$R \vee S$

$(R \wedge \neg S) \vee (\neg R \wedge S)$

$(R \vee S) \wedge (\neg R \vee \neg S)$

$A + 0 = A$

$A \cdot 1 = A$

$$\boxed{\begin{array}{ll} \wedge \equiv \cdot \\ \vee \equiv + \end{array}}$$

$0 \vee A = A$

$A : B^n \longrightarrow \{0, 1\}$

$$A + \overline{A} = 1$$

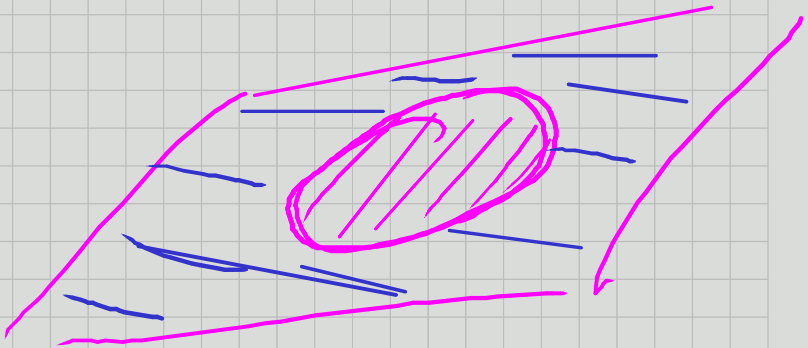
$$A \cdot \overline{A} = 0$$

$$A \vee \neg A = 1$$

$$A \wedge \neg A = 0$$

$$\neg(\neg A) = A$$

$$\overline{A} = \neg A$$



$$\overline{A \cdot B} \equiv \neg(A \wedge B)$$

$$\overline{A} + \overline{B} \equiv \neg A \vee \neg B$$

$$\overline{A + B} \equiv \neg(A \vee B)$$

$$\overline{A} \cdot \overline{B} \equiv (\neg A) \wedge (\neg B)$$

$\neg(A \wedge B)$ $\quad\quad$ $\neg A \vee \neg B$

0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	

1 $(R \land \neg S) \lor (\neg R \land S)$

2 $(R \lor S) \land (\neg R \lor \neg S)$

3 $A \land (B \lor C)$

4 $(A \land B) \lor (A \land C)$

5 $(R \lor S) \land \neg R \lor (R \lor S) \land \neg S$

6 $\neg R \land (R \lor S)$

7 $\underbrace{(\neg R \land R)}_{\begin{smallmatrix}\parallel\\ 0\end{smallmatrix}} \lor (\neg R \land S) \lor (\neg S \land R) \lor \underbrace{(\neg S \lor S)}_{\begin{smallmatrix}/\!/\\ 0\end{smallmatrix}}$

$0 \lor \boxed{(\neg R \land S) \lor (\neg S \land R)} \lor 0$

1) [illegible]

2) [illegible]

3) [illegible]

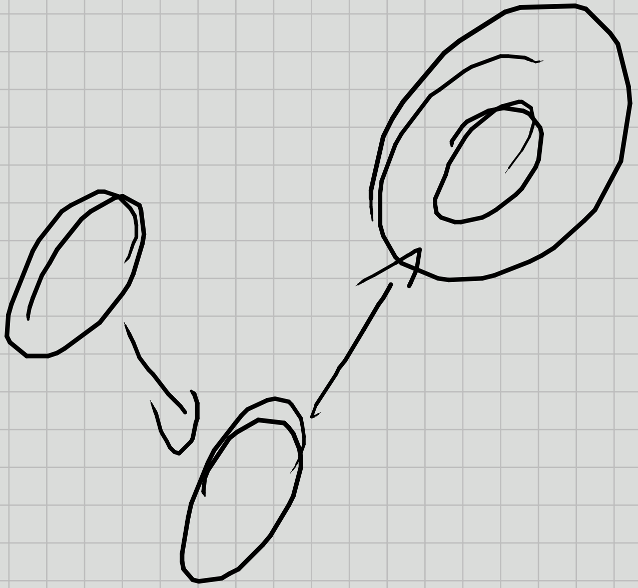
4) [illegible]

$$\textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3} \wedge \textcircled{\neg 4}$$

