МФТИ, сложность вычислений, осень 2023
Семинар 02. Классы **P**, **NP**, **coNP**

Определение. Класс **NP** состоит в точности из тех языков A, для которых существует некая машина V двух аргументов, работающая за полином от длины первого из них, т.ч. $\forall x \in \Sigma^*$ выполнено $x \in A \Leftrightarrow \exists s : V(x, s) = 1$. В таком случае V называют верификатором, а подходящее s — сертификатом для данного x.

Определение. Недетерминированной машиной Тьюринга называется кортеж объектов $\langle \Sigma, \Gamma, Q, q_{start}, q_{accept}, q_{reject}, k, \delta \rangle$, такой что Σ, Γ, Q — непустые конечные множества, $\Sigma \subset \Gamma$, k — целое положительное число, $q_{start}, q_{accept}, q_{reject}$ — три различных элемента Q, $\delta : (Q \setminus \{q_{accept}, q_{reject}\}) \times \Gamma^k \rightrightarrows Q \times \Gamma^k \times \{L, N, R\}^k$ (т.е. δ — многозначная функция). Говорим, что $x \in \Sigma^*$ принимается машиной M, если существует хотя бы одна ветвь вычислений (то есть последовательность выбора значений в δ), приводящая в q_{accept}.

Определение. Язык $L \subset \Sigma^*$ распознаётся недетерминированной машиной Тьюринга M, если для каждого $x \in \Sigma^*$ вычисление $M(x)$ останавливается на всех ветвях, причём слово x принимается машиной M, если и только если $x \in L$.

Определение. Язык $L \subset \Sigma^*$ распознаётся недетерминированной машиной Тьюринга M за время $O(T(n))$ (или просто $T(n)$), если M распознаёт L, а также для каждого $x \in \Sigma^*$ вычисление $M(x)$ останавливается не более чем за $O(T(n))$ шагов в любой ветви вычислений (при любых выборах значений δ), где $n = |x|$.

Определение. Если $T(n) : \mathbb{N} \to \mathbb{N}$, то класс $\mathbf{NTIME}(T(n))$ состоит в точности из тех языков, которые распознаются хоть какой-нибудь недетерминированной машиной Тьюринга за время $T(n)$.

Теорема. $\mathbf{NP} = \bigcup\limits_{c=1}^{\infty} \mathbf{NTIME}(n^c)$.

Определение. $\mathbf{coNP} = \{A \mid \overline{A} \in \mathbf{NP}\} = \{\overline{A} \mid A \in \mathbf{NP}\}$.

Теорема. Класс **coNP** состоит в точности из языков A, для которых существует некая машина V двух аргументов, работающая за полином от длины первого из них, т.ч. $\forall x \in \Sigma^*$ выполнено $x \in A \Leftrightarrow \forall s : V(x, s) = 1$.

Утверждение. Пусть $\mathsf{PRIMES} = \{n \mid$ число n является простым$\}$. Тогда $\mathsf{PRIMES} \in \mathbf{NP}$. Более того, $\mathsf{PRIMES} \in \mathbf{P}$.

Определение. Пусть G — граф. Максимальным паросочетанием в нём называется максимальное (по мощности) множество рёбер, попарно не имеющих общих концов.

Утверждение. Пусть $\mathsf{MATCHING} = \{(G, k) \mid$ максимальное паросочетание в графе G имеет размер в точности $k\}$. Тогда $\mathsf{MATCHING} \in \mathbf{P}$.

---

1. Пусть $A, B \in \mathbf{P}$. Докажите, что $\overline{A}, A \cup B, A \cap B \in \mathbf{P}$. Иными словами, класс **P** замкнут относительно дополнений, объединений и пересечений.

2. Докажите, что класс **P** замкнут относительно звезды Клини (то есть если $A \in \mathbf{P}$, то $A^* \in \mathbf{P}$, где $A^* = \{w_1 \dots w_k \mid k \geqslant 0, w_1, \dots, w_k \in A\}$).

3. Покажите, что в сертификатном определении класса **NP** можно добавить требование полиномиальности длины s (т.е. $|s| \leqslant p(|x|)$ для некоторого полинома $p(\cdot)$). Покажите также, что можно считать, что требование можно усилить до $|s| = p(|x|)$. Более того, в качестве $p(n)$ достаточно брать многочлен вида n^c.

