Занятие 5. Экспресс-курс по ТИ

Алгоритмы

1. Лабиринт представляет из себя доску $n \times n$, переход между некоторыми клетками разрешен (нет стены) или запрещен (есть стена). В одной из клеток лабиринта есть робот, который умеет двигаться за ход на одну клетку в любом из 4 направлений (робот знает в какой клетке начинает). Если он пытается пройти в стену, он в этом ходу остается на месте. Когда робот попадает в клетку $(0,0)$, он сразу побеждает. Придумайте алгоритм, по которому робот побеждает и оцените его время работы (или докажите, что его не существует), если:
(a) Робот знает лабиринт;

(b) Робот не знает лабиринт, но он узнаёт, когда сходил в стену;

(c) Робот не получает никакой обратной информации от своих ходов.

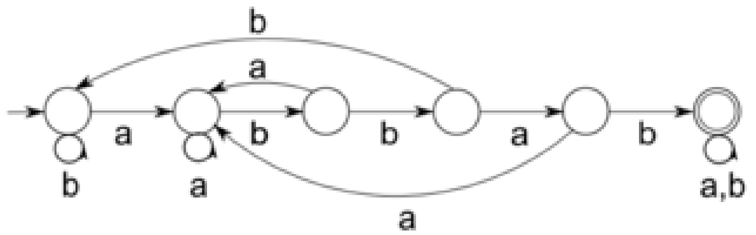
Определение. Пусть $f(x)$ и $g(x)$ – две функции, определённые в некоторой проколотой окрестности точки x_0. Тогда f является O-большим от g при $x \to x_0$, если существует такая константа C, что для некоторой окрестности x_0 выполняется

$$|f(x)| \leqslant C \cdot |g(x)|$$

2. Даны n неотрицательных чисел до $2 \cdot 10^9$. Придумайте алгоритм сортировки, который работает за время $O(n)$ и использует не больше $5 \cdot 32 \cdot n$ бит памяти. Используя формулу Стирлинга докажите, что если убрать ограничение на числа, то быстрее, чем за $O(n \cdot \log n)$ отсортировать не получится.

3. (**) Придумайте алгоритм работающий за полиномиальное время, который бы решал задачу коммивояжера на плоскости в 2-приближении (то есть длина найденного пути была бы не более чем в 2 раза хуже оптимального решения).

Автоматы



Определение. Детерминированный конечный автомат (DFA) – математическая модель с конечным числом *состояний*, некоторым алфавитом из символов и переходами – из каждого состояния ведёт переход в какое-то (возможно, в себя же) состояние для каждого символа. Одно из состояний – начальное.

В автомат подают строку по одному символу за раз, и на каждом ходу он переходит в следующее состояние в соответствии с переходом из его текущего состояния.

Некоторые состояния конечного автомата называются *принимающими*, их обычно обводят в кружок.

Определение. Говорится, что автомат M принимает строку w, если после его вычисления над этой строкой он находится в принимающем состоянии.

4. Постройте детерминированный конечный автомат для
(а) строк из букв $\{a, b\}$, содержащих чётное число символов a;
(б) строк из букв $\{a, b\}$, в которых после b всегда идёт a;
(в) целых неотрицательных чисел, поданных в двоичной записи, не делящихся на 8;
(г) целых неотрицательных чисел, поданных в 4-ичной записи, делящихся на 3.

5. Существует ли конечный автомат, который принимает ровно множество строк из $\{a, b\}$, в которых символов a меньше, чем символов b?

6. Можно ли построить DFA для следующих множеств
(а) строк над алфавитом $\{a, b\}$, которые начинаются и заканчиваются на один и тот же символ;
(б) строк вида $a^x b^y$, где $x \leqslant y$ (a^x – символ a, повторенный x раз);
(в) строк из букв $\{a, b\}$, которые являются палиндромами (при записи наоборот получается та же строка).

7. (*) Пусть L_1, L_2 - множества строк, которые принимаются DFA D_1 и D_2 соответственно. Всегда ли есть DFA, который принимает $L_1 \cup L_2 = \{w_1 w_2 \mid w_1 \in L_1, w_2 \in L_2\}$?

