02. KDA

~ Краен детерминиран автомат ~

Интуитивно: КДА е програма, използваща константно количество памет

Пример:

Задача

Вход: дума α

Изход:
да, ако $|\alpha|$ е четно
не, ако $|\alpha|$ е нечетно

```
string str;
cin >> str;

cout<<str.length%2==0;
```

Не използва константна памет

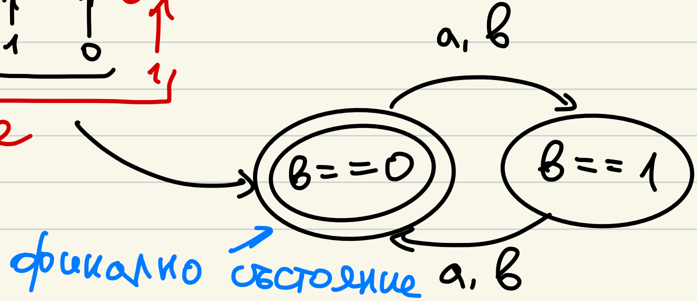
```
char c; bool b=0;      //counter=0
while (cin >> c) {
    b != b;            //counter++
}
cout << b==0;          //counter%2==0
```

вход: $\alpha = a\ b\ a\ a\ b$

b = 0 1 0 1 0 1

да

не



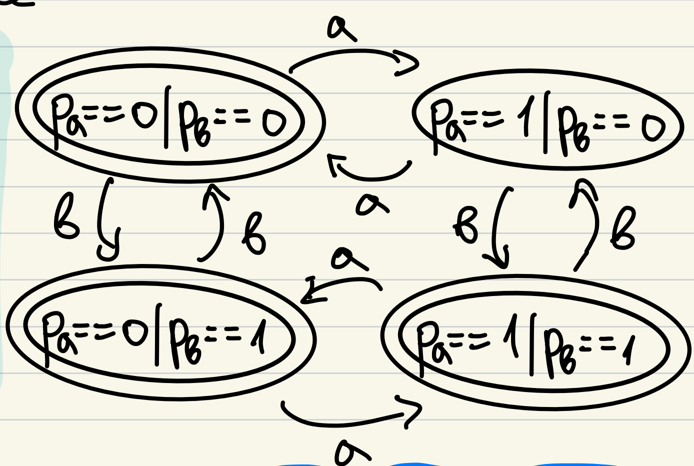
Задача 2:

вход: дума α

изход: да, ако бр. а е четно или бр. на в е нечетно

не, иначе

```
char c; bool pa=0, pb=0;
while (cin>>c){
  if (c=='a') pa=!pa;
  else pb!=pb
}
cout <..<
```



Def/ Краен автомат кортеже

$A: \langle \Sigma, Q, s, F, \delta \rangle$, където:

* Σ - крайна азбука
* Q - крайно мн-во от състояния
* $s \in Q$ - начално състояние
* $F \subseteq Q$ - мн-во от финални състояния (даващи отг. да)
* δ - функция на преходите $\delta: Q \times \Sigma \to Q$

Пример: (задача 2)

$\Sigma = \{a, b\}$

$Q = \{(0|0), (0|1), (1|0), (0|1)\}$

$S = \{(0|0)\}$

$F = \{(0|0), (0|1), (1|1)\}$

$\delta((0|0), a) = (1|0)$

$\delta((0|0), b) = (0|1)$

$\delta((0|1), a) = (1|1)$

$\delta((0|1), b) = (0|0)$

$\vdots$

Деф Език на автомат A

$L(A) = \{\alpha \in \Sigma^* \mid$ пътят, опр. от думата α и започващ от нач. състояние, приключва във финално състояние$\}$

$aaababbb \in L$, $aaababbba \notin L$

отсъствах доста

Припомнеме какво взехме до тук:

Език $L \subseteq \Sigma^*$ ← множеството от всички думи над азбуката Σ

Операции над езици: $\cap$; $\cup$; $\setminus$; $*$; $\cdot$
звезда, конкатенация

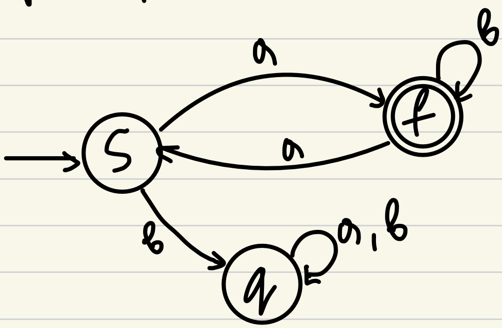
Краен автомат $A \langle \Sigma, Q, s, F, \delta \rangle$

Σ - азбука, Q - кр. мн-во от състояния

s - нач. с-ние, $F \subseteq Q$ - финални състояния, $\delta: Q \times \Sigma \to Q$ - ф-я на преходите

Тотален автомат – δ-тотална

Примери:

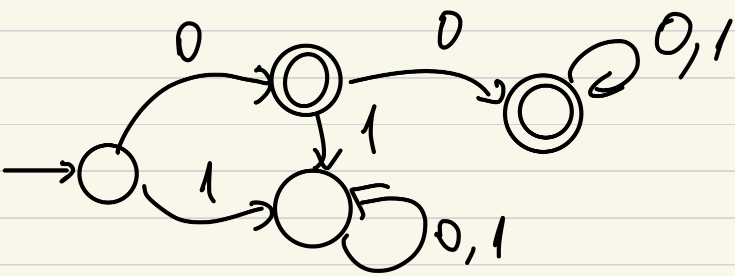


✓ abb
✗ bbb
✗ babb
✓ aaab

$\delta(s,a) = f$

$\delta(f,b) = f$

$\delta(q,a) = q$



дефинираме: $\delta^*: Q \times \Sigma^* \to Q$

индуктивна дефиниция по дълж. на α

за $\delta^*(p, \alpha)$, за всяко състояние $p \in Q$

* База: $|\alpha| = 0$, $\alpha = \varepsilon$, тогава деф

$\delta^*(p, \varepsilon) \overset{def}{=} p \quad \forall p \in Q$

* И.Х.: нека сме деф $\delta^*(p, \alpha)$ $\forall \alpha$, $|\alpha| = n$
* И.С.: ще деф $\delta^*(p, \alpha)$ за $|\alpha| = n+1$

