Haladó informatikai algoritmusok

Mintaillesztés
Véges determinisztikus automata használata

Fogalmak és jelölések

- Legyen Σ véges jelkészlet, ábécé $\Rightarrow$ **betűk**
- Jelölje Σ^* a Σ elemeiből képezhető összes véges jelsorozat halmazát, beleértve az üres sorozatot is, amit λ-val jelölünk.
- $\Sigma^* = \{x_1 \dots x_n : x_i \in \Sigma, i = 1, \dots, n\} \bigcup \{\lambda\} \Rightarrow$**szavak**
- $X \in \Sigma^*$ szó hossza az $X = x_1 \dots x_n$ sorozat elemeinek a száma, jele $|X| = n$.
- A továbbiakban feltételezzük, hogy a szavakat tömb ábrázolja, tehát az X szó i-edik elemére az $X[i]$ tömbelem-kiválasztással hivatkozunk.

Fogalmak és jelölések

- $x = x_1 \dots x_m$, $y = y_1 \dots y_n \in \Sigma^*$ szavak **konkatenációja**: $xy = x_1 \dots x_m \cdot y_1 \dots y_n$
- Az u szó kezdő szelete, vagy **prefixe** v-nek, jele $u \sqsubset v$, ha $\exists w \in \Sigma^*$, hogy $uw = v$
- Az u szó végződése, vagy **szuffixe** v-nek, jele $u \sqsupset v$, ha $\exists w \in \Sigma^*$, hogy $wu = v$
- Az $X \in \Sigma^*$ szó első i betűjéből álló prefixére az $X_i = X[1..i]$ jelölést használjuk.
- Általában, egy $X = x_1 \dots x_n \in \Sigma^*$ szó és $1 \leq i \leq j \leq |X|$ esetén $X_{i,j} = X[i..j] = x_i \dots x_j$ jelölést használjuk.
- Azt mondjuk, hogy az S szó **előfordul** i **eltolással** az A szóban (S i eltolással **illeszkedik** az A-ra), ha $A[i+1..i+m] = S$, ahol $m = |S|$.

Mintaillesztési probléma

Bemenet: $A, S \subseteq \Sigma^*$, A a szöveg, S minta
Kimenet: A legkisebb (összes) olyan i, amelyre $A[i+1..i+m] = S$, ahol $m = |S|$ a minta hossza

Naiv algoritmus:

1. $n := |A|$;
2. $m := |S|$;
3. **for** $i := 0$ **to** $n - m$ **do**
4. **if** $A[i+1..i+m] = S[1..m]$ **then begin**
5. $WriteLn(i+1)$; **Exit**
6. **End**;

A naiv algoritmus futási ideje: $T_{lr}(n, m) = O(n \cdot m)$

Mintaillesztés véges determinisztikus automatával

Véges determinisztikus automata olyan $M = (Q, q_0, F, \Sigma, \delta)$ rendezett ötös, ahol

- Q véges halmaz, az **állapotok halmaza**,
- $q_0 \in Q$ a **kezdőállapot**,
- $F \subseteq Q$ a **végállapotok** (elfogadó állapotok) halmaza,
- Σ véges halmaz, a **bemeneti jelek** halmaza,
- $\delta : Q \times \Sigma \to Q$, az automata **átmenetfüggvénye**

Jelölje $\delta^* : Q \times \Sigma^* \to Q$ a δ átmentfüggvény kiterjesztését szavakra:

$$\delta^*(q, w) = \begin{cases} q, \text{ ha } w = \lambda, \\ \delta(\delta^*(q, x_1 \dots x_{n-1}), x_n), \text{ ha } w = x_1 \dots x_n \end{cases}$$

Automata működése - algoritmus

Legyen $M = (Q, q_0, F, \Sigma, \delta)$ olyan véges determinisztikus automata, amely a $\Sigma^* S$ nyelvet ismeri fel, azaz $L(M) = \Sigma^* S$. Mivel $\Sigma^* S$ az összes olyan szavak halmaza, amelyek az S mintára végződnek, ezért $A_i \in L(M)$ akkor és csak akkor, ha az S minta $i - m$ eltolással előfordul az A szövegben. $\Rightarrow$

