JAO - Języki, Automaty i Obliczenia - Wykład 1

Języki formalne są abstrakcyjnie zbiorami słów nad alfabetem skończonym Σ.

Język formalny L to opis pewnego problemu decyzyjnego: słowa to kody instancji (wejścia) problemu, $x \in L$ oznacza, że dla instancji x odpowiedzią jest "tak".

Podstawowe operacje to $\cup$, * oraz konkatenacja (złożenie, dopisanie słowa)

$$L^* = \bigcup_{i=0}^{\infty} L^i, \quad L^+ = \bigcup_{i=1}^{\infty} L^i$$

Przyjmujemy że $L^0 = \varepsilon$, gdzie ε oznacza *słowo puste* (ale niezerowe).

- Wyrażenia regularne standardowe - stałe to słowa (włącznie z pustym), operacje to $\cup$, * oraz konkatenacja, na przykład: zbiór ciągów binarnych zawierających 111 jest wartością wyrażenia:
  $(0 \cup 1)^* \cdot 111 \cdot (0 \cup 1)^*$
- rozszerzone wyrażenia regularne - dodatkowo operacje dopełnienia i przecięcia zbiorów, na przykład zbiór ciągów binarnych nie zawierających 111:
  $(0 \cup 1)^* - (0 \cup 1)^* \cdot 111 \cdot (0 \cup 1)^*$

Wyrażenia regularne są bardzo ubogim narzędziem problemów decyzyjnych, Języki opisywane przez wyrażenia regularne nazywamy **językami regularnymi**.
Zachodzi: Dla każdego języka *unarnego* mamy: $L \subseteq 1^*$ implikuje L^* regularny .

Niech $\Sigma = \{0, 1\}$. Wyrażenie

$$\Sigma^* \cdot 0\,0\,\Sigma^*$$

opisuje język słów zawierających 00.

Wyrażenie $(1 + 10)^*$ opisuje język słów nie zawierających 00 i zaczynających sie od 1.

Wyrażenie

$$(0 + \epsilon) \cdot (1 + 10)^*$$

opisuje język wszystkich słów binarnych nie zawierających 00.

Jakie wyrażenie standardowe opisuje słowa nie zawierające 000 ?

Dla $n \geq 1$ zdefiniujmy słowo α_n, które koduje ciąg kolejnych zapisów binarnych liczb od 0 do $2^n - 1$ zapisanych binarnie, każdy za pomocą n cyfr (ewentualnie na początku dodajemy zera). Kolejne zapisy oddzielamy specjalnym znakiem #.

$$\alpha_n = \# \cdot bin(0) \cdot \# \cdot bin(1)\# \cdot bin(2)\# \ldots \# \cdot bin(2^n - 1) \cdot \#.$$

Lemat

Niech $\Sigma = \{0, 1, \#\}$. Zbiór $\Sigma^* - \{\alpha_n\}$ można opisać standardowym wyrażeniem regularnym o rozmiarze $O(poly(n))$.

Twierdzenie

Istnieje rozszerzone wyrażenie regularne opisujące $\{\alpha_n\}$ używające tylko operacji $\cup, \cdot, ^*$ oraz jednej operacji dopełnienia, względem którego każde standardowe wyrażenie ma rozmiar wykładniczy.

Niech $\Sigma = \{0, 1, \ldots k\}$ oraz niech L_k będzie językiem słów, w których między każdymi dwoma kolejnymi symbolami $i+1$, oraz przed pierwszym z nich, są dwa symbole i (niekoniecznie bezpośrednio).

Niech $\Sigma_j = \Sigma - \{j\}$, oraz

$$L_{(i+1)} = (\Sigma^*_{i+1} \cdot i \cdot \Sigma^*_{i+1} \cdot i \cdot \Sigma^*_{i+1} \cdot i+1 \cdot \Sigma^*_{i+1})^*$$

Wtedy $L_k = \bigcap_{i=0}^{k-1} L_{(i+1)}$

Najkrótsze słowo należące do L_k ma długość wykładniczą. Zatem standardowe wyrażenie regularne też ma długość wykładniczą.

