Лекции по Дискретной математике 4 семестр

Ilya Yaroshevskiy

13 мая 2023 г.

Оглавление

Лекция 1

1.1 Производящие функции

Последовательность $a_0, a_1, a_2, \ldots$. Запишем в виде ряда

$$A(t) = a_0 + a_1 t + a_2 t^2 + \ldots$$

, где $A(t)$ — **производящая функция**

Свойство 1.

- $A(t) = a_0 + a_1 t + a_2 t + \ldots$
- $B(t) = b_0 + b_1 t + b_2 t + \ldots$

$$A(t) + B(t) = C(t)$$
$$c_n = a_n + b_n$$
$$C(t) = c_0 + c_1 t + c_2 t^2 + \ldots$$

Свойство 2.

- $A(t) = a_0 + a_1 t + a_2 t + \ldots$
- $B(t) = b_0 + b_1 t + b_2 t + \ldots$

$$A(t) \cdot B(t) = C(t)$$
$$(a_0 + a_1 t + a_2 t^2 + \ldots) \cdot (b_0 + b_1 t + b_2 t^2) =$$
$$= a_0 b_0 + (a_0 b_1 + a_1 b_0) t + (a_0 b_2 + a_1 b_1 + a_2 b_0) t^2 + \ldots$$
$$c_n = \sum_{k=0}^{n} a_k b_{n-k}$$
$$C(t) = c_0 + c_1 t + c_2 t^2 + \ldots$$

Свойство 3.

- $A(t) = a_0 + a_1 t + a_2 t + \ldots$
- $B(t) = b_0 + b_1 t + b_2 t + \ldots$, $b_0 \neq 0$

$$\frac{A(t)}{B(t)} = C(t)$$

$$C(t) \cdot B(t) = A(t)$$

$$c_n = a_n - \sum_{k=0}^{n-1} c_k b_{n-k}$$

Если $b_0 = 1$ *и* $a_i, b_i \in \mathbb{Z}$, *то* $c_i \in \mathbb{Z}$

Свойство 4.

- $A(t) = a_0 + a_1 t + a_2 t + \dots$

$$A'(t) = a_1 + 2 \cdot a_2 t + 3 \cdot a_3 t^2 + \dots$$

$$a'_n = n \cdot a_n t^{n-1}$$

Свойство 5.

- $A(t) = a_0 + a_1 t + a_2 t + \dots$

$$\int A(t) = a_0 t + \frac{1}{2} a_1 t^2 + \frac{1}{3} a_2 t^3 + \dots$$

$$a'_n = \frac{1}{n+1} \cdot a_n t^{n+1}$$

Свойство 6.

- $A(t) = a_0 + a_1 t + a_2 t + \dots$
- $B(t) = b_0 + b_1 t + b_2 t + \dots$, $b_0 = 0$

$$C(t) = A(B(t))$$

$$c_n = \sum_{k=1}^{n} a_i \sum_{n=k_1+k_2+\dots+k_i} \prod_{j=1}^{i} b_{k_j}$$

Лекция 2

2.1 Производящие функции

Определение. Полином — степенныой ряд, у которого начиная с некоторого места n все коэффиценты 0.

Обозначение. $\deg p = n$

Определение. $\frac{P(t)}{Q(t)}$ — **дробно рациональная функция**

2.1.1 Рекурентные соотношения

Определение.

$$m : a_0, a_1, \ldots, a_{m-1}$$

$k \leq m, n \geq m$

$$a_n = c_1 a_{n-1} + \cdots + a_k a_{n-k}$$

, где $c_1, \ldots, c_k$ — коэффиценты рекурентности

Пример.

- $m = 2,\ k = 2$
- $f_0 = f_1 = 1$
- $c_1 = c_2 = 1$

$\mathrm{f_n = f_{n\text{ - }1} + f_{n\text{ - }2}}$ — числа Фибоначи

Определение. Квазиполином

$$f(n) = \sum_{i=1}^{k} p_i(n) r_i^n$$

, где p_i — полином, r_i — числа

Теорема 2.1.1.

- $a_0, a_1, \ldots, a_n, \ldots$

Тогда эквивалентны:

1. $A(t) = \frac{P(t)}{Q(t)}$, P, Q — полиномы, $q_0 \neq 0$
2. для $n \geq m$ a_n задается линейным рекурентным соотношением: $a_n = c_1 a_{n-1} + \cdots + c_k a_{n-k}$, причем:

- $Q(t) = 1 - c_1 t - c_2 t^2 - \cdots - c_k t^k$
- $\deg P \leq m - 1$

3. a_n — квазиполином

$$a_n = \sum_{i=1}^{k} p_i(n) r_i^n \tag{2.1}$$

причем:

- r_i — обратные величины корням $Q(t)$
- k — число различных его корней
- $\deg p_i =$ (кратность корня(r_i^{-1})) $- 1$
(2.1 кроме $\leq m$ первых членов)

2.1.2 Рекурента в рациональную ПФ

$$A(t) = \frac{P(t)}{Q(t)}$$

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_k a_{n-k}$$

$$m = \deg P + 1 \quad k = \deg Q$$

$$p_i = a_i - \sum_{j=1}^{\min(k,i)} a_{i-j} c_j$$

$$a_n = \frac{p_n - \sum_{i=1}^{n} a_{n-i} q_i}{q_0}$$

$$c_i = -q_i$$

$$a_n = \sum_{i=1}^{\min(n,k)} c_i a_{n-i} [+p \text{ если } n < m]$$

Лекция 3

3.1 Ассимпточеское поведение линейных рекуррент

3.1.1 Квазимногчлен в рациональную ПФ

Лемма 1.

- $a_n = n^k r^n$
- $A(t) = \frac{P(t)}{Q(t)}$

Тогда $Q(t) = (1 - rt)^{k+1}$

$$A(t) = \frac{1}{r} \cdot \frac{P_k'(t)(1-rt) + r(k+1)P_k(t) - \sum_{i=0}^{k} r\binom{k+1}{i} P_i(t)(1-rt)^{k-i+1}}{(1-rt)^{k+2}}$$

Доказательство. Доделать $\square$

Лемма 2.

- $a_n = p(n)r^n$
- $A(t) = \frac{P(t)}{Q(t)}$

Тогда $Q(t) = (1 - rt)^{\deg p + 1}$

Следствие 3.1.0.1. Квазимногочлен $\Rightarrow$ Рациональная ПФ:
Корни $Q(t)$: $\frac{1}{r_i}$ кратности $\deg p_i + 1$

3.1.2 Рациональная ПФ в квазимногочлен

- $A(t) = \frac{P(t)}{Q(t)}$

$$Q(t) = \prod_{i=1}^{s} (1 - r_i t)^{f_i}$$

$$\frac{P(t)}{Q(t)} = \sum_{i=1}^{s} \frac{P_i(t)}{(1 - r_i t)^{f_i}}$$

Лемма 3.

$$A(t) = \frac{P(t)}{(1-rt)^{k+1}}$$

Тогда

$$a_n = p(n)r^n$$

, *p — полином,* $\deg p = k$

$$A(t) = P(t)U(t)$$
$$U(t) = (1 + rt + r^2t^2 + \dots)^{k+1}$$
$$a_n = \sum_{i=0}^{n} p_i u_{n-i}$$

Следствие 3.1.0.2.

$$a_n = \sum_{x_1+x_2+\dots+x_{k+1}=n} r^n = \binom{n+1+k-1}{k} r^n = \binom{n+k}{k} r^n =$$
$$= \frac{1}{k!}(n+k)(n+k-1)\dots(n+1)r^n = p_k(n)r^n$$
$$\sum_{i=0}^{m} p_i u_i = \left(\sum_{i=1}^{m} \frac{p_{n-i}(n)}{r^i}\right) r^n$$

3.1.3 Оценка ассимптотического поведения

Обратные корни: $\begin{matrix} r_1 & f_1 \\ r_2 & f_2 \\ \vdots & \vdots \\ r_s & f_s \end{matrix}$

Свойство 1.