Нека $\alpha = \beta . c$ $\quad |\beta| = n$, $c \in \Sigma$, тогава

$\delta^*(p, \beta.c) = \delta^*(\delta^*(p, \beta), c)$

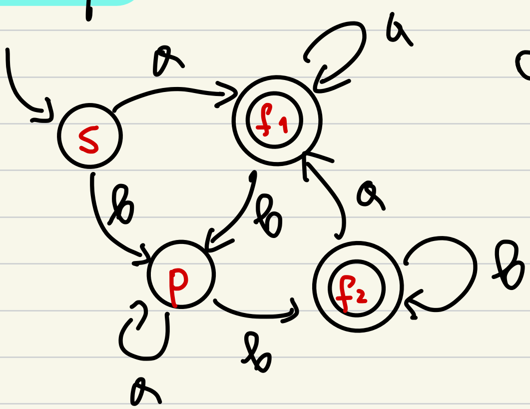
$\delta^*(p, \alpha) = q \iff$ пътечката, опр. от α и започваща в състояние p, завършваща в q

---

$L(\mathcal{A}) = \{\alpha \in \Sigma^* \mid \delta^*(s, \alpha) \in F\}$

Пример:

$\mathcal{A}$



финални означ. с ◎ или $\bigcirc^*$

$\delta(p, b) = f_2$

$\delta^*(p, aba) = f_1$

$\delta(f_1, a) = f_1$

$\delta^*(f_2, baab) = p$

? $abab \in L(\mathcal{A})$

$\delta^*(s, abab) = f_2 \in F$

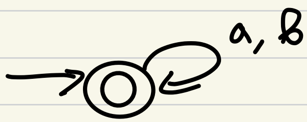
? $aaba \in L(\mathcal{A})$

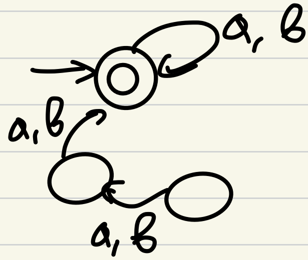
$\delta^*(s, aaba) = p \notin F \Rightarrow aaba \notin L(\mathcal{A})$

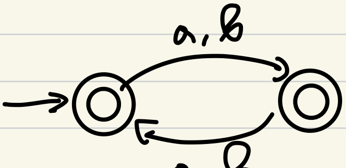
Казваме, че

* езикът L е автоматен, ако има КДА $\mathcal{A}$, такъв, че $L(\mathcal{A}) = L$
* автоматът $\mathcal{A}$ разпознава дума α, ако $\alpha \in L(\mathcal{A})$
* автоматът $\mathcal{A}$ отхвърля дума α, ако $\alpha \notin L(\mathcal{A})$

Зад. Да се построи автомат $\mathcal{A}$ с език $L(\mathcal{A}) = \Sigma^*$







! Съществува изоморфизъм м/у автомати

Критерий: $L(\mathcal{A}) = \Sigma^* \Leftrightarrow$ всяко състояние, достижимо от началното е финално

Д-во: ($\Rightarrow$) $L(\mathcal{A}) = \Sigma^*$, и нека p е достижимо от s

Тогава има дума $\alpha \in \Sigma^*$: $\delta^*(s,\alpha) = p$,
но $\alpha \in \Sigma^* = L(\mathcal{A})$

тогава $\delta^*(s,\alpha) \in F = p$

($\Leftarrow$) Нека всяко състояние, дост. от s, е финално

Нека $\alpha \in \Sigma^*$, тогава състоянието $\delta^*(s,\alpha)$ е достижимо от s

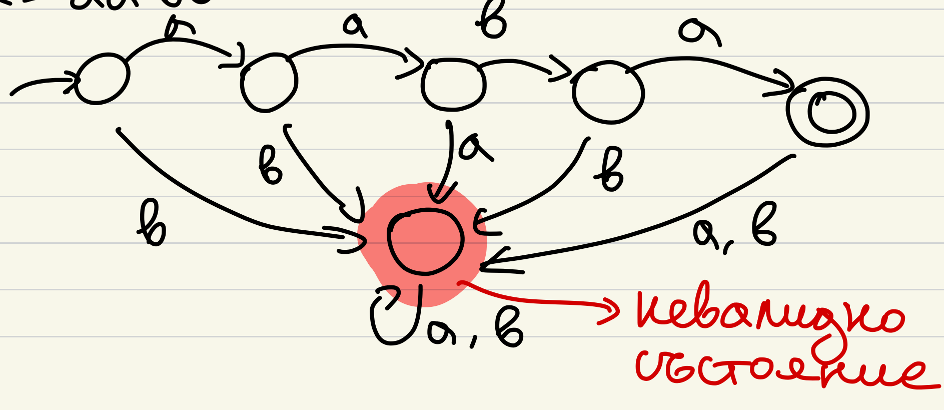
Тогава $\delta^*(s,\alpha) \in F$, т.е. $\alpha \in L(\mathcal{A})$

$$\left.\begin{array}{l} \Rightarrow \Sigma^* \subseteq L(\mathcal{A}) \\ \text{и } L(\mathcal{A}) \subseteq \Sigma^* \end{array}\right\} \Rightarrow \Sigma^* = L(\mathcal{A})$$

$L(\mathcal{A}) = \emptyset \Leftrightarrow$ Всяко състояние p, достижимо от началното, е нефинално

Зад. Нека $\alpha \in \Sigma^*$, да се построи автомат $\mathcal{A}$ с $L(\mathcal{A}) = \{\alpha\}$

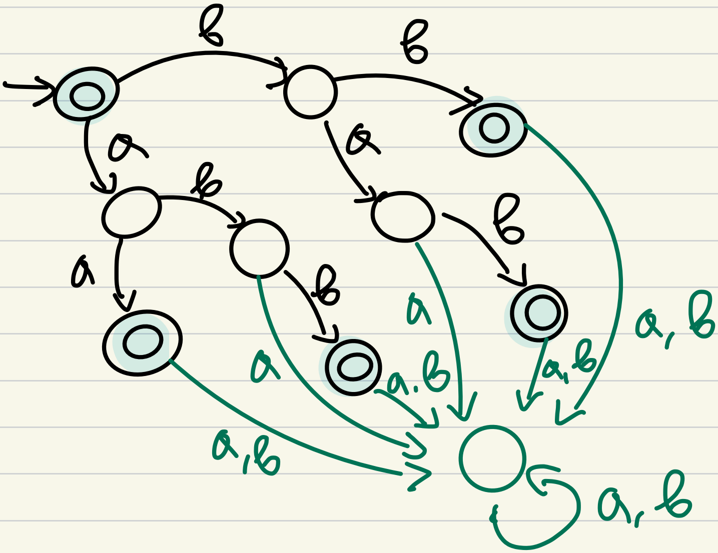
Нека $\alpha = aaba$



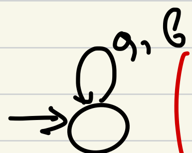
⊛

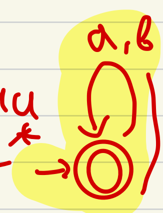
Зад. Нека L е „краен", тогава има автомат $\mathcal{A}$, такъв, че $L(\mathcal{A}) = L$