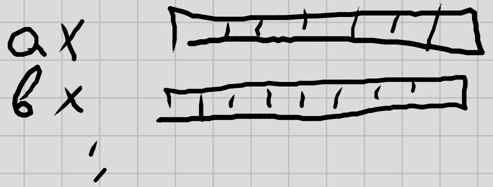
truth table

0	0	0	$\alpha_1 \wedge \beta_1 \wedge \delta_1 \wedge \gamma_1$
0	0	1	$\alpha_2 \wedge \beta_2 \wedge \delta_2 \wedge \gamma_2$
	⋮		
1	1	1	$\alpha_8 \wedge \beta_8 \wedge \delta_8 \wedge \gamma_8$

Automatons and Grammars



CPU



$cx := ax + bx$

$ax \leftarrow cx$

$\sin(ax)$

max $a_1 \quad a_2 \quad a_3 \; . \; . \; . \; a_n$

$5, \; 8, \; 7, \; . \; . \; . \; , \; 1$

$T \sim$ #operations

Q: what is operation?

Q: what is computer

compute = вычислять

число

number → Black Box → number

function: mapping $A \to B$

in some sense, computer implements a function

$\boxed{\Sigma = \{a, b, \dots\}}$ $\Sigma_{01} = \{0,1\} \equiv \{a,b\}$

$\boxed{\text{word}} = \alpha_1 \alpha_2 \alpha_3 \alpha_4 \dots \alpha_n$

$\alpha_i \in \Sigma = \{0,1\}$

$\{\text{word}\}_0^\infty$ word ≡ string

$|w| = n$, # of different words

$|W| = 10$ - all words length 10

int32
int64
uint10

$\{ w \mid |W| = 10 \}$

$\Sigma = \{0, 1\}$

1) 00000 00000
2) 00000 00001
3) 00000 00010

$\vdots$ $\vdots$

k) 11111 11111

$\# w \sim 2^n$, $n = |w|$

$\{012\} \Rightarrow 3^n$

Lemma: $\#$ all words $|w| = n$

$= |\Sigma|^n \qquad \forall n$

proof by induction

$n = 1 \quad 2 \quad 3 \quad 4 \quad \ldots$

$n = 1, \ \Sigma = \{a, b\} \hookrightarrow W_1 = \{a, b\}, \ |W_1| = 2 = 2^1$

$n = k \quad \hookrightarrow W_k = \{w_1^k, w_2^k, \ldots, w_{2^k}^k\}, \quad w_i^k \neq w_j^k \quad \forall i \neq j$

$n = k+1$

$W_{k+1} = \{w_1^k{}^\wedge a, \ w_1^k{}^\wedge b, \ldots\}$

$\alpha_1 \alpha_2 \ldots \alpha_k \beta, \ \beta \in \Sigma$

$\forall i \quad w_i^k \Rightarrow \begin{cases} w_i^k a \\ w_i^k b \end{cases}$

2 new words for each one old word

aabaa

aabab

aa < aaa

aa < aab

aab < aaba

aab < aabb

aaba < aabaa, aabab

ab < aba

ab < abb

ba < baa

ba < bab

bb < bba

bb < bbb

4 words => 8 words => 16 words => 32 word =>

$\frac{8}{4} = 2$

$\frac{16}{8} = 2$

$\frac{32}{16} = 2$

64

128

256

512

$1024 = 2^{10} \approx 10^3$

2048

$$\# \text{new words} = 2 \cdot \# \text{old words}$$

$$\# \text{new words} = 2 \cdot 2^k = \boxed{2^{k+1}}$$

$$\{word\}_0^\infty \longrightarrow \{0, 1\}$$

this function defines a Language

finite or infinite

Language - set of $|$words$|$ that belong to it.

Language can include words of different length.

Examples of languages:

1: $\underbrace{aabb}$ string length = 4

$\{0,1\} \equiv \{a,b\}$

$L = \{\text{"}aabb\text{"}\}$ – language that has only one word

2. $L = \{\text{"}ab\text{"}, \text{"}ba\text{"}\}$

3. $L = \{a, b\} = \{\alpha \mid \alpha \in \Sigma\}$

all possible words length = 1

4. $L = \{\varepsilon\}$

ε – word that contains no symbols

5. $L = \{\emptyset\}$

What problem are we going to solve?

given a language and a word figure out if this word belongs to te language

$$f(word) = \begin{cases} 0 - \text{if } w \notin L \\ 1 - \text{if } w \in L \end{cases}$$

example 1: bad language filter for group chats.

example 2: find all occurences of a word in a book

example 3: Compiler.

encodes, corresponds to

Each binary string generates a subset of a set (of size N).

