Vyhledávání v textu:

Σ := abeceda -> množina všech znaků,
chceme její konečnost i konstantnost
zároveň i rozumně malou, abychom mohli indexovat

Σ^* := řetězce -> množina všech řetězců

α, β, γ ... řetězce

a, b, c ... znaky

$|\alpha|$ ... délka řetězce

ε ... prázdný řetězec

$\alpha\beta$ ... zřetězení

$\alpha[i]$ ... i-tý znak od nuly

$\alpha[i:j]$ ... podřetězec (i tam je, j tam není)

$\alpha[:j]$ ... prefix

$\alpha[i:]$ ... suffix

Každý podřetězec je prefixem suffixu.

Problém: dostanu seno σ, $S := |\sigma|$ -> což je vstupní řetězec
jehlu ι, $J := |\iota|$ -> což je hledaný řetězec
výstup: $\{i \mid \sigma[i:i+J] = \iota\}$

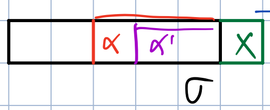
Triviální přístup:

$\Theta(S \cdot J)$ -> zkusím každý začátek a porovnám s jehlou následující

Nefunkční: $\Theta(S)$? Vstup: uououos, hledám uouos
Postupně prohledávám, když nenajdu shodu z hledaným, prostě zahodím.
Zbyde mi jen uos, takže nic nenajdu, přestože text uouos obsahuje.

Inkrementální algoritmus:

Postupně čtu vstup a podle toho dopočítávám výstup -> Pokud αx není prefixem jehly, zkracujeme α na α' tak dlouho, než $\alpha' x$ je prefixem jehly



stav α := nejdelší prefix jehly, který je suffixem sena

<-> Jehly při čtení uouo|uo|s| -> zde uouo; -> zde uouos; -> zde bude uou; zde bude uouo (uouou už není prefix jehly)

nový stav -> ε (prázdný řetězec)

-> $\alpha' x$ suffix σx
-> α' suffix σ
-> $\alpha' x$ prefix jehly -> α' prefix jehly

} z toho plyne: $|\alpha'| \leq |\alpha|$
α' je suffixem α

Předvýpočet posunutí:

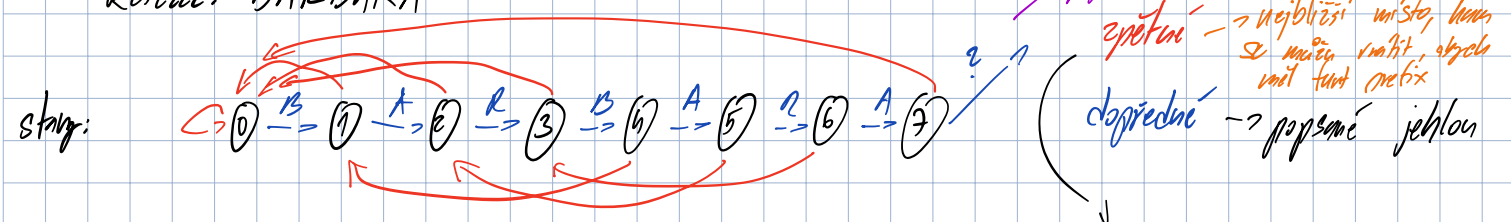
$\alpha' \neq \alpha$

Pro stav $\alpha := z(\alpha) :=$ nejdelší α' vlastní sufix α, který je prefixem jehly

KMP algoritmus (Knuth, Morris, Pratt):

Lze popsat pomocí vyhledávajícího automatu.

Řetězec: BARBARA



ε B BA BAR BARB BARBA BARBAR BARBARA

pole $z[0 \ldots 7]$

KMP:

Vrať (i, x):

Dokud jehla$[i] \neq x$:

Jeli $i = 0$: vrátíme 0

$i = z[i]$

Vrátíme $i+1$

Hledám jehly: Rychlost?:

Hledej (σ)

1. $i = 0$
2. Pro $j = 0 \ldots |\sigma| - 1$:
3. $i =$ Vrať $(i, \sigma[j])$
4. Dokud $i = |$jehla$|$:
5. Ohlásím výskyt na $j - |\text{jehla}| + 1$.