- $\exists r_i : |r_i|$ — *max*
- $\forall j \neq i :\ |r_j| < |r_i|$

r_i *вещественные* $a_n \sim n^{f_i-1} \cdot r_i^n$

Свойство 2. *Несколько r_i имеют* $\max |r_i|$

1. $r_i \in \mathbb{R}$, $r_i = \pm r$. *Если разной кратности у r_i, r_j, соответсвенно $f_i > f_j$*
   Тогда $a_n \sim n^{f_i-1}r_i^n(+n^{f_i-1}r_j^n)$
   Если одинаковой кратности $f_i = f_j$
   Тогда $a_n \sim c_1 n^{f_i-1}r^n + c_2 n^{f_j-1}(-r)^n$

Свойство 3. $r_1, r_2, \dots, r_l$ — *орбратные корни максимальной степени* $\max |r_i|$ *и* $\max f_i$

$$r_i = z_i e^{i\phi_i}$$
$$a_n \sim n^{f_i} z^n \sum_{j=1}^{l} e^{i\phi_j}$$

Если $\phi_j = \frac{2\pi a_j}{b_j}$, *$n$ делится на* $\mathrm{LCM}(b_j)$ *классов*

Пример. Числа каталана:

$$c_n = \sum_{k=0}^{n-1} c_k c_{n-k-1}$$

$$C(t) = c_0 + c_1 t + c_2 t^2 + \dots$$

$$C(t)^2 = c_0^2 + (c_0 c_1 + c_1 c_0)t + \dots$$

$$C(t)^2 \cdot t + 1 = C(t)$$

$$t \cdot C(t)^2 - C(t) + 1 = 0$$

$$C(t) = \frac{1 - \sqrt{1-4t}}{2t}$$

Примечание. Рассмотрим $(1-t)^\alpha$:

$$(1-t)^\alpha = \sum_{i=0}^{\infty} \binom{\alpha}{i} t^i = P_\alpha(t)$$

$$\binom{\alpha}{t} = \frac{\alpha(\alpha-1)(\alpha-2)\cdot(\alpha-i+1)}{1\cdot 2\cdot 3\cdot\dots\cdot i}$$

$$P_{\frac{1}{2}}(-4t) = 1 - 2t - 2t^2 - 4t^3 - 10t^4$$

$$\binom{\frac{1}{2}}{2} = \frac{\frac{1}{2}\cdot\left(-\frac{1}{2}\right)}{1\cdot 2} = -\frac{1}{8}$$

$$\binom{\frac{1}{2}}{3} = \frac{\frac{1}{2}\cdot\left(-\frac{1}{2}\right)\left(-\frac{3}{2}\right)}{1\cdot 2\cdot 3} = \frac{1}{16}$$

Лекция 4

4.1 Производящие функции для объектов

- Оюъединение
  A, B $A \cap B = \emptyset$ $C = A \cup B$
  $A(t)$ $B(t)$
  $$C(t) = A(t) + B(t)$$
  $$c_n = a_n + b_n$$

- Пара
  $C = A \times B$ $\text{Pair}(A, B)$
  $$C(t) = A(t) \cdot B(t)$$
  $$c_n = \sum_{i=0}^{n} a_n b_n$$

- Последовательности
  $C = \text{Seq } A = A^0 \cup A^1 \cup A^2 \cup A^3 \cup \dots$ $a_0 = 0$
  $$C(t) = 1 + A(t) + A(t) \cdot A(t) + A(t)^3 + \dots$$
  $$C(t) = \frac{1}{1 - A(t)}$$

- Множества
  ε вес 0
  Set $A = \bigtimes_{a \in A} (\varepsilon \cup a)$
  $$C(t) = \prod_{a \in A} (1 + t^{\omega(a)}) = \prod_{k=0}^{\infty} (1 + t^k)^{a_k}$$
  Пример. Set $\{\square, \boxminus\}$ $a_1 = 1, a_2 = 1$
  $$C(t) = (1 + t)(1 + t^2) = t^3 + t^2 + t + 1$$

- Мультимножества
  $$\text{MSet} A = \bigtimes_{a \in A} (\varepsilon \cup a \cup a^2 \cup \dots) = \prod_{a \in A} \text{Seq}\{a\}$$

$$C(t) = \prod_{a \in A} \frac{1}{1 - t^{\omega a}} = \prod_{k=1}^{\infty} \left(\frac{1}{1 - t^k} \right)^{a_k} = \prod_{k=1}^{\infty} (1 - t^k)^{-a_k}$$

Пример. MSet{▯,⊟}

$$C(t) = \frac{1}{(1-t)(1-t^2)} = \frac{1}{(1-t^2)(1+t)}$$

$$c_n = dn + e + f \cdot (-1)^n$$

Пример. $\mathrm{Seq}_{=k}(A) = A^k$ — ровно 3 элемента
$\mathrm{Seq}_{\geq k}(A) = A^k \times \mathrm{Seq}(A) \frac{A(t)^k}{1-A(t)}$
$\mathrm{Seq}_{\leq k}(A) = \frac{1}{1-A(t)} - \frac{A(t)^{k+1}}{1-A(t)} = \frac{1-A(t)^{k+1}}{1-A(t)}$

Лекция 5

5.1 Производящие функции для регулярных языков

L — регулярный язык

$$|L \cap \Sigma^n| = a_n$$

$$L(t) = a_0 + a_1 t + \dots$$

Примечание. L — регулярная спецификация
ψ — регулярное выражение:

1. $L(\psi) = L$
2. $\forall x \in Ł\ \exists!$ способ x удовлетворяющий ψ

Лемма 4. *Σ — конечный алфавит, $L \subset \Sigma^*$*
L — регулярная спецификация $\Leftrightarrow$ L получаетя из Σ:

1. *Дизъюнктное объединение $+$*
2. *Прямое произведение $\times$*
3. *Последовательность Seq*

Доказательство. Общее рассждение: по индукции рассмотрим для каждой операции во что она перейдет, надо показать что единственность вывода сохраняется Не работает $\square$

Пример.

$$ab^*|a^*b$$

$$a \times \operatorname{Seq} b | \operatorname{Seq} a \times b$$

объединение дизъюнктное? $\Rightarrow$ не регелярная спецификация

Пример.