Ha van olyan $M = (Q, q_0, F, \Sigma, \delta)$ véges determinisztikus automatánk, amely a $\Sigma^* S$ nyelvet ismeri fel, azaz $L(M) = \Sigma^* S$, akkor az algoritmus mintaillesztést valósít meg:

1. $q := 0$;
2. $n := |A|$;
3. $m := |S|$;
4. **for** $i := 1$ **to** n **do begin**
5. $q := \delta(q, A[i])$;
6. if $q \in F$ then begin $\{S = A[i - m + 1..i]\}$
7. $WriteLn(i - m + 1)$;
8. Exit;
9. **end**;
10. **end**;

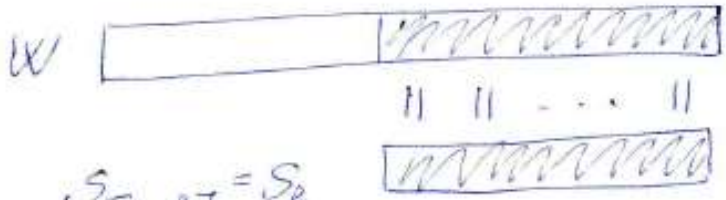
Az algoritmus futási ideje, eltekintve az automata előállítási költségétől $O(n)$.

Átmenetfüggvény

Hogyan és mekkora költséggel lehet előállítani az M automatát (átmenetfüggvényét)?
Legyen $\sigma : \Sigma^* \to \{0, 1, \ldots, m\}$ az S minta **szuffix-függvénye**

$$\sigma(W) = max\{k : S_k \sqsupset W\}$$

Az S minta akkor és csak akkor illeszkedik az A szövegre i eltolással, ha $\sigma(A_{i+m}) = m$.



$S_{[1..k]} = S_k$

S_k szuffixe W-nek, ha $\exists w \in \Sigma^*$, $w.S_k = W$

↑ keressük a legnagyobb k-t

pl: W = a b a a c b a b a a

a b a a b . .

$S_k \Rightarrow k = 4$

Lemma 1

Ha $q = \sigma(W)$, akkor $\sigma(Wx) = \sigma(S_q x)$ minden $W \in \Sigma^*$ és $x \in \Sigma$-ra.

Bizonyítás:

Legyen $u = \sigma(Wx)$ és $r = \sigma(S_q x) \Rightarrow$

a legnagyobb olyan u, amelyre $S_u \sqsupset Wx \Rightarrow$

$S_{u-1} \sqsupset W \Rightarrow$

σ definíciója miatt $u - 1 \leq q \Rightarrow$

$S_{u-1} \sqsupset S_q \Rightarrow$

$S_u = S_{u-1}x \sqsupset S_q x$

de r a legnagyobb olyan index, hogy $S_r \sqsupset S_q x$, tehát $u \leq r$.

Másrészt $S_r \sqsupset S_q x \sqsupset Wx \Rightarrow r \leq u$.

$\Downarrow$

$u = r$

Mire van szükségünk

Olyan automatát akarunk szerkeszteni, amelynek δ átmenetfüggvénye a σ függvényt számítja:

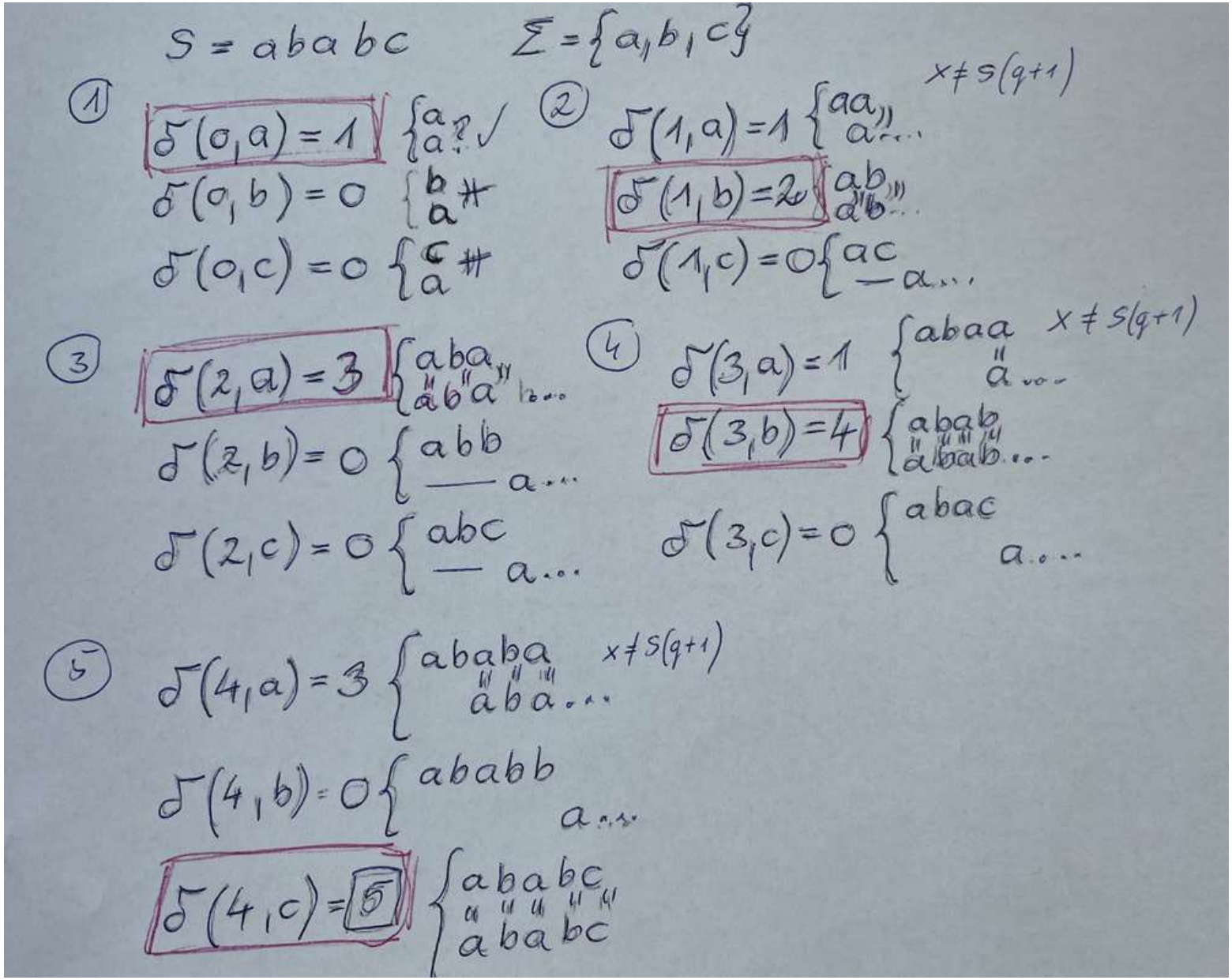
$$\delta^*(0, W) = \sigma(W)$$

Legyen $Q = \{0, 1, \ldots, m\}$, ahol $m = |S|$.
Minden $0 \leq q \leq m$ állapotra és $x \in \Sigma$ jelre:

$$\delta(q, x) = \begin{cases} q+1, \text{ ha } x = S[q+1], \\ max\{k : S_k \sqsupset S_q x\}, \text{ ha } x \neq S[q+1] \end{cases}$$

Nyilvánvaló, hogy $\delta(q, x) = \sigma(S_q x)$.