4. Докажите, что функцию δ в определении недетерминированной машины можно считать двузначной (то есть воспринимать как пару функций δ_0 и δ_1) — тогда время работы возрастает в константное число раз.

5. Покажите, что если $A \in \mathbf{NTIME}(T(n))$, то $A \in \mathbf{DTIME}\left(2^{O(T(n))}\right)$. В частности, недетерминизм не расширяет класс разрешимых языков **R**.

6. Докажите, что класс **NP** замкнут относительно объединений, пересечений и звезды Клини.

7. Докажите также, что если $\mathcal{C}$ — произвольный сложностной класс, про который известно, что $\mathcal{C} \subset \mathbf{co}\mathcal{C}$, то $\mathcal{C} = \mathbf{co}\mathcal{C}$. Здесь, как и всюду в дальнейшем, $\mathbf{co}\mathcal{C} = \{A \mid \overline{A} \in \mathcal{C}\} = \{\overline{A} \mid A \in \mathcal{C}\}$. То же следует и из

посылки $\mathbf{co}\mathcal{C} \subset \mathcal{C}$. Выведите отсюда, что замкнутость $\mathbf{NP}$ относительно дополнений влечёт равенство $\mathbf{NP} = \mathbf{coNP}$.

8. Докажите, что класс $\mathbf{NP} \cap \mathbf{coNP}$ замкнут относительно дополнений.

9. Покажите, что $\mathbf{P} \subset \mathbf{NP} \cap \mathbf{coNP}$. Докажите, что если $\mathbf{P}$ совпадает хотя бы с одним из классов $\mathbf{NP}$ и $\mathbf{coNP}$, то он совпадает и с другим.

10. Докажите, что $\mathsf{GI} = \{(G_1, G_2) \mid \text{графы } G_1 \text{ и } G_2 \text{ изоморфны}\} \in \mathbf{NP}$. Неизвестно, лежит ли этот язык в $\mathbf{P}$. Докажите, что $\mathsf{GNI} = \{(G_1, G_2) \mid \text{графы } G_1 \text{ и } G_2 \text{ неизоморфны}\} \in \mathbf{coNP}$.

11. Докажите, что $\mathsf{TAUT} = \{\varphi \mid \text{пропозициональная формула } \varphi \text{ является тавтологией}\} \in \mathbf{coNP}$. Докажите, что $\mathsf{SAT} = \{\varphi \mid \text{пропозициональная формула } \varphi \text{ выполнима}\} \in \mathbf{NP}$.

12. Пусть $\mathsf{SUBSETSUM} = \{(n_1, n_2, \ldots, n_k, N) \mid \text{из набора чисел } n_1, \ldots, n_k \text{ можно выбрать подмножество с суммой } N\}$. Пусть также $\mathsf{UNARYSUBSETSUM} = \{(1^{n_1}, 1^{n_2}, \ldots, 1^{n_k}, 1^N) \mid \text{из набора чисел } n_1, \ldots, n_k \text{ можно выбрать подмножество с суммой } N\}$ (здесь под 1^x подразумевается строка из x единиц). Докажите, что $\mathsf{SUBSETSUM} \in \mathbf{NP}$ и $\mathsf{UNARYSUBSETSUM} \in \mathbf{P}$. Можно ли сказать, что $\mathsf{SUBSETSUM} \in \mathbf{P}$?

13. Установите принадлежность следующих языков классам $\mathbf{P}, \mathbf{NP}, \mathbf{coNP}$:

а) $\mathsf{PATH} = \{(G, s, t) \mid \text{в графе } G \text{ есть путь из } s \text{ в } t\}$;

б) $\mathsf{SPATH} = \{(G, s, t, k) \mid \text{в графе } G \text{ есть путь из } s \text{ в } t \text{ длины не больше } k\}$;

в) $\mathsf{LPATH} = \{(G, s, t, k) \mid \text{в графе } G \text{ есть простой путь из } s \text{ в } t \text{ хотя бы с } k \text{ ребрами}\}$;

г) $\mathsf{EULPATH} = \{(G, s, t) \mid \text{в графе } G \text{ есть эйлеров путь из } s \text{ в } t\}$;

д) $\mathsf{LTSP} = \{(G, l) \mid G$ — взвешенный граф, кратчайший проходящий через все вершины путь в котором имеет длину хотя бы $l\}$.