The opposite is True as well, for each subset there exists a binary string that generates this subset.

We say that there exists isomorphisme between set of all possible binary strings (of length N) and set of all possible subsets of finite set of size N.

M - our set

$N = |M|$

The above statement is True for infinite sets too.

For finite case: $2^N \longleftrightarrow$ {all possible subsets of M}

For both finite and infinite case we write it the following way:

$$\{\text{all possible subsets of } M\} \equiv 2^M$$

Example: every Language is a subset of a set of all possible words in Σ

Regular Expressions

$$A, B \subseteq \{w \mid w \in \Sigma^*\} = 2^{\Sigma^*}$$

2^M - set of all subsets of M

$|M| = n$

$$\begin{array}{cccc} 1 & 2 & \ldots & n \\ 0 & 0 & & 0 \\ 1 & 1 & & 1 \end{array}$$

$0 \quad 1\,0\,1\,0\,1 \ldots$

$\boxed{2 \quad 4 \quad 6}$

$00 \ldots 001$

$00 \ldots 010$

$00 \ldots 011$

$\vdots$

$11 \ldots 111$

A, B - some regular Languages $\quad A, B \subset REG$

1. Union

$$\forall A, B \quad A \cup B = \{w \mid w \in A \text{ or } w \in B\}$$

2. Concatenation

$$A \circ B = \{xy \mid x \in A, \; y \in B\}$$

3. Star (a.k.a. Iteration)

$$A^* = \{x_1 x_2 \ldots x_k \mid \text{each } x_i \in A \text{ for } k \geq 0\}$$

$k=3 \longrightarrow X_1 X_2 X_3$ - 3 words

$k=0 \longrightarrow \varepsilon$

$A=\{\emptyset\}$ $\quad$ $A^* = \cancel{\{\emptyset\}}$ $\quad$ $A^* = \{\varepsilon\}$

By definition, empty Language is considered to be Regular.

$L = \{\emptyset\}$

By definition, Languages that consist of one symbol from Alphabet
are considered to be Regular. (word has length one)

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

By definition, All Languages that can be derived from Regular Languages
by applying Regular Operations are considered to be Regular too.

$L_1 = \{a\}, L = L_1^* \subset REG$ | $(\{a\}\{b\})^* \cup \{b\}$

ε $\quad (\underset{\{a\}\cdot\{a\}}{\underset{\parallel}{aa}} \mid \underset{\{b\}}{\underset{\parallel}{b}})^* \varepsilon (a \mid aa)$

$\varepsilon \in REG$

Let A be a Lagnuage that consists of finite number of words.
Is A Regular or Irregular?

Can we prove it?

Statement 1: any word alone represents a Regular Language.

$W = \alpha_1 \alpha_2 \alpha_3 \ldots \alpha_n$

$L_1 = \{\alpha_1\}, L_2 = \{\alpha_2\}, \ldots, L_n = \{\alpha_n\}$

$L_w = \underbrace{L_1 \cdot L_2 \cdot \ldots \cdot L_{n-1}}_{L_{n-1}} \cdot L_n = \{\alpha_1 \alpha_2 \ldots \alpha_n\}$

Statement 2: $L = \{w_1, w_2, \ldots, w_k\} = L_{w_1} \cup L_{w_2} \cup \ldots \cup L_{w_k}$

What happens if we apply Regular Operations infinite number of times?