Példa:


S = ababc
Σ = {a, b, c}
x ≠ s(q+1)
① δ(0,a) = 1 {a a?✓
δ(0,b) = 0 {b #
a
δ(0,c) = 0 {c #
a
② δ(1,a) = 1 {aa
a...
δ(1,b) = 2 {ab
ab...
δ(1,c) = 0 {ac
— a...
③ δ(2,a) = 3 {aba
aba...
δ(2,b) = 0 {abb
— a...
δ(2,c) = 0 {abc
— a...
④ δ(3,a) = 1 {abaa
a...
δ(3,b) = 4 {abab
ababab...
δ(3,c) = 0 {abac
a...
⑤ δ(4,a) = 3 {ababa
aba...
δ(4,b) = 0 {ababb
a...
δ(4,c) = 5 {ababc
ababc

Lemma 2

Minden $W \in \Sigma^*$ szóra $\delta^*(0, W) = \sigma(W)$.

Bizonyítás:

Bizonyítás W hossza szerinti indukcióval.

$W = \lambda$ esetén $\delta^*(0, W) = 0 = \sigma(W)$.

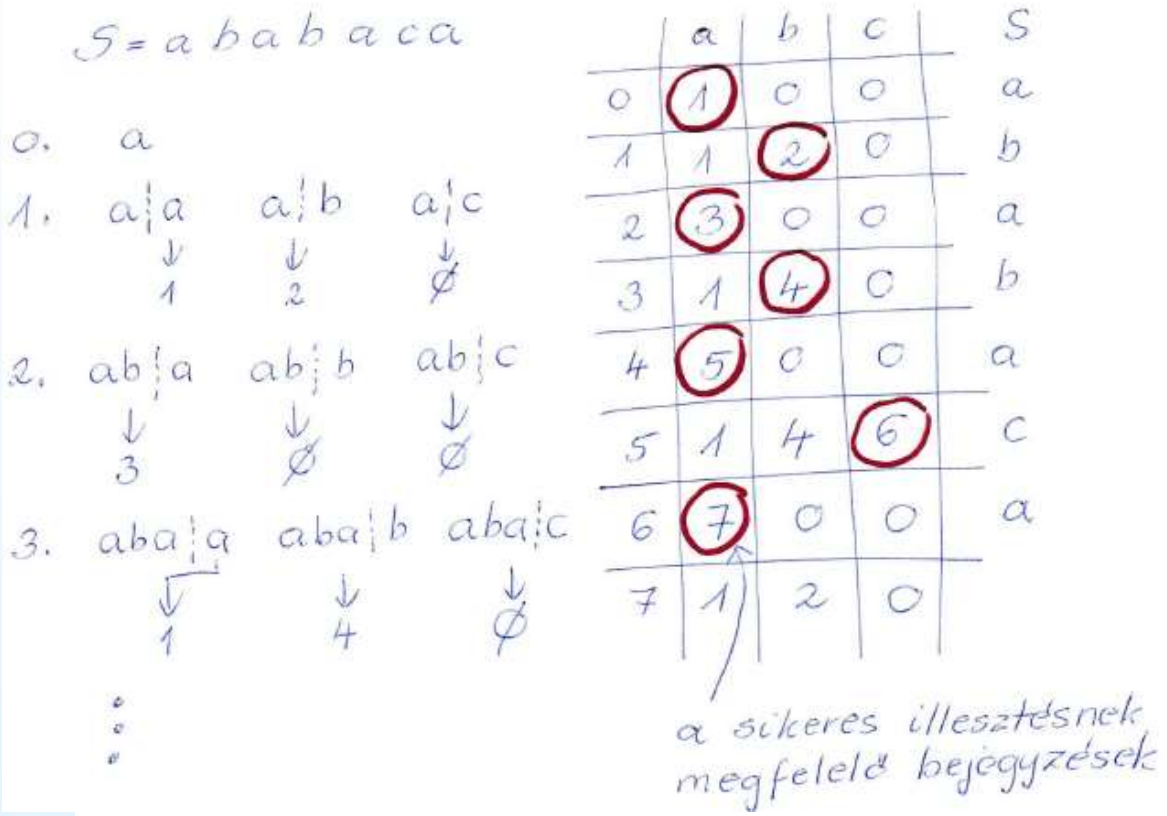
Legyen $W = Vx$, $V \in \Sigma^*$, $x \in \Sigma$ és tegyük fel, hogy $\delta^*(0, V) = \sigma(V) = q$.

