Wyszukiwanie wzorców

Niech $x, w \in \Sigma^*$

$w \sqsubset x \overset{def}{\equiv}$ w jest prefiksem x

$w \sqsupset x$ sufiksem x

Problem

uzorzec tekst

Dane: P, T

Znaleźć wszystkie przesunięcia, z którymi P występuje w T.

Oznaczenia: X_k - k-literowy prefix X

X - słowo

x - litera

x_k - k-ta litera słowa X

$m = |P|$

$n = |T|$

Algorytm naiwny: $O(n \cdot m)$

```
for s ∈ [0..n-m]
    for i ∈ [0..m]
        if t_{s+i} ≠ p_i break
    yield s
```

trochę inaczej oryginalnie

Algorytm Karpa-Rabina

Niech $|\Sigma| = d$

Słowa nad Σ traktujemy jako liczby d-arne

Niech q - l. pierwsza (t. że dq mieści się w słowie maszynowym)

Niech p - liczba odpowiadająca wzorcowi

t_i - — || — $t_i \dots t_{i+m-1}$

Wówczas

$p \neq_{mod\, q} t_i \Rightarrow$ wzorzec nie występuje z przesunięciem i

[pseudokod]

Algorytmy wyszukiwania wzorca automatami skończonymi

$P = aabbabab$

$q_0\ q_1\ q_2\ q_3\ q_4\ q_5\ q_6\ q_7\ q_8$

$T = aaabaabb$

$\langle q_3, b\rangle \to q_4$

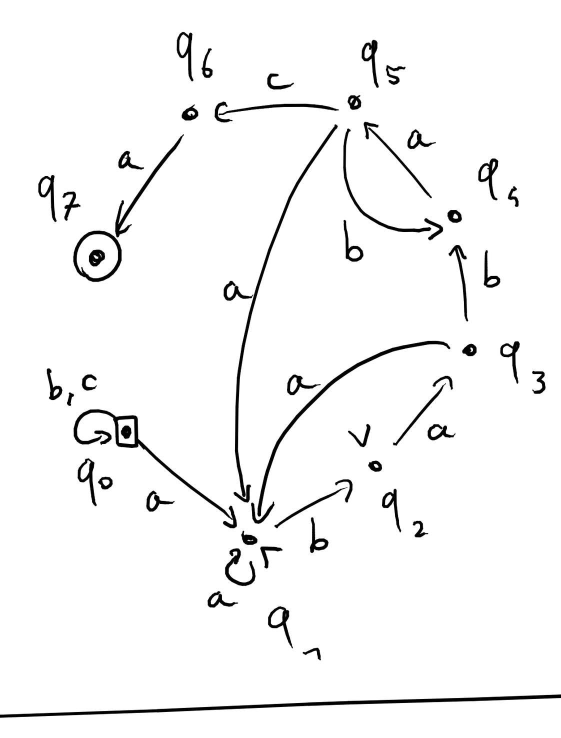
$\langle q_3, a\rangle \to q_1$

Przykład: budujemy automat dla wzorca P:

$P = ababaca$

$\underline{n} = \{0, \dots, n-1\}$

$Q = \underline{8} = \underline{|P|+1}$



⊡ początkowy

⊙ akceptujący

(jeśli nie ma strzałki, to do q_0)

$m = |P|$

Dla wzorca P definiujemy funkcję $\sigma : \Sigma^* \to \underline{m+1}$:

$\sigma(x) = \max\{k \mid P_k \sqsupset x\}$

$\underline{n} = \{0, \dots, n-1\}$

Budujemy automat M_P:

Cel: $\hat{\delta}(q_0, x) = \sigma(x)$

$Q = \underline{m+1}$

$F = \{m\}$

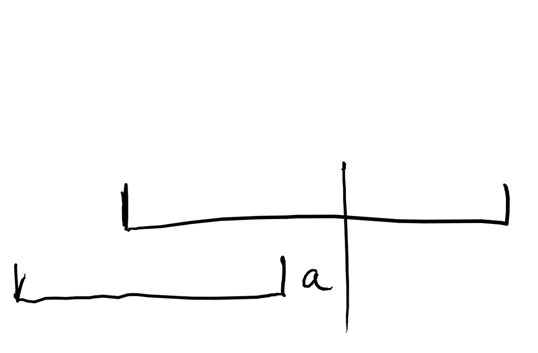
$q_0 = 0$

$\delta(q, a) = \sigma(P_q a)$

Program symulujący M_P:

```
q ← 0
for i ∈ [1..n]
    q ← δ(q, T_i)
    when q = m
        yield i-m
```

Obliczanie δ:



Bardzo naiwnie: $\Theta(m^3 \cdot |\Sigma|)$

Można: $\Theta(m \cdot |\Sigma|)$

Koszt algorytmu: $\underbrace{\text{budowa automatu}}_{\Theta(m \cdot d)} + \Theta(n)$

może być duże

Algorytm KMP (Knuth, Morris, Pratt)

$\delta : Q \times \Sigma \to Q$

$\Pi : Q \to \dots$