$$(ab^*)^*$$

$$\operatorname{Seq}(a \times \operatorname{Seq} b)$$

Теорема 5.1.1. Если у L есть регулярная спецификация, то L — дробно рациональная

Теорема 5.1.2 (Производящая функция регулярного языка)**.** L — регулярный язык над Σ, ДКА A:

- Состояния $Q,\ |Q| = n$
- $s \in Q$ — стартовое сотояние
- $T \subset Q$ — терминальные

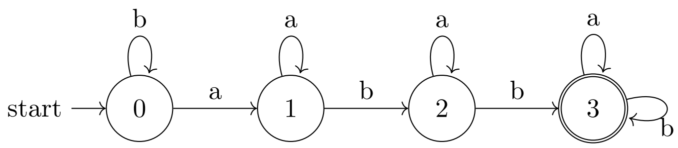
$$u = (\overbrace{0, 0, \dots, \underbrace{1}_{s}, 0, \dots, 0}^{n})$$

$$v = (\overbrace{0, \underbrace{1}_{\in T}, 0, \underbrace{1}_{\in T}, \dots, \underbrace{1}_{\in T}, 0}^{n})$$

$$D = (d_{ij})^T,\ d_{ij} = |\{c | i \xrightarrow{c} j\}|$$

$$L(t) = \vec{u}(I - tD)^{-1}\vec{v}$$

Пример. Язык из слов, которые содержат abb как подстроку



$$\begin{pmatrix} L_0 \\ L_1 \\ L_2 \\ L_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} + t \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} L_0 \\ L_1 \\ L_2 \\ L_3 \end{pmatrix}$$

$$L_0 = \frac{t^3}{(1-t)(1-2t)(1-t-t^2)}$$

5.2 Автомат КМП и автокор. многочлен

Конструкция Гуибаса-Одлызко

$$p = \boxed{\mathrm{p}_1, p_2, \dots, p_k}$$

$$c_i = [p[i+1\dots k] = p[1\dots k-i]]$$

$$c(t) = c_0 + c_1 t + c_2 t^2 + \dots + c_{k-1} t^{k-1}$$

Пример. $p = \mathrm{aabbaa}$
$c = (1, 0, 0, 0, 1, 1)$
$c(t) = 1 + t^4 + t^5$

Теорема 5.2.1.

- $\Sigma,\ |\Sigma| = m$

S_n — количество слов длины n, не содержащих p

$$S(t) = s_0 + s_1 t + s_2 t^2 + \dots$$

$$S(t) = \frac{c(t)}{t^k + (1-mt)c(t)}$$

Пример. $p = \text{abb}$

$$c(t) = 1$$

$$\frac{1}{t^3 + (1-2t)\cdot 1} = \frac{1}{1 - 2t + t^3}$$

5.2.1 Пентагональная формула Эйлера

$$p_0 \; p_1 \; p_2 \; \dots \; p_n \; \dots$$

p_n — количество разбиений n на слагаемые из $\mathbb{N}$. Порядок не важен

- $U = \{0\},\ u_1 = 1,\ U(t) = t$
- $N = \text{Seq}^{+}U$ =положительно целые числа
- $P = \text{MSet}\ N$

$$P(t) = \prod_{k=1}^{\infty} \frac{1}{1 - t^k}$$

$$Q(t) = \prod_{k=1}^{\infty} (1 - t^k)$$

$$R(t) = \prod_{k=1}^{\infty} (1 + t^k)\ [t^n]R \to r_n$$

r_n — количество разбиений на различные слагаемые

$$[t^n]Q = \sum_{\substack{\text{разбиение } n \text{ на} \\ \text{различные слагаемые}}} (-1)^{\text{число слагаемых}}$$

$$q_n = e_n - o_n$$

e_n — число разбиений на четное число различных слагаемы, o_n — число разбиений на нечетное число различных слагаемы,

Теорема 5.2.2.

$$Q(t) = 1 + \sum_{k=1}^{\infty} (-1)^k (t^{\frac{3k^2-k}{2}} + t^{\frac{3k^2+k}{2}})$$

Лемма 5.

$$n \neq \frac{ek^2 \pm k}{2}, \textit{то } e_n = o_n$$

$$n = \frac{ek^2 \pm k}{2}, \textit{то } e_n = o_n + (-1)^k$$

Лекция 6

6.1 Помеченные КО и экспоненциальные производящие функции

$$a_0\ a_1\ a_2\ \dots\ a_n\ \dots \quad A(t) = a_0 + a_1 t + a_2 t^2 + \dots + a_n t^n + \dots$$

Определение. Экспоненциальная производящая функция:

$$a(t) = \frac{a_0}{0!} + \frac{a_1}{1!} \cdot t + \frac{a_2}{2!} \cdot t^2 + \dots + \frac{a_n}{n!} \cdot t^n + \dots$$

Обозначение. Мы будет обозначать ЭПФ так-же большой буквой

Пример. $1, 1, 1, 1, , 1, 1, 1, 1, 1$

ОПФ $\frac{1}{1-t}$

ЭПФ $1 + 1 \cdot t + \frac{1}{2!} \cdot t^2 + \frac{1}{3!} \cdot t^3 + \dots = \sum\limits_{n=0}^{+\infty} \frac{1}{n!} \cdot t^n = e^t = \exp(t)$

Пример. $1, 1, 2, 6, 24, \dots, n!, \dots \quad a_n = n!$

ОПФ $1 + t + 2 \cdot t^2 + 6 \cdot t^3 + \dots + n! \cdot t^n + \dots$

ЭПФ $\sum\limits_{n=0}^{+\infty} \frac{n!}{n!} \cdot t^n = \frac{1}{1-t}$

$$A(t) = \frac{a_0}{0!} + \frac{a_1}{1!} \cdot t + \frac{a_2}{2!} \cdot t^2 + \dots + \frac{a_n}{n!} \cdot t^n + \dots$$
$$B(t) = \frac{b_0}{0!} + \frac{b_1}{1!} \cdot t + \frac{b_2}{2!} \cdot t^2 + \dots + \frac{b_n}{n!} \cdot t^n + \dots$$

Свойство 4.

$$C(t) = A(t) \pm B(t) \quad c_n = a_n \pm b_n$$

Свойство 5.

$$C(t) = a(t) \cdot B(t)$$
$$\frac{C_n}{n!} = \sum_{k=0}^{n} \frac{a_k}{k!} \cdot \frac{b_{n-k}}{(n-k)!}$$
$$c_n = \sum_{k=0}^{n} \binom{n}{k} a_k b_{n-k}$$

Свойство 6.