$$\begin{aligned}\delta^*(0, Vx) &= \delta(\delta^*(0, V), x) \\ &= \delta(\sigma(V), x) \\ &= \delta(q, x) \\ &= \sigma(S_q x) \\ &= \sigma(Vx)\end{aligned}$$

Az átmenetfüggvény kiszámítása

```
1.  m := |S| ;
2.  for q := 0 to m do
3.     for x ∈ Σ do begin
4.        k := min(m + 1, q + 2);
5.        repeat
6.           k := k − 1;
7.        until S_k ⊐ S_q x ;
8.     δ(q, x) := k;
9.     end;
10. end;
```

Az algoritmus futási ideje $O(m^3|\Sigma|)$.



	a	b	c	S
0	1	0	0	a
1	1	2	0	b
2	3	0	0	a
3	1	4	0	b
4	5	0	0	a
5	1	4	6	c
6	7	0	0	a
7	1	2	0	

Az automata működése:

A szöveg összes $A[i]$ jele alatt feltüntettük az automata $\sigma(A_i)$ állapotát, amelybe az A_i prefix feldolgozása után kerül.

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$A[i]$	a	b	a	b	a	b	a	c	a	b	a	b	a	c	a
$\sigma(A_i)$	0	1	2	3	4	5	4	5	6	7	2	3	4	5	6	7

$\max\{k : S_k \sqsupset A_i\}$

A minta 2-szer fordul elő a szövegben.

Az átmenetfüggvény kiszámítására szolgáló eljárás működése:

$S = abababaca$ $\quad \Sigma = \{a, b, c\}$ $\quad m = |S| = 7$

1. $q := \phi$

$x := a \quad k := \min(8, 2) = 2$ (m+1, q+2)

$k = 1$

$S_1 \,\square\, S_0 x$? $\quad \delta(0, a) := 1$

$a \,\square\, a$ ✓

$x := b \quad k := \min(8, 2) = 2$

$k = 1$ $\quad k = 0$

$S_1 \,\square\, S_0 b \quad S_0 \,\square\, b$ ✓

$a \,\square\, b$ ✗ $\quad \delta(0, b) := 0$

$x = c \quad k = 0$

$\delta(0, c) := 0$

2. $q := 1$

$x = a \quad k := \min(8, 3) = 3$

$k = 2$

$S_2 \,\square\, S_1 a$

$ab \,\square\, aa$ ✗

$k = 1$

$S_1 \,\square\, S_1 a \quad \delta(1, a) := 1$

$a \,\square\, aa$

$x = b \quad k := 3$

$k = 2$

$S_2 \,\square\, S_1 b$

$ab \,\square\, ab$ ✓ $\quad \delta(1, b) := 2$

$x := c \quad k := 3$

$k := 2$

$S_2 \,\square\, S_1 c$

$ab \,\square\, ac$ ✗

$k := 1$

$S_1 \,\square\, S_1 c$

$a \,\square\, ac$ ✗ $\quad \delta(1, c) := 0$

3. $q := 2$

$x := a \quad k := \min(8, 4) = 4$

$k := 3$

$S_3 \,\square\, S_2 a$

$aba \,\square\, aba$ ✓ $\quad \delta(2, a) := 3$

$x := b \quad k := 4$

$k := 3$

$S_3 \,\square\, S_2 b$

$aba \,\square\, abb$ ✗

$k := 2$

$S_2 \,\square\, S_2 b$

$ab \,\square\, abb$ ✗

$k := 1$

$S_1 \,\square\, S_2 b \quad \delta(2, b) := 0$

$a \,\square\, abb$ ✗

$x := c \quad \delta(2, c) := 0$

4. $q := 3$

$x := a \quad k := \min(8, 5) = 5$

$k := 4$

$S_4 \,\square\, S_3 a$

$abab \,\square\, abaa$ ✗

$k := 3$

$S_3 \,\square\, S_3 a$

$aba \,\square\, abaa$ ✗

$k := 2$

$ab \,\square\, abaa$ ✗

$k := 1$

$a \,\square\, abaa$ ✓

$\delta(3, a) := 1$

⋮