

String matching (vyhl'adávanie vzorky v texte)

Vstup: vzorka (pattern) P dĺžky m , text T dĺžky n .

Úloha: Nájdí všetky indexy $\{i_1, i_2, \dots\}$ t.ž. $T[i_j..i_j + m - 1] = P$.

Príklad:

Vstup: $P = \text{ma}$, $T = \text{Ema ma mamu}$

Výstup: 1, 4, 7

Vstup: $P = \text{"a ma"}$, $T = \text{Ema ma mamu}$

Výstup: 2, 5

Triviálny algoritmus

```
1  for ( i = 0; i <= n - m; i ++ ) {  
2      j = 0;  
3      while ( j < m && P[ j ] == T[ i + j ] ) { // (*)  
4          j ++;  
5      }  
6      if ( j == m ) {  
7          print ( i );  
8      }  
9  }
```

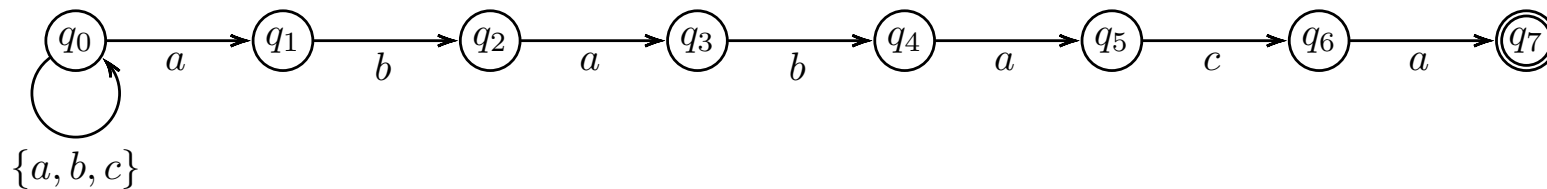
Najhorší prípad: $O(mn)$, $P = a^m$, $T = a^n$