Нека $L = \{aa, bab, \varepsilon, abb, bb\}$



Идея:

1 Ако $L = \emptyset$ $\rightarrow$ 

(два фундаментални автомата, $L = \Sigma^*$ 

)

2 Ако $L \neq \emptyset$ и е краен

Нека $k = \max\{|\alpha| \mid \alpha \in L\}$

Разгл. автомат със състояния

$$Q = \{\alpha \in \Sigma^* \mid |\alpha| \leq k\} \cup \{err\}$$

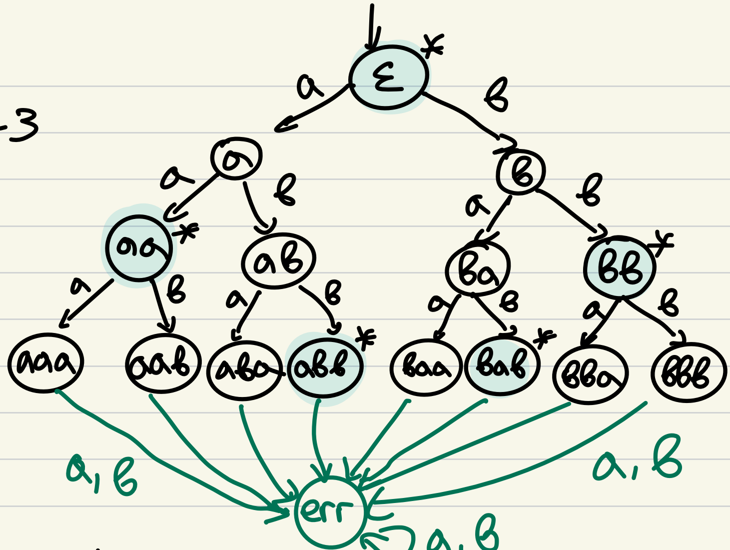
Деф. $s = \varepsilon$, $F = L$

$$\text{за } \forall a \in \Sigma \begin{cases} \delta(p, a) = \begin{cases} err, & \text{ако } |p| = k \\ pa, & \text{ако } |p| < k \end{cases} \\ \delta(err, a) = err \end{cases}$$

За (*):

$\max\{|\alpha| \mid \alpha \in L\} = 3$

съставяме преходно дърво:



За упр: постр. A за следните L

* $L_1 = \{\alpha \mid |\alpha| \text{ е четно}\}$

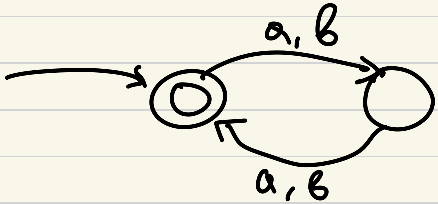
* $L_2 = \{\alpha \mid |\alpha| \equiv 1 \pmod{3}\}$

* $L_3 = \{\alpha \mid \alpha \text{ започва с } ab\}$

* $L_4 = \{\alpha \mid \alpha \text{ завършва с } ba\}$

A ?

Зад. 1 $L = \{\alpha \in \Sigma^* \mid |\alpha| \equiv 0 \pmod{2}\}$

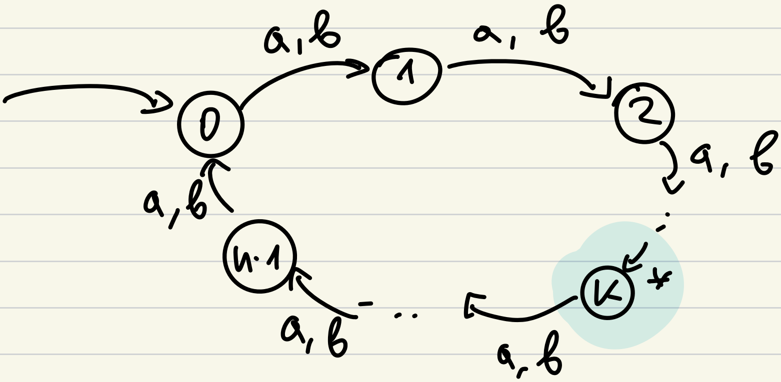


$\varepsilon \in L(A) \iff s \in F$

Зад. 2 Нека $n \geq 2$, $n \in \mathbb{N}$ и $k \in \mathbb{N}$

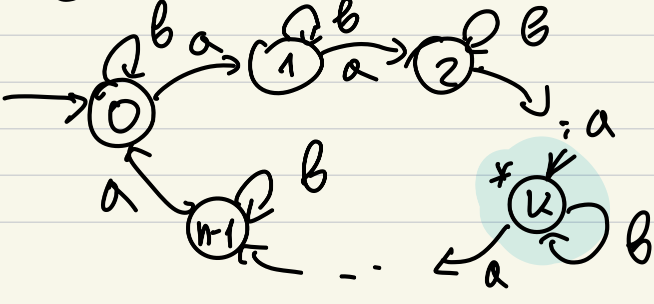
$L = \{\alpha \in \Sigma^* \mid |\alpha| \equiv k \pmod{n}\}$ е автоматен

имаме n състояния

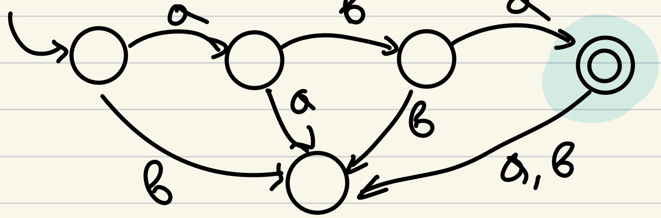


Деф/ Броят на срещанията на буквата с в думата α означаваме с $|\alpha|_c$

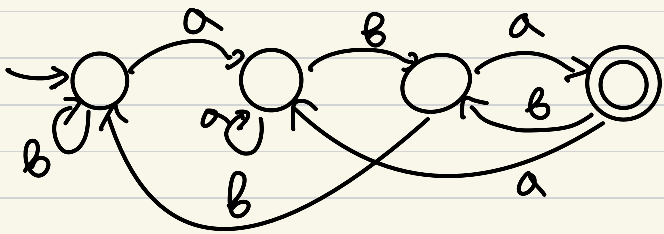
Зад. $L = \{\alpha \in \Sigma^* \mid |\alpha|_a \equiv k \pmod{n}\}$ е авт.



