Sufixové pole

DS pro textové alg.: zpracování řetězců, pattern matching, bioinformatika, ...

vyhledávání podřetězce β, $m = |\beta|$ v řetězci α, $n = |\alpha|$:

- Rabin-Karp (rolující hashování) ... $O(n \cdot m)$ čas
- Knuth-Morris-Pratt
- Aho-Corasick

$O(n+m)$ } vhodné pro pevné β

Co když α je pevné? Uvidíme: lze $O(\log n + m)$ (počet # výskytů na výstupu)

Značení

$\alpha = \alpha[0] \dots \alpha[n-1]$ řetězec

$\alpha[i:j] = \alpha[i] \dots \alpha[j-1]$ podřetězec délky $j-i$, pro $j \le i$ prázdný řetězec ε

$\alpha[:j]$ ... prefix délky j

$\alpha[i:]$ ... sufix začínající $\alpha[i]$ délky $n-i$

$\alpha \le \beta$ ... lex. uspořádání: α prefix β nebo existuje i t.ž. $\alpha[:i] = \beta[:i]$ a $\alpha[i] < \beta[i]$

(„α prefix β“: lze vyhnout pomocí $\alpha\$$, $\beta\$$)

Př.: $\alpha =$ bananas, $n = 7$

(pozice 0 1 2 3 4 5 6 7)

i	sufix	$S[i]$	$R[i]$	$L[i]$
0	ε	7	4	0
1	ananas	1	1	3
2	anas	3	5	1
3	as	5	2	0
4	bananas	0	6	0
5	nanas	2	3	2
6	nas	4	7	0
7	s	6	0	–

Pozorování: pro lib. β, výskyty β v α tvoří interval ve S

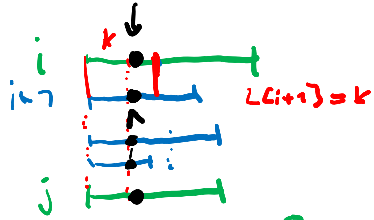
$\Rightarrow$ pomocí bin. vyhledávání lze najít všechny výskyty β v α v čase $O(m \log n + p)$, kde p = # výskytů

Def.: <u>Sufixové pole</u> pro α je permutace $S[0..n]$ t.ž. $\alpha[S[i]:] < \alpha[S[i+1]:]$ pro každé $0 \le i < n$, tj. $S[i]$ je začátek i-tého lex. nejmenšího sufixu α.

- <u>Rankové pole</u> $R[0..n]$ je inverzní permutace k S, tj. $R[i]$ = kolikátý lex. nejmenší sufix je $\alpha[i:]$ (zjistit v $O(n)$ ze S)
- <u>LCP pole</u> $L[0..n-1]$, $L[i] = LCP(\alpha[S[i]:], \alpha[S[i+1]:])$, tj. délka nejdelšího společného prefixu

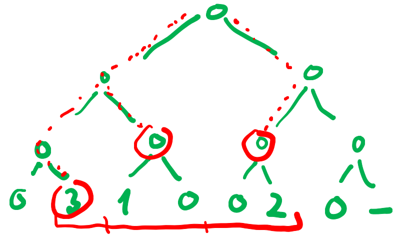
$LCP(\gamma, \delta) = \max k$ s.t. $\gamma[:k] = \delta[:k]$

LCP pro lib. dva sufixy



$LCP(\alpha[S[i]:], \alpha[S[j]:]) = \min(L[i], \dots, L[j-1])$

LCP ≥ k triviálně, LCP ≤ k neboť posloupnost písmen na pozici k+1 je neklesající, nikde neroste

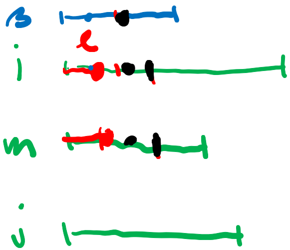


strom pro intervalové dotazy

⇒ v čase $O(\log n)$ (dotaz)

BUILD v $O(n)$

Idea: Zodpovědět LCP pro lib. dva sufixy, lze nalézt lex. nejm. sufix α, který má prefix β v čase $O(m + \log n)$.



aktuální interval $[i,j]$, $\ell = LCP(\beta, \alpha[S[i]:])$

$m := \lfloor (i+j)/2 \rfloor$

if $\ell > LCP(\alpha[S[i]:], \alpha[S[m]:])$ continue $[i,m]$

$<$ $[m,j]$

$=$ porovnej β a $\alpha[S[m]:]$ od pozice ℓ ($\ell+1$ písmeno)

⇒ všechny výskyty β v α lze nalézt potom v čase $O(m + \log n)$.