$$C(t) = \frac{A(t)}{B(t)}$$

$$a_n = \sum_{k=0}^{n} \binom{n}{k} b_k c_{n-k} = \sum_{k=1}^{n} \binom{n}{k} b_k c_{n-k} + b_0 c_n$$

$$c_n = \frac{a - \sum_{k=1}^{n} \binom{n}{k} b_k c_{n-k}}{b_0}$$

Далее все производящие функции — экспоненциальные, а объекты помеченые

6.1.1 Помеченные объекты

Пример. Перестановк. $P_n = n!$ — количество перестановок из n элементов

Пример. Пустые графы. $E_n = 1$ — количество графов с n вершинами
ЭПФ: $\exp(t)$

Пример. Циклы. $C_n = (n-1)!$ — количество циклов из n вершин. Направление обхода фиксировано.
ЭПФ: $\sum\limits_{n=1}^{+\infty} \frac{n!}{n} \cdot \frac{1}{n!} \cdot t^n = \sum\limits_{n=1}^{+\infty} \frac{t^n}{n} = \ln \frac{1}{1-t}$

6.1.2 Операции

1. Дизъюнктное объединение (сумма)
   - A
   - B
   - $A \cap B = \emptyset$
   - $C = A \cup B$

   $$c_n = a_n + b_n \quad C(t) = A(t) + B(t)$$

2. Пара (произведение)
   - A
   - B
   - $C = A \times B$

   $$C = \{\langle \underbrace{a}_{k \text{ атомов}}, \underbrace{b}_{n-k \text{ атомов}} \rangle\}$$

   Получим последовательность $c_1 c_2 \dots c_n$. Перенумеруем элементы:
   Первые k в $d_1 d_2 \dots d_k$, где $d_i = |\{c_j | 1 \le j \le k,\ c_k \le c_i\}|$.
   А остальные $c_{k+1} \dots c_n$ в $e_1 \dots e_{n-k}$, где $e_i = |\{c_j | k+1 \le j \le n,\ c_j \le c_{i+k}\}|$.
   Пусть $d_i = a_i$, а $e_i = b_i$

   $$c_n = \sum_{k=0}^{n} \binom{n}{k} a_k b_{n-k} \quad C(t) = A(t) \cdot B(t)$$

Пример. Пары перестановок. $C(t) = \frac{1}{(1-t)^2}$. Тогда $c_n = (n+1)n!$

3. Последовательность

$$C = \operatorname{Seq} A = \emptyset + A \times \operatorname{Seq} A$$

$$C(t) = 1 + A(t) \cdot C(t)$$

$$C(t) = \frac{1}{1 - A(t)}$$

Пример.

- $U = \{\circ\}$
- $U(t) = t$
- $\operatorname{Seq} U = P$

$$P(t) = \frac{1}{1-t}$$

4. Множества (Set)

- $\operatorname{Set}_k A$ — множества, содержащие k обхектов

$$B_k = \operatorname{Seq}_k A = \underbrace{A \times A \times \dots \times A}_{k} \quad B_k(t) = A(t)^k$$

$$\operatorname{Set}_k A = \operatorname{Seq}_k A/_\sim$$

$[x_1 x_2 \dots x_k] \sim [y_1 y_2 \dots y_k]$. $\exists$ перестановка $\pi : x_i = y_{\pi[i]}$

$$C_k(t) = \frac{1}{k!} \quad B_k(t) = \frac{A(t)^k}{k!}$$

$$\operatorname{Set} A = \bigcup_{k=0}^{\infty} \operatorname{Set}_k A = \sum_{k=0}^{\infty} \frac{A(t)^k}{k!} = e^{A(t)}$$

Пример.

- $U = \{\circ\}$
- $U(t) = t$

$$\operatorname{Set} U = E \quad E(t) = e^t$$

, где E — пустые графы

Пример. Циклы.

- $U = \{\circ\}$
- $U(t) = t$
- $B = \operatorname{Set} \operatorname{Cyc} U$

$$B(t) = e^{C(t)} = e^{\ln \frac{1}{1-t}} = \frac{1}{1-t}$$

Набор помеченных циклов являеся престановкой

5. Циклы

    - $\mathrm{Cyc}_k A$ — количество циклов длины k

$$C = \mathrm{Cyc}_k A = \mathrm{Seq}_k A/_\sim$$

, где классы эквивалентности с точностью до циклических сдвигов.
$[x_1 \dots x_k] \sim [y_1 \dots y_k]$. $\exists i:\ x_j = y_{(i+j) \mod k+1}$

$$\mathrm{Cyc}\, U = \ln \frac{1}{1-t}$$

$$C_k(t) = \frac{1}{k} A(t)^k$$

$$C(t) = \sum_{k=1}^{\infty} \frac{1}{k} A(t)^k = \ln \frac{1}{1-A(t)}$$

$$\mathrm{Set}\,\mathrm{Cyc}\, U = P$$

$$\mathrm{Set}\,\mathrm{Cyc}\, A \simeq \mathrm{Seq}\, A$$

6.1.3 Обобщение

Теорема 6.1.1 (о подстановке)**.**

- A — помеченные КО — $A(t)$
- B — помеченные КО — $B(t)$

$C = A[B]$ — вместо каждого атома A подставляем КО B, перенумеруем получившиеся атомы произвольным образом

$$C(t) = A(B(t))$$

Пример. $A \times A$ — пара атомов. Их две $B(t) = t^2 = 2 \cdot \frac{1}{2!} \cdot t^2$. Подставляем $B(A(t)) = A(t)^2$

Лекция 7

Рассмотрим деревья:

$$T = t \times Seq\, T$$

, где t — корень

$$A(t) = t \cdot \phi(A(t))$$
$$\phi(s) = \frac{1}{1-s}$$

Решить это уравнение в общем виде можно с помощью формулы Лагранжа

Теорема 7.0.1 (формула обращения Лагранжа)**.**

$$a_n = \frac{1}{n} \cdot [s^{n-1}](\phi(s))^n$$

, где $[s^n]A(s)$ — коэффицент при s^n в $A(s)$

$$A(t) = \sum_{n=0}^{\infty} a_n t^n$$

Пример. Применим ее для деревьев

$$a_n = \frac{1}{n} \cdot [s^{n-1}] \left(\frac{1}{1-s}\right)^n$$
$$\left(\frac{1}{1-s}\right)^n = (1 + s + s^2 + s^3 + \dots + s^k + \dots)^n$$
$$(1-s)^{-n} = 1 - \binom{-n}{1} s + \binom{-n}{2} s^2 - \binom{-n}{3} s^3$$
$$\binom{-n}{n-1} = \frac{-n \cdot (-n-1) \cdot (-n-2) \cdot \dots \cdot (-n-(n-1)+1)}{1 \cdot 2 \cdot 3 \cdot \dots \cdot (n-1)}$$
$$\frac{1}{n}(-1)^{n-1}\binom{-n}{n-1} = \frac{(2n-2)!}{(n-1)!n!} = \binom{2n-2}{n-1}$$

Пример.

$$\phi(s) = e^s$$
$$\frac{a_n}{n!} = \frac{1}{n} \cdot [s^{n-1}] e^{ns}$$
$$e^{ns} = 1 + \frac{1}{1!}(ns) + \frac{1}{2!}(ns)^2 + \frac{1}{3!}(ns)^3 + \dots$$
$$[s^{n-1}] e^{ns} = \frac{n^{n-1}}{(n-1)!}$$

7.1 Производящая функция от нескольких перменных

$\binom{n}{k}$ образуют таблицу:

$n \backslash k$					
	1				
	1	1			
	1	2	1		
	1	3	3	1	
	1	4	6	4	1

$$A_k(t) = \sum_{n=0}^{\infty} \binom{n}{k} t^n$$

$$B_n(t) = \sum_{n=0}^{\infty} \binom{n}{l} t^k$$

$$C(u, z) = \sum_{n,k} \binom{n}{k} z^n u^k = \frac{1}{1 - z - uz}$$

Посмотрим на $C(u, z)$ так: n — вес, k — стоимость. Будем считать, что z — не берем объект, uz — берем объект

$$\mathrm{Seq}\,\{z, uz\} = [], [z], [uz], [z, z], [z, uz], [uz, z], [uz, uz], \ldots$$

$$A(u, z) = z + uz$$

7.1.1 Числа Стирлинга I рода

Исправить

Помеченные перстановки, Set Cyc Z

$$\bigcup_{k=0}^{\infty} \mathrm{Set}_{=k} \mathrm{Cyc}\, Z$$

$$\mathrm{Set}\,\mathrm{Cyc}\, Z = e^{\ln \frac{1}{1-Z}} = \frac{1}{1 - Z}$$

$$\bigcup_{k=0}^{\infty} \mathrm{Set}_{=k} \left(u \times \mathrm{Cyc}\, Z\right) \mapsto \sum_{n,k} \begin{bmatrix} n \\ k \end{bmatrix} \frac{1}{n!} z^n u^k$$

$$\mathrm{Set}_{=k}(A) =$$

k! = A(Z)^k $\overline{k!}$

$$u \times \mathrm{Cyc}\, Z = \sum_{n=0}^{\infty} \frac{\left(u \ln \left(\frac{1}{1-Z}\right)\right)^k}{k!} = e^{u \ln \frac{1}{1-Z}} = (1 - Z)^{-u}$$

$$(1 - Z)^{-u} = \sum_{n,k} \frac{\begin{bmatrix} n \\ k \end{bmatrix}}{n!} Z^n u^k$$

7.1.2 Числа Стирлинга II рода

$$\begin{Bmatrix} n \\ k \end{Bmatrix} \quad \operatorname{Set} \operatorname{Set}_{>0} Z$$

$$\bigcup_{k=0}^{\infty} \operatorname{Set}_{=k}(u \times \operatorname{Set}_{>0} Z)$$

$$\sum_{k=0}^{\infty} \frac{\left(u(e^Z - 1)^k\right)}{k!} = e^{ue^Z - u} = \sum_{n,k} \frac{\begin{Bmatrix} n \\ k \end{Bmatrix}}{n!} z^n u^k$$

7.1.3 Средняя стоимоть

- $A \quad a_{n,k} = [z^n u^k] A(u, z)$ — количество объектов веса n стоимости k

$$w_n = \frac{\sum_{k=0}^{\infty} k a_{n,k}}{\sum_{k=0}^{\infty} a_{n,k}} = \frac{[z^n] \left(\frac{\partial}{\partial u} A(u, z)\right)\big|_{u=1}}{[z^n] A(1, z)}$$

1. Разбиение на слагаемые, порядок важен Аналогично рассотовке перегорожок, $\operatorname{Seq} \operatorname{Seq}_{>0} Z$

$$\operatorname{Seq}\ (u \times \operatorname{Seq}_{>0} Z)$$

$$\frac{1}{1 - z} - 1 = \frac{z}{1 - z}$$

$$A(u, z) = \frac{1}{1 - \frac{uz}{1-z}} = \frac{1 - z}{1 - z - uz}$$

$$\left.\frac{\partial A(u, z)}{\partial u}\right|_{u=1} = \left.\frac{z(1 - z)}{(1 - z - uz)^2}\right|_{u=1} = \frac{z(1 - z)}{(1 - 2z)^2}$$

Числитель

$$[z^n] \frac{z(1 - z)}{(1 - 2z)^2} = \frac{2^n (n + 1)}{4}$$

Знаменатель

$$[z^n] \frac{1 - z}{1 - 2z} = 2^n - 2^{n-1} = 2^{n-1}$$

Среднее число слагаемых:

$$\frac{2^n (n + 1)}{2^{n-1} \cdot 4} = \frac{n + 1}{2}$$

2. Среднее число циклов в перестановке

$$A(u, z) = (1 - z)^{-u}$$

$$\frac{\partial}{\partial u} A(u, z) = \frac{\partial}{\partial u} e^{u \ln \frac{1}{1-z}} = \ln \frac{1}{1 - z} \cdot e^{u \ln \frac{1}{1-z}}$$

Подставляем $u = 1$:

Числитель

$$[z^n]\frac{\ln\left(\frac{1}{1-z}\right)}{1-z} = B(z)$$

Знаменатель

$$(1-z)^{-u}u = 1 = \frac{1}{1-z}$$

$$[z^n]\frac{1}{1-z} = 1$$

$$(z + \frac{1}{2}z^2 + \frac{1}{3}z^3 + \dots + \frac{1}{k}z^k + \dots) \cdot (1 + z + z^2 + \dots)$$

$$[z^n]B(z) = \sum_{k=1}^{n} \frac{1}{k} = H_n \sim \log n$$

TODO Лекция 8

Лекция 9

9.1 Теория вычислимости

Σ — алфавит, $\Sigma^* = \bigcup_{k=0}^{\infty} \Sigma^k$. $L \subset \Sigma^*$ — формальный язык.

Определение. L — **разрешимый (рекурсивный)**, если $\exists$ программа P, такая что:

- $x \in L \implies P(x) = 1$
- $x \notin L \implies P(x) = 0$

Примечание. Раньше называли рекурсивным, так как использовали рекурсивные функции. Это определение появилось до языков программирования

Примечание. Множетво разрешимых языков счетно

```
fn p(x: Word) -> Res {
    loop { }
    return 1;
}
```

Примечание. Любой конечный язык является разрешимым

```
fn p(x: Word) -> Res {
    if x == x_1 { // x_1 ∈ L
        return 1;
    }
    if x == x_2 { // x_2 ∈ L
        return 1;
    }
    // ...
}
```

Определение. L — **полуразрешимый (перечислимый, рекурсивно перечислмый)**, если $\exists p$, такая что:

1. $x \in L \implies p(x) = 1$
2. $x \notin L \implies p(x) \neq 1$

Примечание. Множество полуразрешимых языков счетно

Примечание. Разрешимый $\implies$ Полуразрешимый

Примечание. Σ^*, $\mathbb{N}^+$, $Prog$ — будут для нас эквивалентными понятиями, когда будем говорить про формальный язык.

- $\Sigma^* \leftrightarrow Prog$ — те строки, которые не являются программами, будем считать программами, которые 'зависают' на любом входе
- $\Sigma^* \leftrightarrow \mathbb{N}^+$ — занумеруем строки в градуированном лексикографичеком порядке

$$\varepsilon, `0`, `1`, `00`, `01`, `10`, `11`, `000`, `001`$$

Определение. Арифметические операции, `if`, `for`, `while`, вызов функций. p — программа, x — слово — запустить программу p на слове x, запустить программу p на слове x с ограничением на время $TL = t$, и ограничением на память $ML = m$

Определение. L — перечислимый, если $\exists p$ которая на пустом входе выводит любое слово из языка хотя бы один раз. $\forall x \in L\ \exists t(x) \quad P\big|_{TL=t}$ выводит x

Пример.

```
fn zeroes() {
    for i in [0..] {
        let s = '0' * i;
        println!("{}", s)
    }
}
```

Теорема 9.1.1. L — перечислимый $\Leftrightarrow$ L — полуразрешимый

Доказательство.