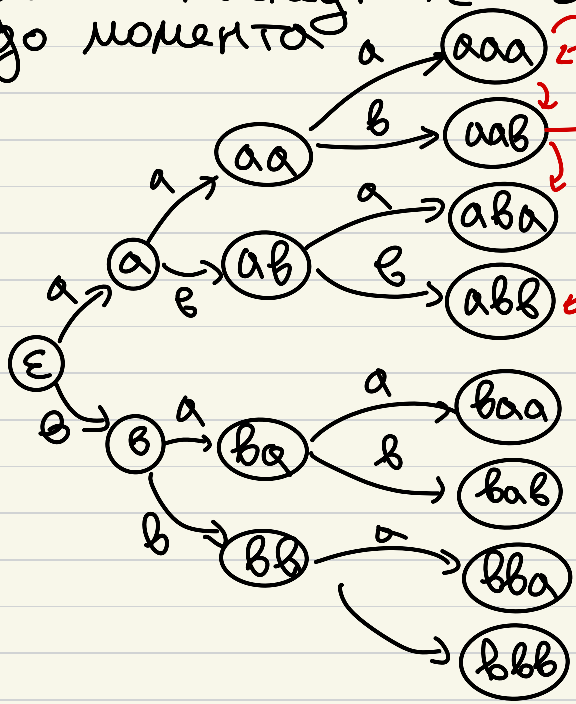
Зад. 4 $L = \{aba\}$



Зад. 5 $L = \{\alpha \in \Sigma^* \mid$ завършва на $aba\}$



Идея: В състоянията на автомата кодираме последните 3 символа прочетени до момента



мислим, че ако се намираме в част aaa, то прочетената част до момента завършва на aaa

на състояние xyz добавяме преход буква 'c' към yzc

Описание на автомата за

$L = \{\alpha \in \Sigma^* \mid \alpha$ завършва на $aba\}$

$Q = \{\beta \in \Sigma^* \mid |\beta| \leq 3\}$ $F = \{aba\}$ $s = \varepsilon$

$\forall c \in \Sigma^*$ $\delta(\beta, c) = \beta c$, ако $|\beta| < 3$

$\delta(xyz, c) = yzc$, където $x, y, z, c \in \Sigma$

Тв* Ще док, че

$$\delta^*(s, \alpha) = \begin{cases} \alpha, & \text{ако } |\alpha| < 3 \\ xyz, & \text{ако } |\alpha| \geq 3 \text{ и } \alpha \text{ зав на } xyz \end{cases}$$

Искаме да док, че $L(\mathcal{A}) = L$

$\Rightarrow$ $L(\mathcal{A}) \subseteq L$

Нека $\alpha \in L(\mathcal{A})$, тогава $\delta^*(s, \alpha) \in F$

Тогава $\delta^*(s, \alpha) = aba$

От Тв*, $|\alpha| \geq 3$ и α зав. на aba

$\Rightarrow \alpha \in L$

$\Leftarrow$ $L \subseteq L(\mathcal{A})$

Нека $\alpha \in L$, тогава α зав на aba, от Тв*

$\delta^*(s, \alpha) = aba$, $aba \in F \Rightarrow \delta^*(s, \alpha) \in F$

$\Rightarrow \alpha \in L(\mathcal{A})$

Д-во на Тв*, индукция по $|\alpha|$

База: $|\alpha| = 0$

$\alpha = \varepsilon$, тогава $\delta^*(s, \varepsilon) = s = \varepsilon$

$\Rightarrow |\alpha| < 3$ и $\alpha = \varepsilon$

усл. за α е изпълнено

И.Х: Нека знаем за всяко α с $|\alpha| \leq n$, че усл. е изпълнено

И.С: $|\alpha| = n+1$

Нека $\alpha = \beta c$ $\quad |\beta| = n$. Тогава по И.Х. Тв* важи за β

1 сл.) $|\beta| < 3$, тогава $|\alpha| \leq 3$. От Тв* $\delta^*(s, \beta) = \beta$

$\delta^*(s, \alpha) = \delta(\delta^*(s, \beta), c) = \delta(\beta, c) = \beta c$

Щом $|\alpha| \leq 3$, то Тв* изисква $\delta^*(s, \alpha) = \alpha = \beta c$

2 сл.) $|\beta| \geq 3$, тогава $|\alpha| > 3$.

От Тв* $\delta^*(s, \beta) = xyz$, където β зав на xyz

$\delta^*(s, \alpha) = \delta(\delta^*(s, \beta), c) = \delta(xyz, c) = yzc$

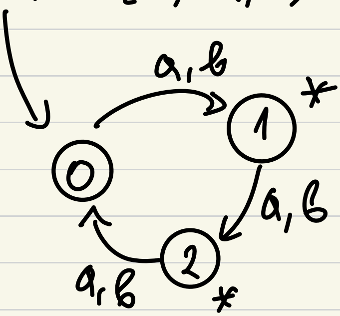
$\alpha = \beta c$ и β зав на xyz, тогава посл. 3 букви на α са yzc

$\Rightarrow$ доказано

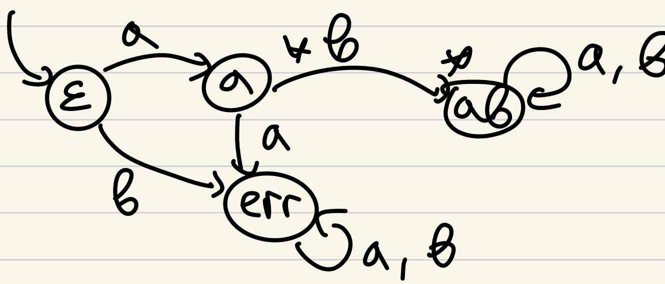
Тв| Нека L_1 и L_2 са автоматни, тогава
$L_1 \cap L_2$, $L_1 \cup L_2$, $L_1 \setminus L_2$ са автоматни

$*$ $L_1 \cap L_2$ $\quad \alpha \in L_1 \cap L_2 \Leftrightarrow \alpha \in L_1 \;\&\; \alpha \in L_2$