Automaty to abstrakcyjne modele algorytmu decyzyjnego.
Zaletą i jednocześnie wadą automatu jest uproszczony model.
Nieformalny opis automatu skończonego:

- automat *czyta* słowo wejściowe symbol po symbolu,
- ma skończoną liczbę stanów
- język $L(A)$ słów *akceptowanych* przez automat składa sie z ciągów po wczytaniu których automat znajdzie się w stanie *akceptującym*

Formalny opis deterministycznego automatu skończonego:

$$A = (\Sigma, Q, \delta, q_0, F), \text{ gdzie}$$

- Σ jest skończonym alfabetem wejściowym;
- Q jest skończonym zbiorem stanów, $q_0 \in F$ jest stanem początkowym;
- $F \subseteq Q$ jest zbiorem stanów akceptujących, tzn. stanów w których automat *daje* odpowiedź "TAK",
- $\delta : \ Q \times \Sigma \to Q$ - funkcja opisująca działanie automatu A.

Przejście $\delta(q, a) = q'$ oznacza, że automat bedąc w stanie q i czytając kolejny symbol równy a przechodzi do stanu q'.
Zapisujemy to również jako:

$$q \xrightarrow{a} q'$$

Piszemy $q \xrightarrow{w} q'$, dla $w \in \Sigma^*$ gdy automat przechodzi do q' po wczytaniu wszystkich kolejnych liter słowa w.

Język akceptowany przez automat A to

$$L(A) = \{ w : q_0 \xrightarrow{w} q' \in F \}$$

Niech

$$L = \{a^n : n \mod 5 = 3\}$$

Konstruujemy automat:

- $\Sigma = \{a\}$,
- $Q = [0, 4]$,
- $q_0 = 0$,
- $F = \{3\}$,
- $\delta(q, 0) = (q + 1) \mod 5$.

Niech L będzie zbiorem zapisów dziesiętnych liczb podzielnych przez 3. Automat czyta od najmniej znaczącej cyfry. Wtedy

$$Q = \{0, 1, 2\},\ \delta(q, cyfra) = (q + cyfra) \mod 3$$

Niech L będzie zbiorem zapisów binarnych liczb podzielnych przez 3. Automat czyta od najmniej znaczącej cyfry. Zauważmy, że $2^k \mod 3 = 1$ gdy k parzyste, natomiast jest równe -1 gdy k nieparzyste. Automat czyta $a_0 a_1 a_2 \ldots a_n$, gdzie a_0 jest najmniej znaczącą cyfrą w zapisie binarnym. Automat pamięta resztę (modulo 3) oraz dodatkowo pamięta czy kolejny symbol a_i jest dla i parzystego. Zatem automat ma sześć stanów postaci: $(\mathit{reszta}, \mathit{parzyste})$ lub $(\mathit{reszta}, \mathit{nieparzyste})$ dla $i \in [0, 2]$

$$q_0 = (0, \mathit{parzyste}),\ F = \{(0, \mathit{parzyste}), (0, \mathit{nieparzyste})\}$$

$$\delta((\mathit{reszta}, \mathit{parzyste}), a) = (\mathit{reszta} + 1 \mod 3, \mathit{nieparzyste})$$

$$\delta((\mathit{reszta}, \mathit{nieparzyste}), a) = (\mathit{reszta} - 1 \mod 3, \mathit{parzyste})$$

Oznaczmy: $x \preceq y$ jeśli słowo x jest podciągiem y, np.

$$ab \preceq aabb$$

$$rytter \preceq arytotier$$

Relacja $\preceq$ to częściowy porządek.

Lemat

[Lemat Higmana] *Jeśli $|\Sigma| < \infty$ to dla każdego $X \subseteq \Sigma^*$ zbiór $Min(X)$ słów minimalnych (w sensie $\preceq$) należących do X jest skończony.*

Niech $P = \{2, 3, 5, 7, ..\}$ będzie zbiorem liczb pierwszych, a $P^{(k)}$ będzie zbiorem zapisów tych liczb w systemie k-arnym.