$\cup\ ,\quad \circ\ ,\quad {}^{*}$

K - some fixed number

$$\bigcup_{n=1}^{K}\{a^n\} \equiv a \cup aa \cup aaa \cancel{\cup \ldots \cup a^K}$$

REG

REG

$a \mid aa \mid aaa \in REG$

$a \mid a^2 \mid a^3 \mid$

$\{aa\} \overset{?}{\in} REG$

$\{a\}\circ\{a\} \equiv \{aa\}$

aaaa

Proof by construction

$\{aba\} \overset{?}{\in} REG$

$\{a\}\circ\{b\}\circ\{a\} \equiv \{aba\}$

$(\{a\}\circ\{b\})\circ\{a\} = \{ab\}\circ\{a\} = \{aba\}$

$\boxed{x}\circ\boxed{y} \quad f(x,y)$

$513 \times$

$\times 271 \times$

$\times\ 380$

$$\begin{array}{r} 513 \\ +\ 35910 \\ +\ 102600 \\ \hline 139023 \end{array}$$

$a \times b \times c = (a \times b) \times c =$

$= d \times c = e$

$\{a|b\} \overset{?}{\in} REG$

$L=\{a,b\}$

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

$\{aa|ba\} \overset{?}{\in} REG$

$L=\{aa,ba\}=$

$=\{aa\}\cup\{ba\}$

$\{a\}\circ\{a\}$ $\{b\}\circ\{a\}$

$\{ab|b|aaa\} \overset{?}{\in} REG$

$L=\{ab,b,aaa\}=$

$=(\{ab\}\cup\{b\})\cup\{aaa\}=$

$=\{ab|b\}\cup\{aaa\}$

$\{bb|ba|bab|aaa\}$

$L=\{bb|ba|bab\}\cup\{aaa\}$

$=L_3\cup\{aaa\}=L_3\cup L_1=L_4$

$L_5=\{w_1|w_2|w_3|w_4|w_5\}=$

$=\{w_1|w_2|w_3|w_4\}\cup\{w_5\}=$

$=L_4\cup L_1$

$L_{100}=L_{99}\cup L_1$

$$\bigcup_{k=1}^{n} w_k = w_1 \cup w_2 \cup w_3 \cup \dots \cup w_{n-1} \cup w_n$$

n – fixed number

Formally speaking we define a family of sets parametrized by index "k", then we unite by this index

$$\bigcup_{k=1}^{\infty} w_k = w_1 \cup w_2 \cup \dots \cup w_n \cup \dots$$

$$\bigcup_{k \in \mathbb{N}}$$

Infinite Union ALWAYS yields a Regular Language. True or False?

$$\forall L \Rightarrow L = \bigcup_{w \in L} \{w\}$$

Any Language is a union of words that make this language

Poperty that either holds or not holds for finite/infinite cases.

Example why proof by induction is not always applicable

Set is bounded if:

$M = \{m_1, m_2, \dots\}, \quad m_i \in \mathbb{R}$

satisfied
violated

$\exists x \in \mathbb{R} : \forall m \in M \hookrightarrow |m| \leq x$

$\overline{\text{example}}$

Bounded ~ There exists Maximum Element

$M = \{1, 2, 3, 4, 5\}$ $\quad x = 5, \quad x = 6$

$x = 5{,}5$

Any set that contains finite number of elements IS bounded. True or False?

Base

1. $N = |M| = 1 \; ; \; M = \{m_1\} : |m_1| \leq |m_1| + 1 \equiv X$

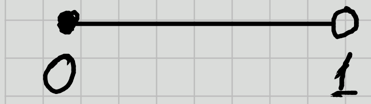
step

2. Assume $N = |M| = n$ statement is true

$M_{n+1} = M_n \cup \{m_{n+1}\} ; \; \exists X_n : \forall m \in M_n \hookrightarrow |m| \leq X_n$

$X_{n+1} = \max\{X_n, |m_{n+1}|\}$

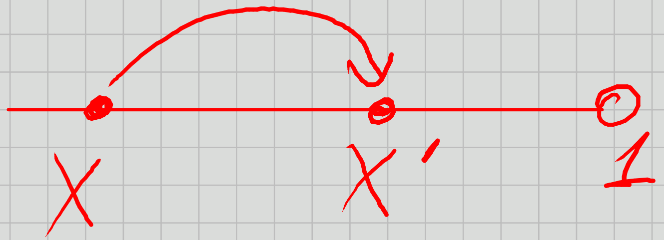
$[0;1) \equiv [0;1] / \{1\}$ $\mathrm{Max}[M] \in M$