($\Rightarrow$) Пусть `listL()` — перечисляет L

```
fn inL(x: Word) -> Res {
    async { listL(); }
    if x.is_printed() {
        return 1;
    }
    return 0;
}
```

($\Leftarrow$) `inL()` — полуразрешиель L. Если напишем `listL()`, то он зависнет. Введем таймер.

```
fn listL() -> Res {
    for t in [0..] {
        for x in sigma[0..t] {
            if inL(x).await(t) {
                println!("{}", x)
            }
```

```
        }
    }
}
```

□

Примечание. Кодировать пару $\langle x, y \rangle$ можем

Определение. Универсальный язык:

$$U = \{\langle p, x\rangle \big| \text{программа } p(x) = 1\}$$

Свойство 1. *U — полуразрешим — Тьюринг полный*

Теорема 9.1.2. U — не разрешим

Доказательство. Допустим существует программа `inU((p, x): (Word, Word))`, которая разрешает U

```
fn q(x: Word) -> Res {
    if inU((x, x)) {
        return 0;
    } else {
        return 1;
    }
}
```

`q` никогда не зависает. Вызовем `q(q)`

- $inU(\langle q, q\rangle) = 1$
  - $q(q) = 0$
  - $\langle q, q\rangle \in U \implies q(q) = 1$
- $inU(\langle q, q\rangle) = 0$
  - $\langle q, q\rangle \notin U \implies q(q) \neq 1$
  - $q(q) = 1$

Получается, что q не возвращает ни 0 ни 1. Противоречие □

Свойство 1. *A, B разрешимы, $A \cup B$ — разрешим*

Свойство 2. *A, B разрешимы, $A \cap B$ — резрешим*

Свойство 3. *A — разрешим, $\overline{A} = \Sigma^* \setminus A$ — разрешим*

Свойство 4. *A, B полуразрешимы, $A \cap B$ — полуразрешим*

Свойство 5. *A, B полуразрешимы, $A \cup B$ — полуразрешим*

Доказательство.

```
fn p() -> Res {
    for i in [0..] {
        if inA.await(t) {
            return 1;
        }
        if inB.await(t) {
            return 1;
        }
    }
}
```

□

Теорема 9.1.3 (Поста)**.** L и $\overline{L}$ оба полуразрешимы $\implies$ L — разрешим

Доказательство.

```
fn inL(x: Word) -> Res {
    for t in [0..] {
        if inL(x).await(t) {
            return 1;
        }
        if inCL(x).await(t) { // CL = L̄
            return 0;
        }
    }
}
```

□

Теорема 9.1.4. Не существует языка программирования, который поддерживает все три свойства

1. Программа не зависает
2. Любой разрешимый язык, распознается программой на этом языке
3. Функция $\langle p, x\rangle \mapsto p(x)$ вычислима

Лекция 10

$q_1, q_2, \dots, q_n, \dots$ — программы

1. $q_i(j)$ не зависает
2. $\forall$ разрешимого A, $A = \{j \big| q_i(j) = 1\}$ для некоторогоi
3. $(i, j) \mapsto q_i(j)$ — вычислимая функция

$$L(p) = \{x \big| p(x) = 1\}$$

- A — перечислимый язык

- p_A — полуразрешитель A
- $L(p_A) = A$
  - $L : Prog \to 2^{\Sigma^*}$
  - $L : Prog \to RE$ (Recursively Enumerable)
- X — свойство языков $X \subset RE$
- Конечные языки $Finite \subset RE$
- Языки, содержащие ε $\underbrace{X_\varepsilon}_{\text{prop}} \subset \underbrace{RE}_{\text{prop}}$

  $\underbrace{A}_{\text{lang}} \in \underbrace{X_\varepsilon}_{\text{prop}} \Leftrightarrow \underbrace{\varepsilon}_{\text{string}} \in \underbrace{A}_{\text{lang}}$

Посмотрим на их типы

- * — `string`
- $lang$ — `set<string>`
- $prop$ — `set<lang>`

Определение. X — **свойство перечислимых языков**.
Язык свойства $L : prop \to lang$ $L(X) = \{p \big| L(p) \in X\}$

Пример. $L(RE) = \{p \big| L(p) \in RE\} = Prog$

```
fn q(p: Prog) -> bool {
    return 1;
}
```

Пример. $L(\emptyset_p) = \{p \big| L(p) \in \emptyset_p\} = \emptyset_e$

```
fn q(p: Prog) -> bool {
    return 0;
}
```

Теорема 10.0.1 (Rice, Успенский-Райс)**.**

- $X \subset RE$
- $X \neq \emptyset$
- $X \neq RE$

Тогда $L(X)$ не разрешим
Т.е. никакое нетривиальное свойство перечислимых языков не разрешимо.

Доказательство. X — нетривиальное свойство.
$\varnothing_C \notin X$
$A \in X$
A полуразрешается p_A
$]L(X)$ — разрешим
q_X — разрешитель $L * X$

$$q_X(p) = \begin{cases} 1 & L(p) \in X \\ 0 & L(p) \notin X \end{cases}$$

```
fn c() {
    loop {
        q_x(c) = 0;
        q_x(p_A) = 1;
    }
}
```

```
fn a(p: Prog, x: Word) {
    let s = "
fn s(y: Word) {
    if p(x) == 1 {
        return p_A(y);
    } else {
        loop { };
    }
}";
    return q_X(s);
}
```

$$p(x) = 1 \implies \forall y\ s(y) = p_A(y) \implies L(s) = L(p_A) = A \in X \implies q_X(s) = 1$$
$$p(x) \neq 1 \implies \forall y\ s(y) \text{ зависает } \implies L(s) = \varnothing_e \notin X \implies q_X(S) = 0$$

$\square$

Определение. HALT $= \{p | p$ останавливается на $\varepsilon\}$

Определение. m-сведение mapping (many-to-one).

