $t_{ni} = \leftarrow$, то: $R_3 = 2R_3 + R_2$
$R_2 = R_1\%2$
$R_1 = R_1/2$

2 Тезис Чёрча-Тьюринга

1. Любое мыслимое на сегоднящний день определение алгоритма будет эквивалентно МТ.

2. Всё, что мы можем вычислить, мы можем вычислить на МТ.

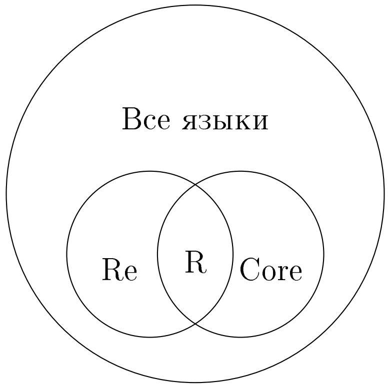
3 Теория вычислимости

3.1 Определения

Опр. L - язык вычислимый (decidable) или разрешимый, если $\exists$MT A, которая его распознаёт т.е.:
A : $\forall$ w $\in$ L A(w) = 1
$\forall$ w $\notin$ L A(w) = 0
R - счётное число
Опр. Язык L - рекурсивно-перечислимый (semidecidable), если $\exists$MT A такая что:
$\forall$ w $\in$ L A(w) = 1
$\forall$ w $\notin$ L A(w) = MT ничего не говорит.
Re - счётное число
R $\subset$ Re
Всего MT - счётное число, т.к. текст имеет конечную длину т.е. |R| и |Re| - счётно.
L $\subseteq$ Σ^*
$|\Sigma^*|$ = континуум $\Rightarrow$ $\exists$ L $\notin$ Re т.е. не распознаваемый MT.
Опр. Язык L - ко-рекурсивно-перечислимый (CoRe), если $\exists$MT A такая что:
$\forall$ w $\notin$ L A(w) = 1
$\forall$ w $\in$ L A(w) = машина зацикливается.
Утв. R = Re $\cap$ Core.
Доказательство:
$\subseteq$:
L $\in$ R $\Rightarrow$ $\exists$ MT A, которую можно разделить на две MT:

1. A_1 : $\forall$ w $\in$ L A(w) = 1
   $\forall$ w $\notin$ L A(w) = машина зацикливается.

   L $\in$ Re



2. A_0 : ∀ w ∈ L A(w) = машина зацикливается.

   ∀ w ∉ L A(w) = 1

   L ∈ CoRe

⊇:

L ∈ Re ∩ CoRe $\Rightarrow L \in Re, L \in CoRe$

Запустим параллельно:

$$\left.\begin{array}{l} A_1 \\ A_0 \end{array}\right\} \text{A}$$

Утв. R замкнут относительно ∪, ∩, ¬ т.е.:

$\forall L_1, L_2 \in R : L_1 \cap L_2 \in R, L_1 \cup L_2 \in R, \Sigma^* \setminus L_1 \in R$

Доказательство:

Запустить обе МТ и ждать результата их работы т.к. они обязательно должны остановиться.

Утв. Re замкнут относительно ∪, ∩ т.е.:

$\forall L_1, L_2 \in Re : L_1 \cap L_2 \in Re, L_1 \cup L_2 \in Re$

Доказательство: Параллельно запускаем две МТ. В случае пересечения мы гарантировано за конечное время дождёмся остановки двух МТ, в случае объединения - одной.

3.2 Теорема "Проблема останова" (Halting problem)

Определить, останавливается ли МТ A на входных данных X?

$L_H = \{''(A, X)'' :$ A останавливается на X$\}$

$L_H \in Re$

$w \in L_H$

$w = (A, X)$

Покажем, что $L_H \notin R$.

От противного:

$$\exists MTM(A, X) = \{ \begin{array}{l} \text{1, если A останавливается на X} \\ \text{0, если A не останавливается на X} \end{array}$$

Машина $M_1 : M_1(A) = M(A, A)$

$$M_2(A) = \{ \begin{array}{l} \text{зацикливается, если } M_1(A) = 1 \\ 1, \text{ если } M_1(A) = 0 \end{array}$$

Рассмотрим чему равно $M_2(M_2)$:

Пусть $M_2(M_2)$ останавливается $\Rightarrow M(M_2, M_2) = 1 \Rightarrow M_1(M_2) = 1 \Rightarrow M_2(M_2)$ зациклится.

Если $M_2(M_2)$ зациклится $\Rightarrow M(M_2, M_2) = 0 \Rightarrow M_1(M_2) = 0 \Rightarrow M_2(M_2) = 1$ т.е. машина не зациклится.

Противоречие $\Rightarrow \nexists M$.

Пример:

Рассмотрим код:

```
for x = 2,3...
    y = 2,3...
    n = 3,4...
    z = 2,3...
    if (x^n + y^n == z^n)
        stop
```

С помощью такой МТ можно было бы доказать великую еорему Ферма, как и любую другую теорему из области конструктивной математики, но, к сожалению, очевидно, что такую МТ построить нельзя.