

Zadanie 1 Automat skończony

Niech Σ będzie alfabetem, a ϵ łańcuchem pustym. Zbiór wszystkich słów jakie mogą być utworzone ze znaków alfabetu Σ oznaczamy przez Σ^+ , przez Σ^* natomiast oznaczamy zbiór $\Sigma^+ \cup \{\epsilon\}$.

Deterministyczny automat skończony (DAS) jest piątką $(Q, \Sigma, \delta, s, F)$, gdzie

- Q jest skończonym zbiorem stanów;
- Σ jest alfabetem;
- $\delta: Q \times \Sigma \rightarrow Q$ jest funkcją przejść;
- $s \in Q$ jest stanem początkowym; oraz
- $F \subseteq Q$ jest zbiorem stanów końcowych.

DAS będąc w stanie q_i po przeczytaniu litery a zmienia stan na q_j zgodnie z funkcją przejścia $\delta(q_i, a) = q_j$. Funkcję przejść można rozszerzyć na cały zbiór Σ^* do postaci $\delta: Q \times \Sigma^* \rightarrow Q$ przyjmując:

$$\begin{aligned}\delta(q, \epsilon) &= q \text{ dla } q \in Q \text{ oraz} \\ \delta(q, wa) &= \delta(\delta(q, w), a) \text{ dla } q \in Q, a \in \Sigma \text{ i } w \in \Sigma^*.\end{aligned}$$

Językiem L akceptowanym przez DAS jest zbiór słów:

$$L = \{w \in \Sigma^* : \delta(s, w) \in F\}.$$



Zaimplementuj funkcję

```
bool akceptuje(map<pair<int, char>, int> &delta,
               int s, set<int> &F, string w)
```

sprawdzając czy łańcuch w należy do języka akceptowanego przez DAS zadany funkcją przejść `delta`, zbiorem stanów końcowych F i stanem początkowym s .

Zadanie 2 Drzewo binarne

Poniższy algorytm wypisuje zawartość drzewa binarnego poziomami, od góry (korzenia k) do dołu (liści), od lewej strony do prawej.

WYPISZ(k)

- 1 Zainicjuj pustą kolejkę q
- 2 Wstaw do kolejki q wierzchołek k
- 3 **while** kolejka q nie jest pusta
- 4 **do** $w \leftarrow$ kolejny wierzchołek kolejki q
- 5 **if** $w \neq \text{NULL}$
- 6 **then** Wstaw do kolejki q lewego syna w
- 7 Wstaw do kolejki q prawego syna w
- 8 Wypisz zawartość wierzchołka w
- 9 Usuń kolejny wierzchołek z kolejki q (czyli w)



Zakładając następującą definicję wierzchołka drzewa binarnego:

```
struct node {  
    int data;  
    node* left;  
    node* right;  
}
```

zapisz algorytm WYPISZ w języku C++ z wykorzystaniem odpowiednich kontenerów biblioteki STL. Przyjmij, że korzeń k jest zadeklarowany jako wskaźnik:

```
node* k;
```