Priemerný prípad: $O(m + n)$

Cieľ: algoritmus s najhorším prípadom $O(m + n)$

Nedeterministický konečný automat pre $\{xP \mid x \in \Sigma^*\}$

$P = \text{ababaca}$

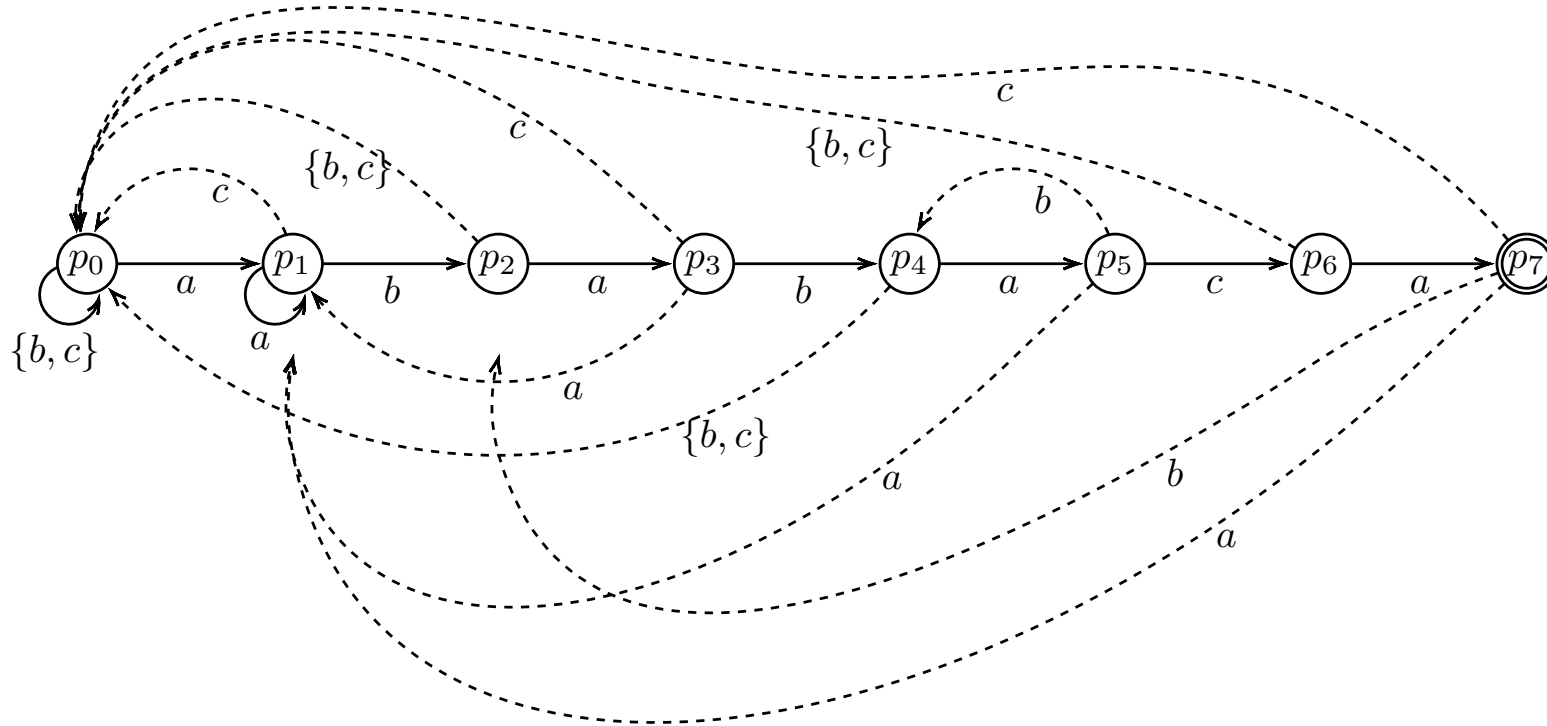


Automat vie dosiahnuť q_i po prečítaní T
práve vtedy keď sufix T dĺžky i je prefixom P .

Počet stavov: $m + 1$

Simulácia na reťazci T v čase $O(mn)$.

Deterministický konečný automat pre $\{xP \mid x \in \Sigma^*\}$



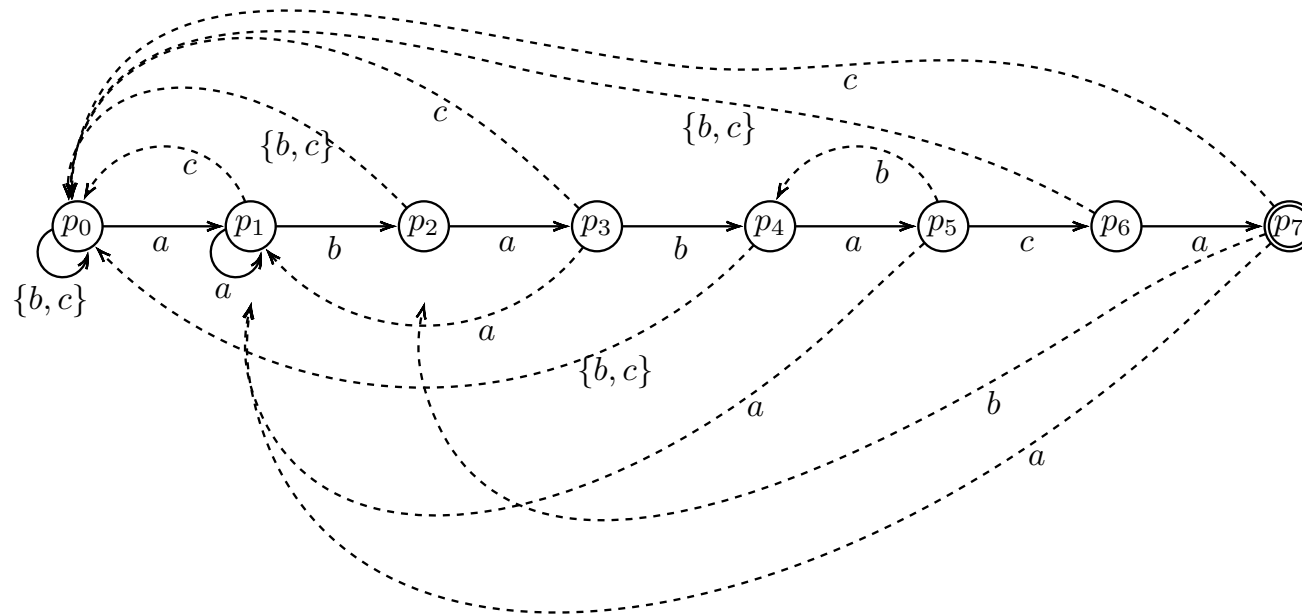
Automat skončí v stave p_i po prečítaní T