- A, B — языки

$A \leq_m B$ — A сводится к языку B, если существует всюду определенная вычислимая функция $f \colon x \in A \Leftrightarrow f(x) \in B$

Лемма 6. *$A \leq_m B$, B разрешим $\implies$ A разрешим*

```
fn inA(x: Word) {
    return inB(f(x));
}
```

Лемма 7. *$A \leq_m B$, A неразрешим $\implies$ B неразрешим*

Лекция 11

11.1 Quine на Java

```java
import java.util.*;
import java.io.*;

class A {
	String s = "?";

	public run() {
		System.out.println(s)
	}

	public static void main(String[] args) {
		new A().run();
	}

}
```

Попробуем вставить код нашей программы в переменную `s`. Тогда будет выводится не код нашей программы. Что мы сделали: Подставили 'старый' код вместо `?` в `s`. Напишем новую программу, которая будет делать это за нас.

```java
import java.util.*;
import java.io.*;

class A {
	String s = "?";

	public void run() {
		int d = s.find('?');
		System.out.println(s.substring(0, d) + s + s.substing(d + 1, 0));
	}

	public static void main(String[] args) {
		new A().run();
	}
```

```

        public String escape(String s) {
                // Escapes symbols ', ", \n
                return s;
        }
}
```

Теорема 11.1.1 (о рекурсии)**.**

- $V(x, y)$ — вычислимая функция от 2х аргументов
- $\exists$ программа $r(t)$
- $\forall t\ V(r, t) = r(t)$

Предположим, что по исходному коду программы можно сделать какие то выводы. Пусть программа зная свой исходный код, сделает наоборот.

Теорема 11.1.2. halt не разрешима

Доказательство. Пусть существует *halt* — программа которая говорит, останавливается ли программа.

```python
def q(x):
    if halt(q):
        while True:
            pass
    else:
        return 1

```

□

Доказательство.

```python
def V(q, x):
    if halt(q):
        while True:
            pass
    else:
        return 1

```

$\forall t : r(t) = V(r, t)$. $V(q, t)$ зависает $\Leftrightarrow$ $q(\varepsilon)$ останавливается. $r(\varepsilon) = V(r, \varepsilon)$ зависает $\Leftrightarrow$ $r(\varepsilon)$ останавливается □

Теорема 11.1.3. U не разрешим

Доказательство.

```
def q(x):
    if u(q, x):
        return 0
    else:
        return 1
```

□

Теорема 11.1.4. Любое нетривиальное свойство перечислимых языков не разрешимо

Доказательство. A — нетривиальное свойство $inA(p)$

- $L \in A$, $inL(x)$
- $M \notin A, (x)$

```
def p(x):
    if inA(p):
        return inM(x)
    else:
        return inL(x)
```

□

Теорема 11.1.5. В любой достаточно богатой формальной системе существует истинное ??? утверждение.

Примечание. Система достаточно богатая, если можно записать утверждение ‘Программа p останавливается на входе x‘

Примечание. Система формальная, если можно проверить, что доказательство верно.

Program p

$s =$ ‘p останавливается на входе x‘
for $t \leftarrow \Sigma^*$ **do**
  if t — доказательство S **then**
    return
  end if
end for

- S — ложно $\implies$ S — истинна
- S — истинно $\implies$ (S доказуемо $\implies$ S ложно)
  S — истинно, S — не доказуемо

Теорема 11.1.6. Любая достаточно богатая система не может доказать свою непротиворечивость

Теорема 11.1.7 (о неподвижной точке)**.** Для любой всюду определенной функции f существует программа $p: \ \forall t \ p(t) = q(t)$, где $q(t) = f(p)$

Доказательство.

```
def p(x):
    q = f(p)
    return q(x)
```

□

Лекция 12

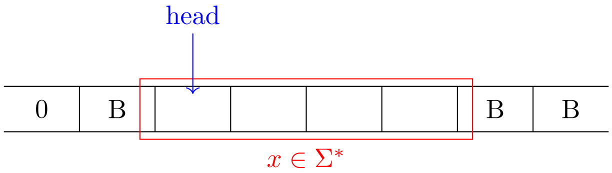
12.1 Абстрактные вычислители

12.1.1 Машина Тьюринга

Можно построить теорию вычислимости поставив в начало машину Тьюринга. Если существует машина Тьюринга, такая что останавливается на слове ...

Определение. **Тезис Тьюринга-Черча** —

Определение. Машина Тьюринга



- Ленточный алфавит Π
- $B \in \Pi$ — пробел
- Входной алфавит $\Sigma \subset \Pi$, $|\Sigma| \geq 2$
- $B \notin \Sigma$
- Q — конечной множество состояний
- $s, Y, N \in Q$
- $\delta : Q \times \Pi \to Q \times \Pi \times \{\leftarrow, \rightarrow, \downarrow\}$ — переходы

Пример. Язык 0^*

	0	1	B
S	S, 0 $\rightarrow$	N, 1, $\downarrow$	Y, B, $\downarrow$

Пример. Палиндромы

	0	1	B	x
S	$r_0, x, \to$			
r_0	$r_0, 0, \to$	$r_0, 1, \to$	$c_0, B, \leftarrow$	$c_0, x, \leftarrow$
c_0	$l, x, \leftarrow$	N		Y
l	$l, 0, \leftarrow$	$l, 1, \leftarrow$	$s, B, \to$	$S, x, \to$
r_1	$r_1, 0, \to$	$r_1, 1, \to$	$c_1, B, \leftarrow$	$c_1, x, \leftarrow$
c_1	$l, x, \leftarrow$	N		Y

Примечание. $0^n1^n2^n$ — не контекстно свободный, распознается машиной Тьюринга

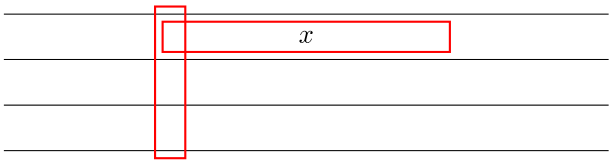
Определение. **Машина Тьюринга** — $\langle \Sigma, \Pi, B \in \Pi \setminus \Sigma, Q, s \in Q, Y \in Q, N \in Q, \delta \rangle$

Определение. **Мгновенное описание машины Тьюринга** (snapshot) — $\langle t, q, pos \rangle$, где t — слово на ленте (без trailing/leading пробелов), q — состояние, pos — позиция головки

Определение. $\langle t, q, pos \rangle \vdash \langle t', q', pos' \rangle$ — **переходит за один шаг**

Определение. $L(m) = \{x \big| \langle x, s, 1 \rangle \vdash^* \langle t, Y, pos \rangle\}$ — множество слов, **допускаемых машиной Тьюринга**

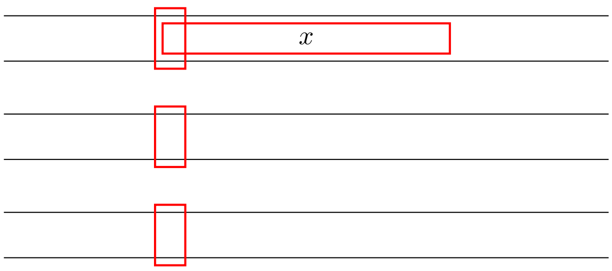
12.1.2 Многодорожечная машина Тьюринга



Определение. Одна головка, но k дорожек
$\delta : Q \times \Pi^k \to Q \times \Pi^k \times \{\leftarrow, \to, \downarrow\}$

Лемма 8. *МДМТ = МТ (равно по мощности),* $\Pi^1 = \Pi^k$

12.1.3 Многоленточная машина Тьюринга



Определение. k дорожек, у каждой головка
$\delta : Q \times \Pi^k \to Q \times \Pi^k \times \{\leftarrow, \to, \downarrow\}^k$

Теорема 12.1.1. МЛМТ $\equiv$ МТ

Доказательство. k лент $\mapsto$ $2k$ дорожек $\mapsto$ 1 лента $\square$

Примечание. Многоленточная машина Тьюринга уже больше похожа на 'обычный компьютер':

- процессор
  - регистры
- память
  - данные
  - код
  - стек

Лента *входа* (только читаем). Для каждого типа памяти заведем ленту, и еще одну ленту для *регистров*, еще одну ленту для *счетчиков*. Ленту *счетчиков* используем для доступа к произвольному адресу в памяти *данных*.

