Московский физико-технический институт
Факультет инноваций и высоких технологий
Сложность вычислений, осень 2017
Схемная сложность

Схемой из функциональных элементов называется ориентированный граф без циклов, каждая вершина которого помечена одной из пяти меток: вход, выход, $\wedge$, $\vee$, $\neg$. Граф обладает следующими ограничениями на входящие и исходящие степени:

Тип метки	Входящая степень	Исходящая степень
вход	0	любая
выход	1	0
$\neg$	1	любая
$\wedge$	2 (вар. любая)	любая
$\vee$	2 (вар. любая)	любая

Если каждой вершине входа присвоить значение 0 или 1, то индуктивно можно определить значение всех остальных вершин: значение вершины, помеченной $\neg$, $\wedge$ или $\vee$ определяются как соответствующая функция от значений вершин, из которых в неё ведут рёбра; значение вершины, помеченной «выход», совпадает со значением вершины, из которой в неё ведёт ребро.

1. Докажите корректность этого определения. (Т.е. при отсутствии циклов значение в каждой вершине посчитается однозначно).

Таким образом, схема с n входными и k выходными вершинами задаёт функцию из $\{0,1\}^n$ в $\{0,1\}^k$. В связи с задачами разрешения рассматривают случай $k = 1$, по умолчанию мы будем предполагать этот случай.

Последовательность схем $\{C_n\}_{n=1}^{\infty}$ (у схемы C_n есть n входных вершин и одна выходная) распознаёт множество $A \subset \{0,1\}^*$, если при всех $x \in \{0,1\}^*$ выполнено $C_{|x|}(x) = 1$ при $x \in A$ и $C_{|x|}(x) = 0$ при $x \notin A$. Говорят также не о распознавании множества, а о вычислении функции из $\{0,1\}^*$ в $\{0,1\}$.

2. Докажите, что любое множество можно распознать некоторой последовательностью схем.

Размером схемы называется количество вершин в соответствующем графе. Классом $\mathbf{SIZE}(f(n))$ называется множество языков, которые распознаются схемами размера $O(f(n))$. Классом $\mathbf{P}/_{\mathrm{poly}}$ называется объединение всех $\mathbf{SIZE}(n^c)$, т.е. множество языков, распознаваемых схемами полиномиального размера.

3. Докажите, что класс $\mathbf{P}/_{\mathrm{poly}}$ не изменится, если в качестве размера вместо числа вершин брать число рёбер.

4. Докажите, что класс $\mathbf{P}/_{\mathrm{poly}}$ не зависит от того, какая входящая степень разрешена для вершин типов $\wedge$ и $\vee$.

5. Докажите, что $\mathbf{P} \subsetneq \mathbf{P}/_{\mathrm{poly}}$.

6. Докажите, что существует разрешимый язык, лежащий в $(\mathbf{P}/_{\mathrm{poly}}) \setminus \mathbf{P}$.

Классом $\mathbf{DTIME}(f(n))/_{a(n)}$ называется множество языков L, для которых существует машина Тьюринга M и последовательность слов α_n длин $a(n)$, такие что $x \in L$ тогда и только тогда, когда $M(x, \alpha_{|x|}) = 1$, при этом машина работает $O(f(|x|))$ шагов.

7. Приведите пример неразрешимого языка, лежащего в $\mathbf{DTIME}(n)/_1$.

8. Докажите, что $\mathbf{P}/_{\text{poly}} = \bigcup_{c,d=1}^{\infty} \mathbf{DTIME}(n^c)/_{n^d}$.

Глубиной схемы называется максимальная длина (ориентированного) пути, идущего от входной вершины до выходной.

9. Докажите, что любой язык распознаётся схемой глубины не больше 4.

Классом $\mathbf{NC}^d$ называется класс языков, распознаваемых схемами полиномиального размера и глубины $O(\log^d n)$, в которых входящая степень всех вершин, помеченных $\wedge$ и $\vee$, равна 2. Классом $\mathbf{NC}$ называется объединение всех $\mathbf{NC}^d$. Классом $\mathbf{AC}^d$ называется класс языков, распознаваемых схемами полиномиального размера и глубины $O(\log^d n)$, в которых входящая степень всех вершин, помеченных $\wedge$ и $\vee$, произвольная.

10. Докажите, что $\mathbf{NC}^d \subset \mathbf{AC}^d \subset \mathbf{NC}^{d+1}$. Докажите, что $\bigcup_{d=1}^{\infty} \mathbf{AC}^d = \mathbf{NC}$.

11. Докажите, что $\mathbf{NC}^0 \neq \mathbf{AC}^0$.

12. Докажите, что язык $\mathsf{PAL} = \{a \mid a = a^R\}$, где a^R — слово a, записанное в обратном порядке, лежит в $\mathbf{AC}^0$.

13. Докажите, что язык $\mathsf{PARITY} = \{x_1 x_2 \dots x_n \mid x_1 + x_2 + \dots + x_n \equiv 0 \mod 2\}$ лежит в $\mathbf{NC}^1$. (Про этот язык можно доказать, что он не лежит в $\mathbf{AC}^0$, таким образом, $\mathbf{AC}^0 \subsetneq \mathbf{NC}^1$).

Классы $\mathbf{NC}^d$ и $\mathbf{AC}^d$ естественным образом распространяются с языков на произвольные булевы функции (т.к. их тоже можно вычислять схемами).

14. Как можно точнее классифицируйте функции сложения и умножения в двоичной записи. (Как минимум, докажите принадлежность этих функций к $\mathbf{NC}$).

15. Как можно точнее классифицируйте функции умножения $n \times n$ матриц по модулю 2 и возведения $n \times n$ матриц в степень.

16. Докажите, что задача PATH лежит в $\mathbf{NC}$. Докажите, что логарифмическая сводимость не выводит из класса $\mathbf{NC}$. Таким образом, $\mathbf{NL} \subset \mathbf{NC}$.

17. Докажите, что $\mathbf{NC}^1 \subset \mathbf{L}$. (Здесь на язык из $\mathbf{NC}^1$ накладывается дополнительное условие лог-равномерности: схему C_n можно построить по 1^n логарифмическим по памяти алгоритмом).

18. Докажите, что $\mathbf{NL} \subset \mathbf{AC}^1$.

19. Рассмотрим язык $\mathsf{MAJORITY} = \{(x_1, \dots, x_n) \mid n$ нечётно и $\sum x_i > \frac{n-1}{2}\}$.

а) Докажите, что $\mathsf{MAJORITY} \in \mathbf{NC}^2$. (Указание: используйте рекурсивный алгоритм, глубина получится равной $O(\log n \log\log n)$).

б) Докажите, что $\mathsf{MAJORITY} \in \mathbf{NC}^1$. (Указание: постройте случайное троичное дерево из элементов, вычисляющих большинство из трёх аргументов и докажите вероятностным методом, что среди них есть правильное. Более сложный вариант — использовать сортирующую сеть Айтая–Комлоша–Семереди).