práve vtedy keď i je dĺžka najdlhšieho sufixu T , ktorý je prefixom P

$\Rightarrow$ automat skončí v p_m práve vtedy, keď T končí na P

Čítaj T znak po znaku, nájdi nový stav, vypíš výskyt ak stav je p_m

Deterministický konečný automat pre $\{xP \mid x \in \Sigma^*\}$



Počet stavov: $m + 1$

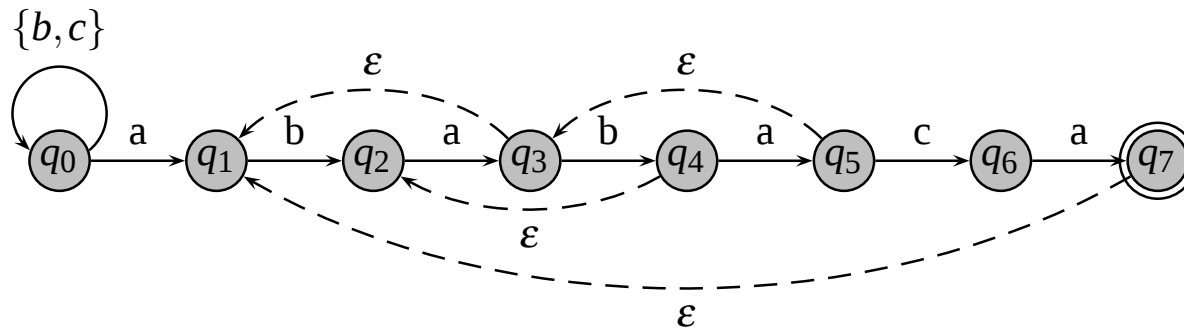
Simulácia na texte T : $O(n)$

Pamäť: $O(m\sigma)$

Vytváranie automatu: $O(m\sigma)$ (ukážeme neskôr)

Spolu $O(n + m\sigma)$

Morrisov-Prattov algoritmus 1970 $O(m + n)$



Epsilonový prechod sa použije keď nie je iná možnosť

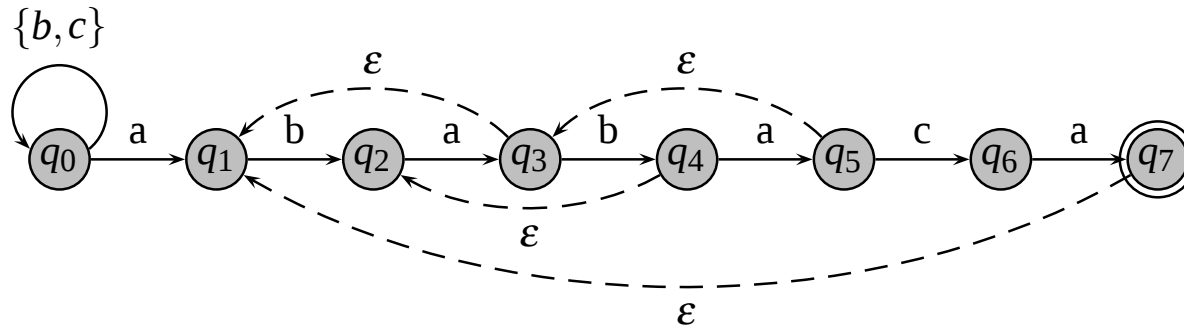
$$\delta(q_i, \varepsilon) = q_{sp[i]} \quad (sp[i] < i)$$

$sp[i]$ = dĺžka najdlhšieho vlastného sufixu $P[0..i - 1]$, ktorý je prefix P .

Automat po prečítaní T skončí v stave q_i ,

kde i je dĺžka najdlhšieho sufixu T , ktorý je prefixom P .

Morrisov-Prattov algoritmus



$i, sp[i], sp[sp[i]], \dots, 0$ je spájaný zoznam všetkých sufixov $P[0 \dots i - 1]$, ktoré sú prefixami P .