Примечание. **Тезис Тьюринга-Черча** (и другие)
L — разрешим (перечислим) $\Leftrightarrow$ L можно распознать (полуразрешить, перечислить) на МТ.

Примечание. **Бонус**
T — время на компьютере, тогда на МТ — $\text{poly}(T)$

Примечание. Потребуем от МТ дополнительных ограничений:

- Лента односторонне бесконечной
- На ленту не записывается B

— Экологичная МТ

Примечание. Односторонняя бесконечная лента + не писать $B \equiv$ МТ

Лекция 13

13.1 Машина Тьюринга

Рассмотрим мгновенной описание МТ с односторонней бесконечной лентой. Назовем строку до головки α, в строку вместе с текущим символом β. q — состояние.

Обозначение. $\alpha\#_q\beta$

13.2 Стековые машины

Обобщим понятие автомата с магазинной памятью с одним стеков на большее число стеков. k-стековая машина.

Теорема 13.2.1. Автомат с двумя стеками — 2-стековая машина $\equiv$ МТ

Доказательство. Будем хранить в одном стеке α в другом β из 13.1 $\square$

Примечание. Добавление стеков не увеличивает мощность

13.3 Счетчиковые машины

Определение. k-счетчиковая машина, машина с k счетчиками, хранящими натуральные числа, переход только по условию равен ли счетчик нулю

k-стековая машина $\mapsto$ $(k+1)$-счетчиковая машина
k-счетчиковая машина $\mapsto$ 2-счетчиковая машина

Теорема 13.3.1. k-стековая машина $\subset$ $(k+1)$-счетчиковая машина

Доказательство. Запишем один стек так:

$$A_0 + A_1 b + A_2 b^2 + \cdots + A_t b^t$$

, где A_i символ на стеке, так что A_0 лежит на самом верху — один счетчик (число в b-ичной системе счисления) для каждого стека. Другой счетчик. Можем сделать операции: деление с остатком, умножение, прибавление константы Доделать $\square$

Теорема 13.3.2. k-счетчиковая машина $\subset$ 2-счетчиковая машина

Доказательство. Доделать $\square$

13.4 Нормальные алгорифмы Маркова

Доделать

13.5 Грамматики нулевого класса

Контекстно свободные грамматики — грамматики второго класса, правила вида $A \to \alpha$. В грамматике нулевого класса правила вида $\alpha \to \beta$, где в обоих случаях α, β — состоят из терминалов и нетерминалов, т.е. $\alpha \in (\Sigma \cup N)^*$, α содержит N, $\beta \in (\Sigma \cup N)^*$

Определение. $\xi \Rightarrow \eta$, если

- $\xi = \xi_1 \alpha \xi_2$
- $\eta = \eta_1 \beta \eta_2$
- $\xi_1 = \eta_1$
- $\xi_2 = \eta_2$
- $\alpha \to \beta \in \Gamma$

$$L(\Gamma) = \{w | S \Rightarrow^* w\}$$

Теорема 13.5.1. $L = L(\Gamma)$ для грамматики 0 класса $\Leftrightarrow$ L — перечислимый

($\Rightarrow$) Запустим BFS по $\Rightarrow$

- $x \in L \implies$ рано или поздно найдем
- $x \notin L \implies$ зависнем

($\Leftarrow$) L перечислим $\implies$ L полуразрешается МТ

- $S \to BU\#_Y UE$
- $U \to cU$
- $U \to \varepsilon$
- $B\#_S \to Z$
- $\forall c \in \Sigma \quad Zc \to cZ$

13.6 Клеточный автомат

Есть лента с ячейками, каждая ячейка раскрашена в один из двух цветов. За один шаг каждая ячейка на d ячеек слева и справа, происходит переход в новое состояние.

- $\delta : Q^{2d+1} \to Q$, где Q — множество клеток

Лекция 14

Определение. PCP (ПСП) — Проблема соответствия Поста

- $|\Sigma| \geq 2$
- n пар слов: $\langle a_i, b_i \rangle$, $a_i, b_i \in \Sigma^*$

$(A, B) \in PCP$
$\exists i_1, i_2, \dots, i_k$
$i_j \in \{1, 2, \dots, n\}$
$k \geq 1$

$$a_{i_1} a_{i_2} \dots a_{i_k} = b_{i_1} b_{i_2} \dots b_{i_k}$$

Теорема 14.0.1. PCP не разрешим (но перечислим)

$$U_{TM} \leq_m PCP_1 \leq_m PCP$$

, где $PCP_1 = \{(A, B) \in PCP \big| \exists$ решение $i_1 = 1\}$

Лемма 9. $PCP_1 \leq_m PCP$

Доказательство. Доделать □

Определение. Язык списка

- $x_1, x_2, \dots, x_n \in \Sigma^*$
- $I_n = \{\boxed{1}, \boxed{2}, \dots, \boxed{n}\}$
- $L_X \subset (\Sigma \cup I_n)^*$

$$\begin{array}{l} S \to x_i S \boxed{i} \quad i = \overline{1, n} \\ S \to \varepsilon \end{array}$$

Пример. UAG $= \{\Gamma \big| \Gamma$ — КС грамматика, $\forall x$ имеет ≤ 1 деревьев разбора$\}$
UAG — не разрешимый
$PCP \leq_m \overline{\text{UAG}}$

Доказательство. Построим язык $L_A \cup L_B$

$$\Gamma_{L_A \cup L_B} : \begin{array}{l} S \to A \\ S \to B \\ A \to \text{язык списка } A \\ B \to \text{язык списка } B \end{array}$$

Если PCP имеет решение, то грамматика является неоднозначной. Если грамматика неоднозначная, то PCP имеет решение. Этим UAG сводится к PCP □

Пример. $\text{DG} = \{\langle \Gamma_1, \Gamma_2 \rangle \big| L_{\Gamma_1} \cap L_{\Gamma_2} \neq \varnothing\}$
DG — не разрешимый

Доказательство. Возьмем язык списка грамматики Γ_A и Γ_B. Они пересекаются тогда и только тогда, когда PCP имеет решение □

Пример. Задача о замощении $\text{TILING}_{4,\frac{1}{4}}$ — 4-сетка, $\frac{1}{4}$ плоскости

- Набор типов полимино $\mathbb{P} = \langle P_1, P_2, \ldots, P_n \rangle$
- $\text{TILING}_{4,\frac{1}{4}} = \{\mathbb{P} \big|$ можно замостить $\frac{1}{4}$ плоскости полимино из $\mathbb{P}\}$

Доказательство. $\overline{\text{HALT}}_{TM} \leq_m \text{TILING}$ Доделать □