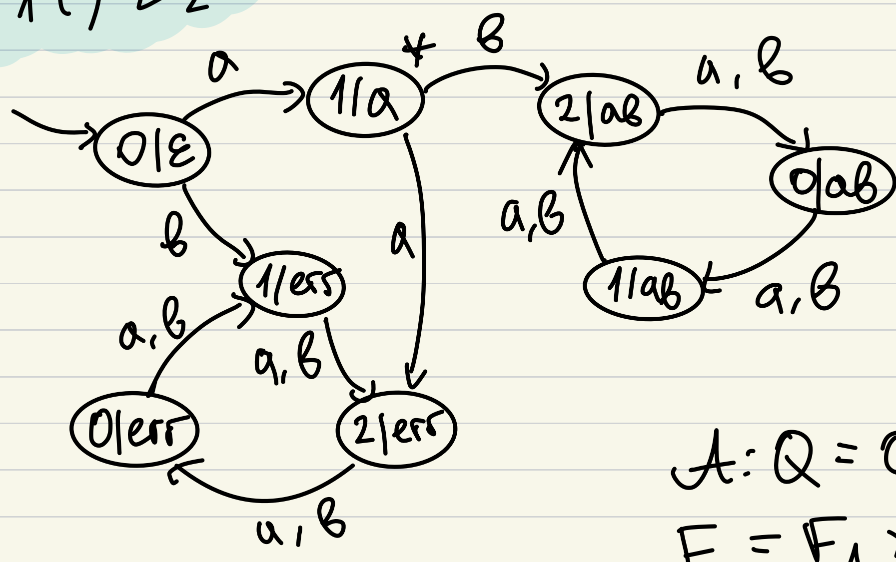
$A_1 = \{\Sigma, Q_1, s_1, \delta_1, F_1\}$



$A_2 = \{\Sigma, Q_2, s_2, \delta_2, F_2\}$



$L_1 \cap L_2$:



$A: Q = Q_1 \times Q_2$

$F = F_1 \times F_2$

$s = \langle s_1, s_2 \rangle$

Вход: abaab

A_1: $0 \xrightarrow{a} 1 \xrightarrow{b} 2 \xrightarrow{a} 0 \xrightarrow{a} 1 \xrightarrow{b} 2$

A_2: $\varepsilon \xrightarrow{a} a \xrightarrow{b} ab \xrightarrow{a} ab \xrightarrow{a} ab \xrightarrow{b} ab$

Тв: $\delta^*(\langle s_1, s_2 \rangle, \alpha) = \langle \delta^*(s_1, \alpha), \delta^*(s_2, \alpha) \rangle$

Д-во: индукция по $|\alpha|$ ← за упр.

$L_1 \cap L_2 = L(A)$

$\alpha \in L_1 \cap L_2 \iff \alpha \in L_1 \,\&\, \alpha \in L_2$

$\iff \alpha \in L(A_1) \,\&\, \alpha \in L(A_2)$

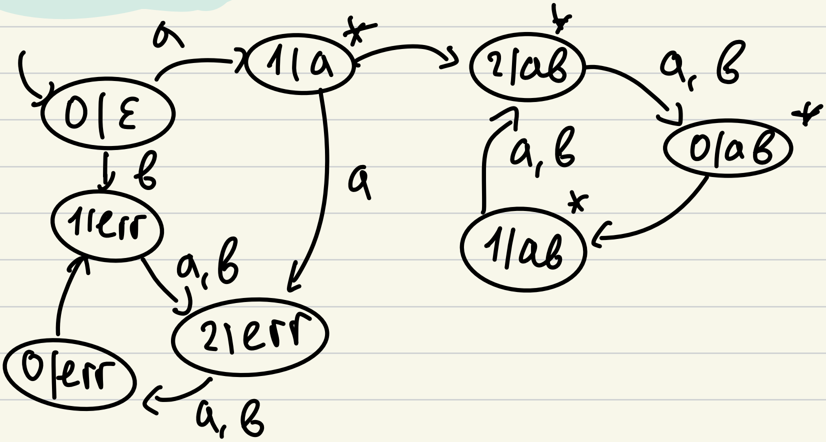
$\iff \delta_1^*(s_1, \alpha) \in F_1 \,\&\, \delta_2^*(s_2, \alpha) \in F_2$

$\iff \langle \delta_1^*(s_1, \alpha), \delta_2^*(s_2, \alpha) \rangle \in F_1 \times F_2$

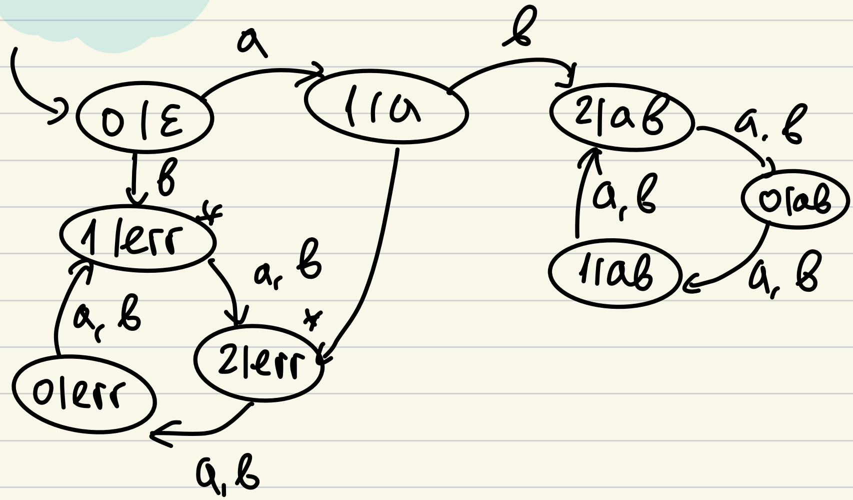
$\iff \delta^*(s, \alpha) \in F$

$\iff \alpha \in L(A)$

$L_1 \cup L_2$



$L_1 \setminus L_2$



$$A_1 = \langle \Sigma, Q_1, s_1, F_1, \delta_1 \rangle$$

$$A_2 = \langle \Sigma, Q_2, s_2, F_2, \delta_2 \rangle$$

A: $Q = Q_1 \times Q_2$ $\quad \delta(\langle p_1, p\rangle, c) = \langle \delta_1(p_1, c), \delta_2(p_2, c)\rangle$

$s = \langle s_1, s_2 \rangle$ $\quad \delta^*(\langle p_1, p_2\rangle, \alpha) = \langle \delta_1^*(p_1, \alpha), \delta_2^*(p_1, \alpha)\rangle$

$$F_\cap = F_1 \times F_2$$

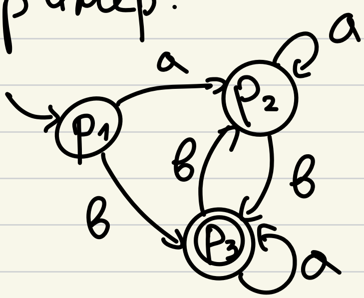
$$F_\cup = \{\langle p_1, p_2\rangle \mid p_1 \in F_1 \text{ или } p_2 \in F_2\} = Q_1 \times F_2 \cup F_1 \times Q_2$$

$$F_\setminus = \{\langle p_1, p_2\rangle \mid p_1 \in F_1 \;\&\; p_2 \notin F_2\} = \cancel{F_1 \setminus (Q_1 \times Q_2)} \; F_1 \times (Q_2 \setminus F_2)$$