Napr. $i = 5, P[0..4] = ababa$

Zoznam q_5, q_3, q_1, q_0

Suffixy/prefixy $ababa, aba, a, \epsilon$

Spracovanie ďalšieho znaku x : nájdí najdlhší z prefixov, za ktorým ide x :

$x = a$, stav $q_0 \rightarrow q_1$; $x = b$, stav $q_3 \rightarrow q_4$; $x = c$, stav $q_5 \rightarrow q_6$

Morrisov-Prattov algoritmus

```
1  state=0;
2  for(i=0; i<n; i++){
3      while(state>0 && T[i]!=P[state]){ //epsilon transitions
4          state=sp[state];
5      }
6      if(T[i]==P[state]){ //move to next state
7          state++;
8      }
9      if(state==m){ //accepting state — print occurrence
10         print i-m+1; state=sp[state];
11     }
12 }
```

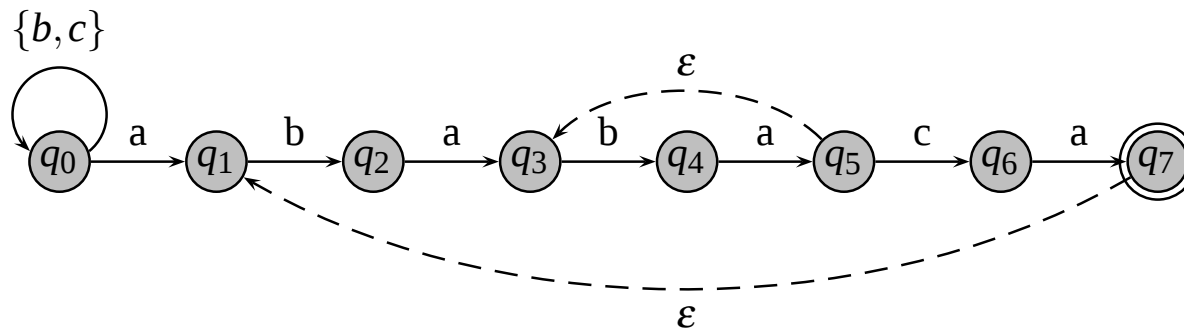
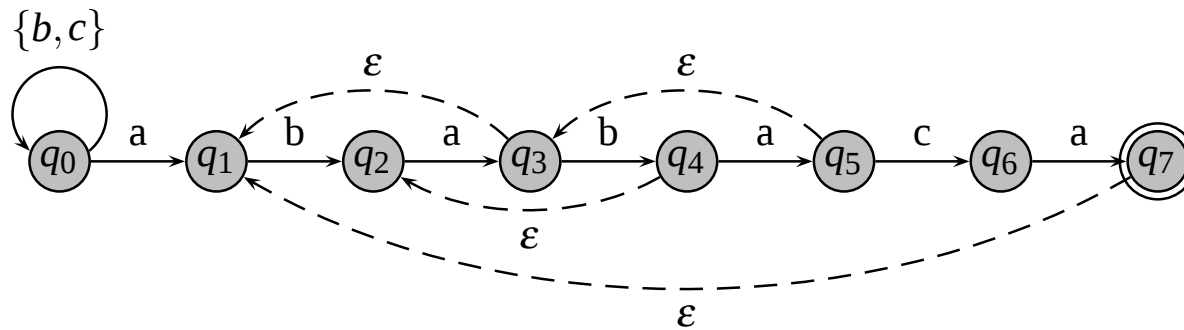
Morrisov-Prattov algoritmus – predspracovanie

```
1  sp[0]=sp[1]=0;
2  j=0;
3  for(i=2; i<=m; i++){
4      // invariant: j=sp[i-1];
5      while(j>0 && P[i-1]!=P[j]) {
6          j=sp[j];
7      }
8      if(P[i-1]==P[j]) {
9          j++;
10     }
11     sp[i]=j;
12 }
```

Morrisov-Prattov algoritmus, 1970

- Vybuduj tabuľku sp v čase $O(m)$, pamäti $O(m)$
- Simuluj automat na T v čase $O(n)$
- Funguje aj pre veľké abecedy,
ku znakom pristupuje iba pomocou porovnania na rovnosť
- Text T môže čítať a spracovávať znak po znaku
ale spracovanie jedného konkrétneho znaku môže trvať až $O(m)$

Knuthov-Morrisov-Prattov algoritmus, 1977



V automate dlhšie epsilonové prechody:

$\delta(q_i, \varepsilon) = q_j$, kde j je dĺžka najdlhšieho vlastného sufixu $P[0..i - 1]$, ktorý je prefix P a $P[j] \neq P[i]$

Knuthov-Morrisov-Prattov algoritmus, predspracovanie

Máme spočítané hodnoty $sp[i]$ z MP algoritmu, chceme $sp_2[i]$ z KMP

```
1  sp2[0]=0;
2  for(i=1; i<=m; i++) {
3      if(i==m || P[sp[i]]!=P[i]) {
4          sp2[i]=sp[i];
5      }
6      else {
7          sp2[i]=sp2[sp[i]];
8      }
9  }
```

Knuthov-Morrisov-Prattov algoritmus, analýza

Veta: Počet ε prechodov v jednej iterácii KMP je navyše $\log_{\phi}(m + 1)$, kde $\phi = \frac{1+\sqrt{5}}{2}$ je zlatý rez.

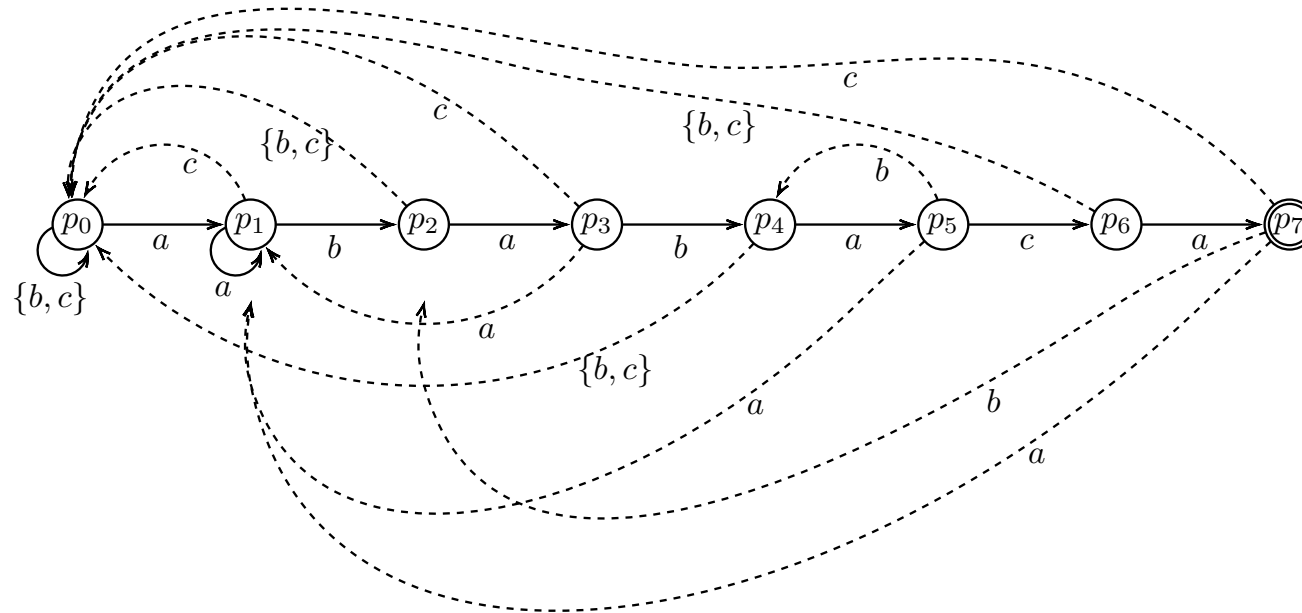
Def. Celé číslo x nazývame periódou slova S ak $1 \leq x \leq m$ a pre každé $i \in \{0, 1, \dots, |S| - x - 1\}$ platí $S[i] = S[i + x]$.

Lema 1: Ak p a q sú periódy slova S a platí, že $p < q$ a $p + q \leq |S|$, tak aj $q - p$ je perióda S .

Lema 2: Nech $j' = \delta'(j, \varepsilon)$ a $j'' = \delta'(j', \varepsilon)$. Potom $j \geq j' + j'' + 1$.

Cvičenie

Spočítajte DFA v čase $O(m\sigma)$ použitím tabuľky $sp[i]$ z MP alg.



Automat skončí v stave p_i po prečítaní T

práve vtedy keď i je dĺžka najdlhšieho sufixu T , ktorý je prefixom P