$Min(P^{(3)}) = \{2,\ 10,\ 111\}$;

$$Min(P^{(10)}) = \{2, 3, 7, 5, 11, 19, 41, 61, 89, 409, 449, 499, 881,$$

$$991, 6469, 6949, 9001, 9049, 9649, 9949, 60649, 666649,$$

$$946669, 60000049, 66000049, 66600049\}$$

Inaczej mówiąc w każdej liczbie pierwszej możemy usunąć pewną liczbę cyfr i otrzymać liczbę pierwszą ze zbioru $Min(P^{(10)})$.

Przypuśćmy, że teza jest fałszywa. Niech Γ będzie zbiorem nieskończonych ciągów słów $x_1, x_2, ..$ nad alfabetem Σ^* (niektóre słowa x_i mogą być spoza X) spełniających:

$$(*)\ i < j \ \rightarrow \ not\ (x_i \preceq x_j)$$

(ale może być $x_j \prec x_i$). $\Gamma \neq \emptyset$, bo możemy wziąć nieskończony ciąg różnych elementów X, gdyż zakładamy (przez zaprzeczenie), że zbiór słów minimalnych zbioru X jest nieskończony.

Wybierzmy spośród tych ciągów pewien "minimalny" ciąg γ w sensie długości, tzn. $|x_1|$ minimalne, jeśli x_1 ustalone to $|x_2|$ minimalne, jeśli x_1, x_2 ustalone to $|x_3|$ minimalne, itd.

Jeśli Z jest dowolnym zbiorem nieskończonych ciągów to niech $first((x_1, x_2, \ldots x_k), Z)$ będzie zbiorem tych ciągów które zaczynają sie od $x_1, x_2, \ldots x_k$.
Istnieje następujący nieskończony ciąg γ wygenerowany przez:

for $i = 1$ **to** ∞ **do**

x_i:= najkrótsze słowo x_i, takie, że $first((x_1, x_2, \ldots x_i), \Gamma) \neq \emptyset$;

return ciąg $\gamma = (x_1, x_2, x_3, x_4, ..)$.

Wybierzmy podciąg nieskończony $(x_{i_1}, x_{i_2}, x_{i_3}, ..)$ wszystkich słów z ciągu γ zaczynających się na pewną taką samą literę a. Usuńmy tę literę z początku każdego z tych słów otrzymując ciąg $(x'_{i_1}, x'_{i_2}..)$.

Wtedy ciąg $\gamma' = (x_1, x_2, \ldots, x_{i_1-1}, x'_{i_1}, x'_{i_2}, x'_{i_3} \ldots)$
spełnia te same warunki co początkowy ciąg, ale jest "mniejszy" na pozycji x_{i_1}, sprzeczność z definicją γ.

Lemat

Dla dowolnego języka L nad skończonym alfabetem $subs(L)$, $sup(L)$ są regularne, gdzie $subs(L), sup(L)$ oznacza odowiednio zbiór podciągów (ang. subsequences) i nadciągów (ang. supersequences) słów języka L.

Dowód regularnści $sup(L)$. Niech $X := \min(L)$. Z Lematu Higmana wynika, ze X jest skończony. Dla $w = a_1 a_2 \dots a_k \in X$ $Y(w) := \Sigma^* \cdot a_1 \cdot \Sigma^* \cdot a_2 \cdot \Sigma^* \cdot a_3 \dots \Sigma^* \cdot a_k \cdot Sigma^*$

$$sup(L) = \bigcup_{w \in X} Y(w).$$

Dowód regularności $subs(L)$. Niech Z bedzie zbiorem minimalnych słów nie należących do $subs(L)$, wtedy:

$$subs(L) = \Sigma^* - \bigcup_{w \in Z} Y(w).$$