Ако L е автоматен и $\overline{L} = \Sigma^* \setminus L$ е автоматен

Ако $\mathcal{A} = \langle \Sigma, Q, s, F, \delta \rangle$, то $\overline{\mathcal{A}} = \langle \Sigma, Q, s, Q \setminus F, \delta \rangle$ разпознава $\overline{L(\mathcal{A})}$

Пример:



$$\alpha \in L(\mathcal{A}) \Leftrightarrow \delta^*(s,\alpha) = p_3$$

$$\alpha \in \overline{L(\mathcal{A})} \Leftrightarrow \alpha \notin L(\mathcal{A}) \Leftrightarrow \underbrace{\delta^*(s,\alpha) \neq p_3}_{\notin F} \Leftrightarrow$$

$$\Leftrightarrow \underbrace{\delta^*(s,\alpha) = p_2 \text{ или } \delta^*(s,\alpha) = p_1}_{\in Q \setminus F}$$

$$\Leftrightarrow \alpha \in L(\overline{\mathcal{A}})$$

Зад. Да се док, че
$L = \{\alpha \in \Sigma^* \mid |\alpha|$ е четно $\Leftrightarrow \alpha$ завършва на $bab\}$
е автоматен

$\varphi_1 \Rightarrow \varphi_2$ е вярно точно тогава, когато φ_1 не е вярно или φ_2 е вярно $(\neg\varphi_1 \vee \varphi_2)$

$\varphi_1 \Leftrightarrow \varphi_2$ е вярно точно тогава, когато φ_1 и φ_2 когато φ_1 и φ са верни или φ_1 и φ_2 са грешни
$((\neg\varphi_1 \wedge \neg\varphi_2) \vee (\varphi_1 \wedge \varphi_2))$

$L = \{\alpha \in \Sigma^* \mid (|\alpha|$ е четна и α завършва на $bab)$ или $(|\alpha|$ не е четна и α не завършва на $bab)\} =$

$= \{\alpha \in \Sigma^* \mid (|\alpha|$ е четна$)$ и $(\alpha$ зав. на $bab)\}$
$\cup \{\alpha \in \Sigma^* \mid (|\alpha|$ не е четна$)$ и $(\alpha$ не зав. на $bab)\} =$

$= [\{\alpha \in \Sigma^* \mid |\alpha|$ е четна$\} \cap \{\alpha \in \Sigma^* \mid \alpha$ зав. на $bab\}]$

$\cup [\overline{\{\alpha \in \Sigma^* \mid |\alpha| \text{ е четна}\}} \cap \overline{\{\alpha \in \Sigma^* \mid \alpha \text{ зав. на } bab\}}]$

$= (L_1 \cap L_2) \cup (\overline{L_1} \cap \overline{L_2})$

сега показваме, че L_1 и L_2 са автоматни

$\downarrow$

Операциите $\cup, \cap, -$ запазват автоматичните езици и езикът L се получава чрез прилагане на краен брой такива операции в/у автоматни езици.
$\Rightarrow L$ е автоматен

Зад. Да се док, че винаги когато L_1 и L_2 са автоматни, то езикът

$L = \{\alpha_1\beta_1\alpha_2\beta_2 \dots \alpha_n\beta_n \mid \alpha \in L_1, \beta \in L_2$ и $\alpha_i, \beta_i \in \Sigma, \alpha = \alpha_1\alpha_2 \dots \alpha_n, \beta = \beta_1\beta_2 \dots \beta_n\}$

също е автоматен

α_i - i-тата буква на α

Пример: $L_1 = \{abb\}, L_2 = \{baa\} \Rightarrow L = \{abbaba\}$

идея: $L_1 = \{\alpha \in \Sigma^* \mid \alpha$ не съдържа $b\}$

$L_2 = \{\alpha \in \Sigma^* \mid \alpha$ не съдържа $a\}$

$L_1 = \{a^n \mid n \in \mathbb{N}\} \quad L_2 = \{b^n \mid n \in \mathbb{N}\}$

$\alpha \in L_1 \quad \beta \in L_2 \Rightarrow \alpha = \overbrace{aa \dots a}^{n} \Rightarrow p = \underbrace{(ab)(ab) \dots (ab)}_{n} \in L$

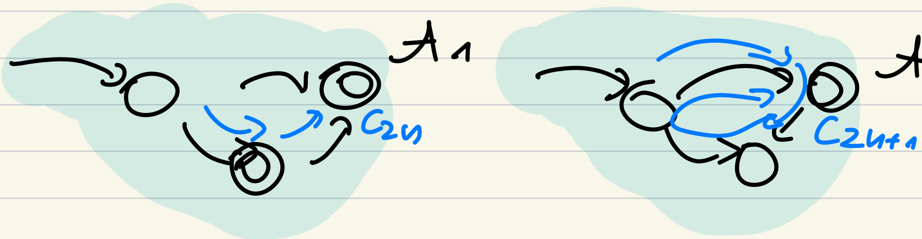
$\beta = \underbrace{bb \dots b}_{n}$

Реш: L_1, L_2 - автоматни

Нека A_1 и A_2 са автомати съотв за L_1 и L_2

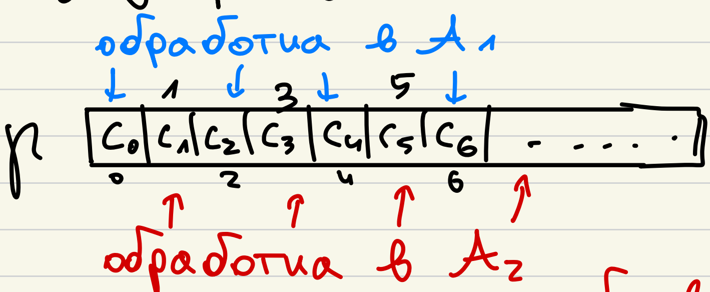
$A_1 = \langle \Sigma, Q_1, s_1, F_1, \delta_1 \rangle$

$A_2 = \langle \Sigma, Q_2, s_2, F_2, \delta_2 \rangle$



вход: дума p

изход: пров. дали $p \in L$



булев флаг

$A: Q = Q_1 \times Q_2 \times \{0, 1\}$ т.е $\langle p, q, f_1 \rangle$ ($p \in Q_1$, $q \in Q_2$, $f_1 \in \{0,1\}$)

на всяка прочетена буква следим:

* до къде сме стигнали в A_1 по p
* до къде сме стигнали в A_2 по p
* f_1 ни казва в кой автомат да обработим прочетената буква