Машина Тьюринга

Определение. Машина Тьюринга — это семерка $M = (\Sigma, \Gamma, Q, q_0, \delta, q_{acc}, q_{rej})$, где:

- Σ – алфавит входных символов, из которых состоит начальная строка на ленте;
- Q и Γ — конечные множества состояний и алфавит символов, которые может записать МТ (рабочий алфавит), причем $\Sigma \subset \Gamma$ и $\sqcup \in \Gamma \setminus \Sigma$ ($\sqcup$ – это символ пробела, которого нет входном алфавите, им заполнена вся бесконечная лента, кроме изначальной строки);
- $q_0, q_{acc}, q_{rej} \in Q$ — начальное, принимающее и отвергающее состояния;
- $\delta : (Q \setminus \{q_{acc}, q_{rej}\}) \times \Gamma \to Q \times \Gamma \times \{-1, +1\}$ — функция переходов. Машина с головкой на символе a переходит из q в q_{next}, причем символ a перезаписывается на символ $b \in \Gamma$, а индекс её головки изменяется на $\Delta \in \{\pm 1\}$, где $\delta(q, a) = (q_{next}, b, \Delta)$
- Головка машины Тьюринга изначально находится на первом символе строки. МТ заканчивает работу, если попадает в q_{acc} или q_{rej}. В первом случае она *принимает* строку, во втором - *отвергает* (иначе – *зацикливается*).
- Если M - машина Тьюринга, то для любой строки с символами из Σ можно определить $M(w) = \{$принимается, не принимается, зацикливается.$\}$

8. Напишите полностью со всеми переходами машину Тьюринга, которая принимает строки из 0 и 1, в которых чётное число единиц.

9. Постройте машину Тьюринга, которая принимает число a в унарном виде (1^a), и принимает тогда и только тогда, когда a – степень двойки.

10. Построить машину Тьюринга, которая по каждому входному слову вида $0^n 1^k$, где $n > k > 0$, выдаёт результат 0, а для всех остальных слов результат 1. Оценить время работы построенной машины.

11. Докажите, что есть множество строк, которое не принимается никакой машиной Тьюринга.

Классы сложности

Определение. Множество строк *(язык)* лежит в классе сложности NP, если существует недетерминированная машина Тьюринга, которая принимает этот язык и использует не больше полиномиального времени от размера входа. Аналогично определяются классы P, EXP (полиномиальное и экспоненциальное время для *детерминированной* машины Тьюринга). Так же определяются классы сложности по памяти: $LOGSPACE$ - логарифмическая память, $NLOGSPACE$ - логпамять + недетерминированная МТ, $PSPACE$ - полиномиальная память.

$$\mathsf{L} \subseteq \mathsf{NL} \subseteq \mathsf{P} \subseteq \mathsf{NP} \subseteq \mathsf{PSPACE} \subseteq \mathsf{EXPTIME}$$

Душное определение. Язык K сводится к языку L за полиномиальное время, если существует машина Тьюринга M, которая на любом входе w (не обязательно из K) останавливается за полиномиальное время $poly(|w|)$ и оставляет на выходной ленте $T(w)$, для которой $w \in K \Longleftrightarrow T(w) \in L$.

Определение. Полная задача в классе — задача, которая лежит в данном классе и к которой можно свести любую задачу из этого класса (для каждого класса сведение своё!). Например, можно доказать, что если язык L принадлежит NP или $PSPACE$ и другой язык K можно **полиномиально** свести к L, то K тоже лежит в соответствующем классе. То же верно, если L лежит в NL или P, а сведение с **логарифмической памятью**, поэтому и сведения для данных классов соответствующие.

12. Докажите, что задача о выполнимости булевой формулы в 2-КНФ лежит в классе NLOGSPACE. Верно ли, что она полна в этом классе?

13. Дано множество натуральных чисел S и число x. Нужно определить, содержит ли S некоторое подмножество, сумма элементов которого равна x, все числа подаются в двоичной записи. Докажите, что эта задача NP-полна (Подсказка: попробуйте свести к этой задаче задачу проверки существования в графе вершинного покрытия данного размера).

14. Докажите, что задача проверки наличия гамильтонова пути в данном ориентированном графе NP-полна.