Лекция 9

Подготовил Чижов Даниил, КН-302

16.11.15

Теория алгоритмов 2015

N = NP?

(Можно ли решить огромное количество важных задач за разумное время?)

- Да
- Нет
- Не знаю и не могу знать
- Все равно

Дальше будем предполагать, что $P \neq NP$

Рассмотрим классы функциональных задач FP и FNP

Задача

Задача поиска $R = \{(x, y) | x, y \in \sum\}$.

Определение

$R \in FP \Rightarrow \exists$ ДМТ, которая по x вычисляет y : $(x, y) \in R$ за полином.

$R \in FNP \Rightarrow \exists$ НМТ, которая по x вычисляет y : $(x, y) \in R$ за полином.

Определение

$R \in FNP \Rightarrow \exists$ ДМТ M_R, которая за полином проверяет, что $(x, y) \in R$.

То есть, умеет быстро проверять, что y является ответом для x.

Задача

$F - SAT$: по F найти $x_1, .., x_n$: $F(x_1, .., x_n) = 1$.

Определение

Сводимость по Куку $\underset{p}{\rightarrow}$.

$R_1 \underset{p}{\rightarrow} R_2$, если $\exists$ МТ для задачи R_1, которая использует оракул R_2 и работает за полином.

Определение

Оракул R_2 - это гипотетическое устройство, решающее R_2 за 1.

Замечание

Сводимость по Карпу - это частный случай сводимости по Куку.

То есть, если что-то свелось по Карпу, то оно свелось и по Куку. Обратное не верно.

$$x_1 \vee \bar{x_2} \vee x_3$$
$$F = x_1 \vee x_2 \vee \bar{x_3}$$
$$\dots\dots\dots$$

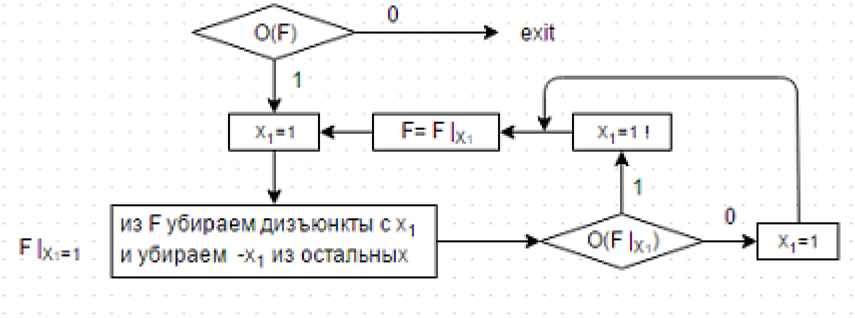
Задача

$F - SAT$: по F найти $x_i,\ F(x_i) = 1$

Оракул решает задачу SAT за 1. Мы ей говорим формулу F, а она отвечает Да или Нет.

Определение

$F|_{x_1}$- *выполнима* $\Leftrightarrow\ \exists x_1, .., x_n : F(x_1, .., x_n) = 1 \wedge x_1 = 1$


O(F)
0
exit
1
x1=1
F= F |x1
x1=1 !
1
0
F |x1=1
из F убираем дизъюнкты с x1
и убираем -x1 из остальных
O(F |x1)
x1=1

Определение

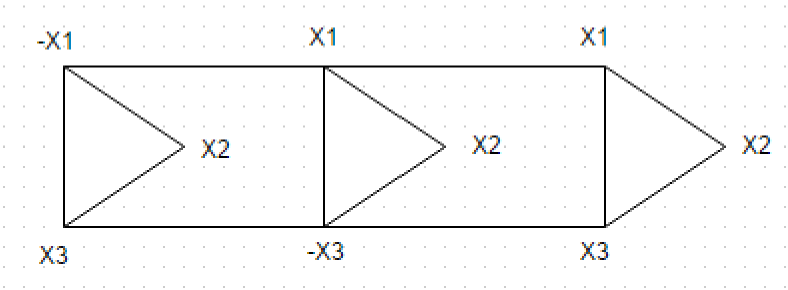
Сводимость по Левину $\underset{l}{\rightarrow}$.
$R_1 \underset{l}{\rightarrow} R_2$, *если* $\exists\, f, g$*- полиномиально вычислимые полиномиальные функции и* $(x, g(y)) \in R_1 \Leftrightarrow (f(x), y) \in R_2$.
f *- преобразует задачу* R_1 *к* R_2,
g *- преобразует ответ.*

Если бы мы сводили по Левину задачу $F - 3SAT$ к задаче F_VC, то выполнили бы следующие действия:
$F \rightarrow Graph \rightarrow VC \rightarrow x_1, .., x_n : \; F(x_1, .., x_n) = 1$.
То есть все сведения, которые мы делали, - это сведения по Левину.

Замечание

Сводимость по Левину - это обобщение сводимости по Карпу для функциональных задач.

$$F = \begin{array}{c} x_1 \vee x_2 \vee \bar{x}_3 \\ \bar{x}_1 \vee x_2 \vee x_3 \\ x_1 \vee x_2 \vee x_3 \end{array} \xrightarrow{f}$$



$$\rightarrow \left\{ \begin{array}{c} x_2 \\ x_2 \\ x_2 \end{array} \right\} \xrightarrow{g} \begin{array}{l} x_2 = 1 \\ x_1 = 0 \\ x_3 = 0 \end{array}$$

Определение

$R \in FNP$ - трудная $\Leftrightarrow \forall R' \in FNP \underset{l}{\to} R$.

Определение

$R \in FNP$ - полная $\Leftrightarrow R \in FNP$ - трудная и $R \in FNP$.

Определение

Самосводимые задачи. FNP-полные задачи всегда сводятся по Карпу к своим NP-полным аналогам.

Определение

Decision version. $R = (x, y)$
$L_R = \{x :\ \exists y : (x, y) \in R\}$

Утверждение

Все FNP-полные задачи самосводимы.

Доказательство

$R \underset{l}{\rightarrow} F - SAT \underset{p}{\rightarrow} SAT \underset{k}{\rightarrow} \textit{Decision version } R$

Алгоритм:

1. Берем R и переводим его в формулу F из задачи $F - SAT$.

2. Мы знаем, что $F - SAT \underset{p}{\rightarrow} SAT$. *Запускаем этот алгоритм.*

3. Каждый вызов $O(F)$ *заменяем на вызов* $O_{L_R}(h(F))$.

4. return $g(\text{решение } F - SAT)$.