$F = F_1 \times F_2 \times \{0\}$

$s = \langle s_1, s_2, 0 \rangle$

$\delta(\langle p_1, p_2, 0 \rangle, c) = \langle \delta_1(p_1, c), p_2, 1 \rangle$

$\delta(\langle p_1, p_2, 1 \rangle, c) = \langle p_1, \delta_2(p_2, c), 0 \rangle$

<u>Тв</u>| Нека $p \in \Sigma^*$, $p = p_1 p_2 \dots p_n$.
Нека even(p) е думата, съставена от буквите в p на четни позиции, а odd(p) е думата, съставена от буквите на нечетни позиции

Пример: $\text{even}(\underset{-}{a}^{0}b^{1}\underset{-}{b}^{2}a^{3}\underset{-}{b}^{4}) = abb$ $\quad \text{odd}(a^{0}\underset{-}{b}^{1}b^{2}\underset{-}{a}^{3}b^{4}) = ba$

(*) Тогава $\delta^*(s, p) = \langle \delta_1^*(s_1, \text{even}(p)), \delta_2^*(s_2, \text{odd}(p)), |p| \% 2 \rangle$

Д-во: Индукция по $|p|$

База: $|p| = 0$, т.е. $p = \varepsilon$

$\delta^*(s, p) = \delta^*(s, \varepsilon) = s = \langle s_1, s_2, 0 \rangle = \langle \delta_1^*(s_1, \varepsilon), \delta_2^*(s_2, \varepsilon), \overset{|\varepsilon| \% 2}{0} \rangle$

И.Х.: Нека знаем, че ако $|p| \leq n$, то (*) е изп.

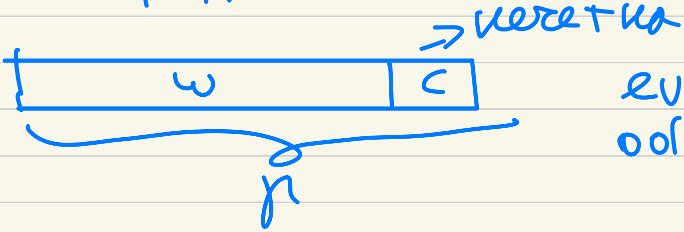
И.С.: Нека $|p| = n+1$. Тогава $p = w.c$, където $c \in \Sigma$ и $|w| = n$

От И.Х. (*) е в сила за w, т.е.

$\delta^*(s, w) = \langle \delta_1^*(s_1, \text{even}(w)), \delta_2^*(s_2, \text{odd}(w)), |w| \% 2 \rangle$

$\delta^*(s, p) = \delta(\delta^*(s, w), c) = \delta(\langle \delta_1^*(s_1, \text{even}(w)), \delta_2^*(s_2, \text{odd}(w)), |w| \% 2 \rangle, c)$

Ако $|w| \% 2 = 0$



$even(p) = even(w)\cdot c$

$odd(p) = odd(w)$

1сл. $|w|$ е четна, тогава: $\delta^*(s, p) =$

$\delta(\langle \delta_1^*(s_1, even(w)), \delta_2^*(s_2, odd(p)), 0\rangle, c) =$

$= \langle \delta_1(\delta_1^*(s_1, even(w)), c), \delta_2^*(s_2, odd(p)), 1\rangle =$

$= \langle \delta_1^*(s_1, even(w)\cdot c), \delta_2^*(s_2, odd(p)), 1\rangle =$

$= \langle \delta_1^*(s_1, even(w).c), \delta_2^*(s_2, odd(p)), |p|\%2\rangle$

2сл. $\Uparrow$ упр.

Сега твърдим, че $L = L(\mathcal{A})$

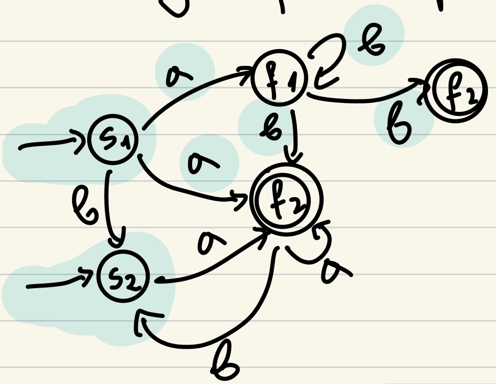
$p \in L \Leftrightarrow |p|$ е четна и $even(p) \in L_1$, $odd(p) \in L_2$

$\Leftrightarrow |p|$ е четна и $\delta_1^*(even(p)) \in F_1$ и $\delta_2^*(odd(p)) \in F_2$

$\Leftrightarrow \langle \delta_1^*(s_1, even(p)), \delta_2^*(s_2, odd(p)), |p|\%2\rangle \in F_1 \times F_2 \times \{0\}$

$\Leftrightarrow \delta^*(s, p) \in F_1 \times F_2 \times \{0\} = F \Leftrightarrow p \in L(\mathcal{A})$

~Недетерминирани крайни автомати~



* една буква може да води до няколко състояния

* повече от 1 начални състояния

Пример:

$ab: s_1 \xrightarrow{a} f_1 \xrightarrow{b} f_1 \quad s_1 \xrightarrow{a} f_1 \xrightarrow{b} f_3$

$s_1 \xrightarrow{a} f_1 \xrightarrow{b} f_2 \quad s_1 \xrightarrow{a} f_2 \xrightarrow{b} s_2$

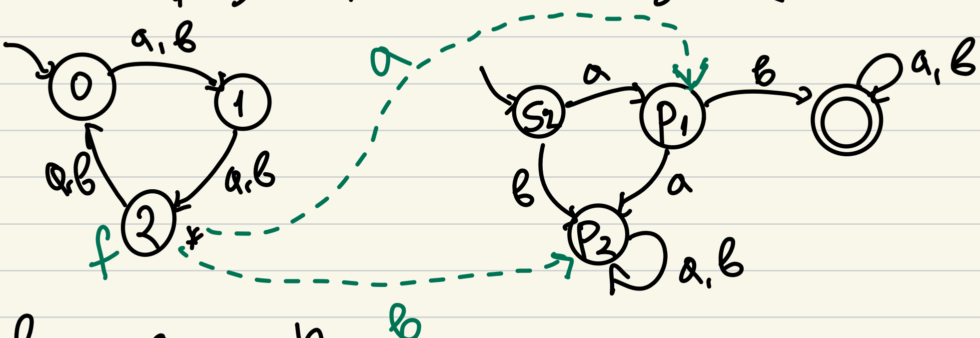
$s_1 \xrightarrow{a} f_3 \xrightarrow{b} s_2$