14. Докажите, что $\mathsf{PRIMES} \in \mathbf{coNP}$.

15. Докажите, что $\mathsf{FACTORING} = \{(N, a, b) \mid \text{у числа } N \text{ существует простой делитель на отрезке } [a, b]\} \in \mathbf{NP}$. Неизвестно, лежит ли этот язык в $\mathbf{P}$.

16. Докажите, что $\mathsf{RELATIVELY\text{-}PRIME} = \{(x, y) \mid x \text{ и } y \text{ взаимно просты}\} \in \mathbf{P}$.

1. Если $A, B \in \mathbf{P}$, то вопрос принадлежности произвольного слова x каждому из языков A, B разрешается за полиномиальное время.
2. Для входного слова x введите $dp[k]$ — логический индикатор того, лежит ли префикс слова x длины k в языке A^*.
3. Во-первых, если V работает лишь полиномиально долго, то слишком много битов s прочитать она просто не успеет, и их можно игнорировать. Во-вторых, короткие сертификаты можно искусственно раздуть до длины ровно $p(n)$ каким-нибудь незначащим мусором. В-третьих, любой многочлен $p(n)$ меньше какого-нибудь многочлена вида n^c во всех $n \geqslant 2$ (случай $n = 1$ можно зашить в машину и не требовать сертификата вообще).
4. Для заданной недетерминированной машины множество возможных значений δ в данной конфигурации — конечно. Если пронумеровать все эти элементы последовательными натуральными числами, то выбор δ_0 и δ_1 может симулировать запись такого номера.
5. Считаем δ двузначной. Если недетерминированная машина работает за $O(T(n))$, то есть за $c \cdot T(n)$, то на детерминированной машине можно смоделировать перебор всех ветвей вычислений за $2^{c \cdot T(n)}$.
6. Воспользуйтесь сертификатным определением **NP**.
7. Если $A \in \mathbf{co}\mathcal{C}$, то $\overline{A} \in \mathcal{C}$. Но раз $\mathcal{C} \subset \mathbf{co}\mathcal{C}$, то $\overline{A} \in \mathbf{co}\mathcal{C}$. Отсюда $\overline{\overline{A}} \in \mathcal{C}$, то есть $A \in \mathcal{C}$.
8. Если язык A лежит и в **NP**, и в **coNP**, то $\overline{A}$ лежит и в **coNP**, и в **NP**.
9. Если $\mathbf{P} = \mathbf{NP}$, то **NP** замкнут относительно дополнений (поскольку **P** замкнут), так что $\mathbf{NP} = \mathbf{coNP}$.
10. Изоморфизм графов можно задать биекцией множеств вершин, что и будет выступать сертификатом. При должном кодировании входных данных, можно считать, что $\mathsf{GNI} = \overline{\mathsf{GI}}$.
11. Воспользуйтесь сертификатным определением классов **NP** и **coNP**.
12. $\mathsf{UNARYSUBSETSUM} \in \mathbf{P}$, поскольку можно ввести динамическое программирование по типу рюкзака: $dp[i][k]$ — индикатор того, можно ли набрать сумму в точности k, используя некоторые из первых i входных чисел. Такая динамика работает за время, пропорциональное N, что не является полиномом при двоичной записи чисел.
13.
а) $\mathsf{PATH} \in \mathbf{P}$;
б) $\mathsf{SPATH} \in \mathbf{P}$;
в) $\mathsf{LPATH} \in \mathbf{NP}$;
г) $\mathsf{EULPATH} \in \mathbf{P}$;
д) $\mathsf{LTSP} \in \mathbf{coNP}$.
14. В дополнении к PRIMES лежат все составные числа. Доказать, что число не является простым, можно с помощью нетривиального делителя.
15. Можно пользоваться без доказательства тем фактом, что простоту числа можно доказать неким сертификатом.
16. Воспользуйтесь алгоритмом Евклида.