Lemma: Hledej (σ) má složitost $\Theta(|\text{sena}|)$.

Dk: Hledej je zjevně lineární, pouze vrať $(i;x)$ může být problém.

dopředných hran je $\leq |\text{sena}|$

zpětných $\leq$ # dopředných $\leq |\text{sena}|$

Celkem tedy nejvíce $2|\sigma|$ hran projde.

Celý algoritmus je tedy $\Theta(|\text{sena}|)$

Dokud α je stav a spustíme automat na vstup $\alpha[1:]$, skončíme ve stavu $z(\alpha)$ (nejdelší vlastní sufix)

- postupně najdu jiný sufix než celou α
- zpětné hrany se dají konstruovat až za chodu → všechny předchozí už hodně vytvořené.

Spustím na: A, AR, ARB, ARBA, ARBAR, ARBARA

stačí pustit jen tohle → ARBARA

Konstrukce automatu:

1) $z[0] = 0,\ z[1] = 0$
2) $i = 0$
3) Pro $j = 2, \dots, J$:
4) $\quad i = \text{Hrob}(i, \text{jehla}[j-1])$
5) $\quad z[j] = i$

složitost Hledej (σ)

$\Theta(J) + \Theta(S)$

V: KMP najde všechny výskyty v čase $\Theta(J+S)$.

Důkaz nahoře.

Obecněji:

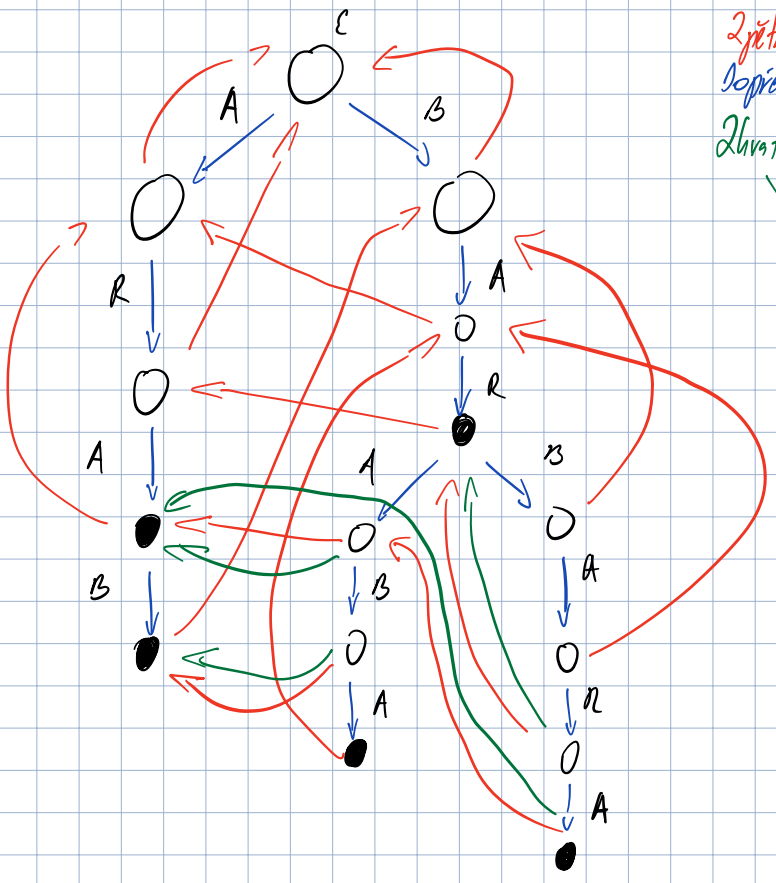
Jehly: $\iota_1 \dots \iota_n$
délek: $J_1 \dots J_n$
Seno σ délky S

Chceme: $\{(i,j) \mid \iota_j = \sigma[j : j + |\iota_i|]\}$

čas $\Theta(S + \sum_i J_i + V)$ → # výskytů

Jehly:

ARA
BAR
ARAB
BARBARA
BARABA



Zpětná
Dopředná
Zkratka
↳ do nejbližšího hranového bodu po zpětných hranách