Dokonce, pro lib. m $LCP(i,m)$ a $LCP(m,j)$ potřebujeme pro nejvýš jedno i a nejvýš jedno j. ("LCP-LR" pole lze v čase $O(n)$)

Aplikace LCP pole

1) histogram pro k-gramy = # výskytů pro každý podřetězec délky k (k-gram) v čase $O(n)$

Př: $\alpha =$ bananas, $n = 7$ (indexy 0 1 2 3 4 5 6 7)

i	sufix	S[i]	R[i]	L[i]
0	ε	7	4	0
1	ananas	1	1	3
2	anas	3	5	1
3	as	5	2	0
4	bananas	0	6	0
5	nanas	2	3	2
6	nas	4	7	0
7	s	6	0	–

$k = 2$

an	2
as	1
ba	1
na	2

interval $[i,j]$ t.ž.

$L[i-1] < k$, $L[\ell] \geq k$ $\forall\, i \leq \ell < j$

$L[j] < k$

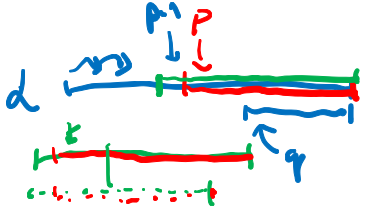
2) nejdelší společný podřetězec = $\max_i L[i]$ v čase $O(n)$

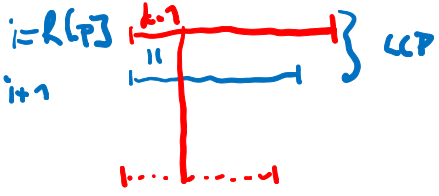
3) nejdelší společný podřetězec v α, β = $\max$ $L[i]$ i, $S[i]$ a $S[i+1]$ z α / β

LCP po $\alpha \# \beta$

konstrukce LCP-pole (Kasai)

Mějme $S, R \rightsquigarrow L$ v čase $O(n)$





☝: if $k = L[R[p-1]]$, pak $L[R[p]] \geq k-1$

BUILD LCP

```
k := 0
for p = 0 ... n-1
    k ← max(0, k-1)
    i ← R[p]
    q ← S[i+1]
    while (p+k<n) ∧ (q+k<n)
          ∧ (α[p+k] = α[q+k])
        k ← k+1
    L[i] ← k
```

☝: mimo while ... $O(n)$ čas

přičtení 1 $\leq$ # odečtení 1 + hodnota konečná $\leq n$ − počáteční hodnota ≥ 0 $\leq n$

$\Rightarrow$ celkem $O(n)$

konstrukce suf. pole · $O(n \log n)$

Def: relace p, σ: $p \equiv_k \sigma$ platí $p[:k] = \sigma[:k]$

$p \leq_k \sigma$ $\quad p[:k] \leq \sigma[:k]$ (↑ lex.)

$R_k[i]$ = # navzájem různých podřetězců délky k až i

$\alpha[i:] \leq_{2k} \alpha[j:] \iff \alpha[i:] <_k \alpha[j:]$ nebo $(\alpha[i:] \equiv_k \alpha[j:]$ a $\alpha[i+k:] \leq_k \alpha[j+k:])$

$R_{2k}[i] \leq R_{2k}[j] \iff R_k[i] < R_k[j]$ nebo $(R_k[i] = R_k[j]$ a $R_k[i+k] \leq R_k[j+k])$

$\iff (R_k[i], R_k[i+k]) \leq_{lex} (R_k[j], R_k[j+k])$

Alg: $k = 2^0$ S_k, R_k podle prvního písmena $O(n \log n)$

$k \to 2k$ $\quad$ bucket sort $(R_k[i], R_k[i+k])$ … 2 průchody $O(n)$

$\Rightarrow$ celkem $O(n \log n)$