$L_1 = \{\alpha \in \Sigma^* \mid |\alpha| \equiv 2 \pmod 3\}$

$L_2 = \{\alpha \in \Sigma^* \mid \alpha$ започва с $ab\}$

$L_1.L_2 = \{\alpha.\beta \mid \alpha \in L_1, \beta \in L_2\}$

$\mathcal{A}_1$ за L_1 $\quad$ $\mathcal{A}_2$ за L_2



вход: дума γ

изход: дали $\gamma \in L_1.L_2$

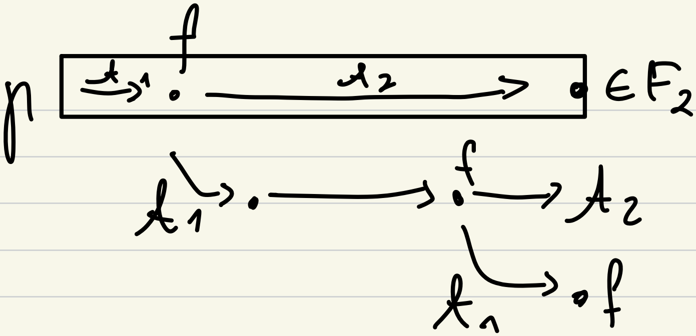
$\boxed{\in L_1 \mid \in L_2}\ \gamma$

f

$\gamma = aaaaa|aba$

$\in L_1 \quad \in L_2$

$\left.\begin{array}{l}\alpha = a^5 \\ \beta = aba\end{array}\right\} \begin{array}{l}\alpha \in L_1 \\ \beta \in L_2\end{array} \quad \alpha.\beta = \gamma \Rightarrow \gamma \in L_1.L_2$



по някой клон да стигнем до F_2 на A_2

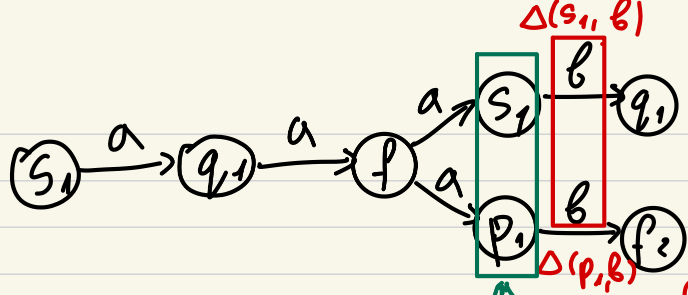
Деф Краен недетерминиран автомат е $N\langle \Sigma, Q, I, F, \Delta \rangle$, където

* Σ, Q, F - същите като при КДА
* $I \subseteq Q$ - множество от начални състояния
* $\Delta : Q \times \Sigma \to P(Q)$ - функция на преход

$$\Delta^* : P(Q) \times \Sigma^* \to P(Q)$$

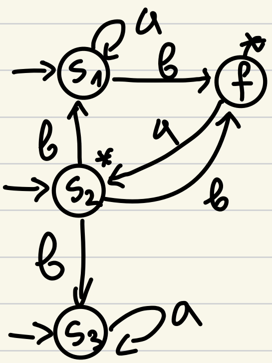
$$\Delta^*(R, \varepsilon) = R \quad \forall R \subseteq Q$$

$$\Delta^*(R, \rho a) = \bigcup_{p \in \Delta^*(R, \rho)} \Delta(p, a)$$



$\Delta^*(\{s_1\}, aa)$ $\Delta^*(\{s_1\}, aaa) = \{s_1, p_1\}$ $\Delta^*(\{s_1\}, aaab)$

$L(\mathcal{N}) = \{\alpha \in \Sigma^* \mid \Delta^*(I, \alpha) \cap F \neq \emptyset\}$



$\Delta^*(I, \alpha)$:

1) $\Delta^*(I, \varepsilon) = \{s_1, s_2, s_3\} = I \in L(\mathcal{N})$
2) $\Delta^*(I, b) = \{f, s_1, s_3\} \in L(\mathcal{N})$
3) $\Delta^*(I, ba) = \{s_1, s_2, s_3\} \in L(\mathcal{N})$
4) $\Delta^*(I, baa) = \{s_1, s_3\} \notin L(\mathcal{N})$

Тв| Нека L-език. Тогава има КДА $\mathcal{A}$ такъв, че $L = L(\mathcal{A})$ $\Leftrightarrow$ има КНА $\mathcal{N}$ т.че $L = L(\mathcal{N})$

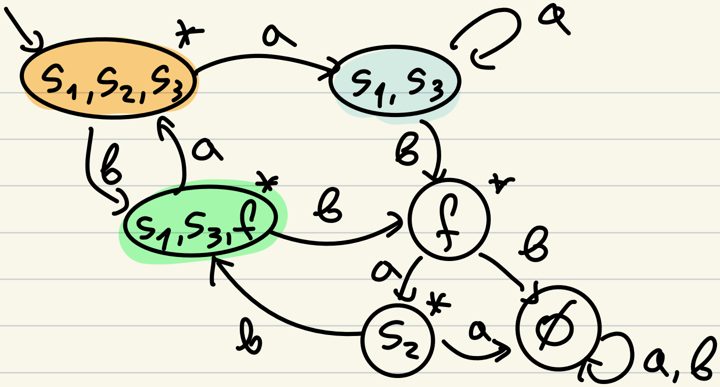
Д-во: (идея)

$\Rightarrow$ Ще построим детерминиран автомат $\mathcal{A}$, в който състоянията са $P(Q)$, за които ще мислим като за текущи нива в дървото на изображение на N

$\mathcal{A}$:

$Q_A = P(Q_N)$ $\delta_A(R, a) = \bigcup_{p \in R} \Delta(p, a)$ $F_A = \{R \in Q_A \mid R \cap F_N \neq \emptyset\}$

$s_A = I$



Q_N	a	b
s_1	s_1	f
s_2	x	s_1, s_3, f
s_3	s_3	x
f	s_2	x
⋮	⋮	⋮

Алгоритъм за построяване на детерминиран автомат A по вход недетерминиран автомат N, такъв, че $L(A) = L(N)$!

1. Поставяме началното състояние да е I_N

2. Строим таблица от вида $\boxed{Q \mid a \mid b}$ ..

$p_1 \mid \Delta(p_1, a) \mid \Delta(p_1, b)$ ..

3. Докато има състояние $\{q_1, .., q_k\}$ и буква σ, върху което няма дефиниран преход, гледаме в таблицата колоната на σ и редовете на $q_1, .., q_k$, събираме всички състояния накуп – R и добавяме преход с σ към R

4. Поставяме като финални състояния тези, които съдържат поне едно финално от F_N