$x = 0{,}999$

$\Delta = 1 - x = 0{,}001$

$x' = x + \frac{1}{2}\Delta = 0{,}999 + 0{,}0005 =$
$= 0{,}9995$



Set has infinite number of elements $\overset{?}{\Longrightarrow}$ Bounded

~~We know that every infinite set IS bounded.~~ $M = N$

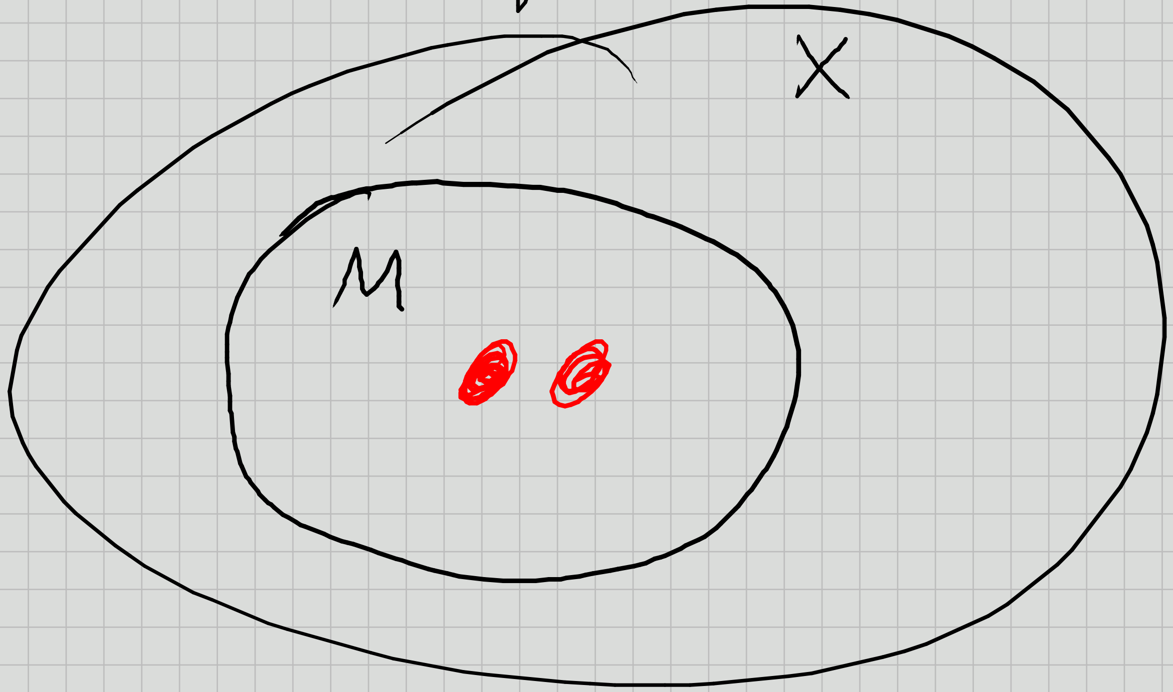
X - entire space

M

$f(\cdot)$ $\quad x \longrightarrow X$

$x, x \longrightarrow X$

$x, x, x \longrightarrow X$



$f: M \to ?$

$M \to M$

$M \to X$

$L_1 \cup L_2$ $\qquad \cup = f(\cdot, \cdot) = f(x, y) \to z$

$\forall x, y \in M \Rightarrow f(x, y) \in M$

f is closed in M; M is closed under f

$M, |M| = \infty \implies M$ not bounded

$[0;1] \subset \mathbb{R}$

Файнайт
йнфинит

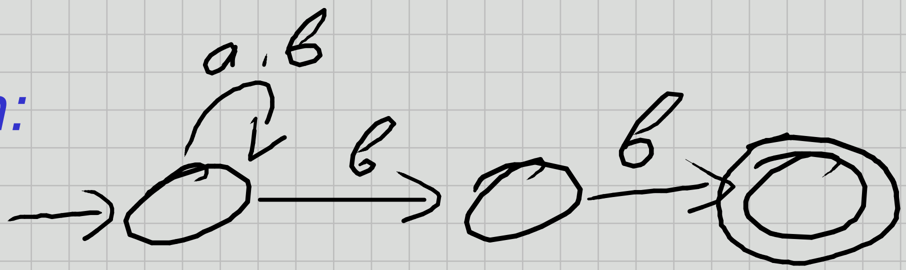
Expression
Language
Automaton
Grammar

Technically these are different

Expression: $(a | b)^* bb$

Language: $\{ w \mid \text{ends with "}bb\text{"} \}$

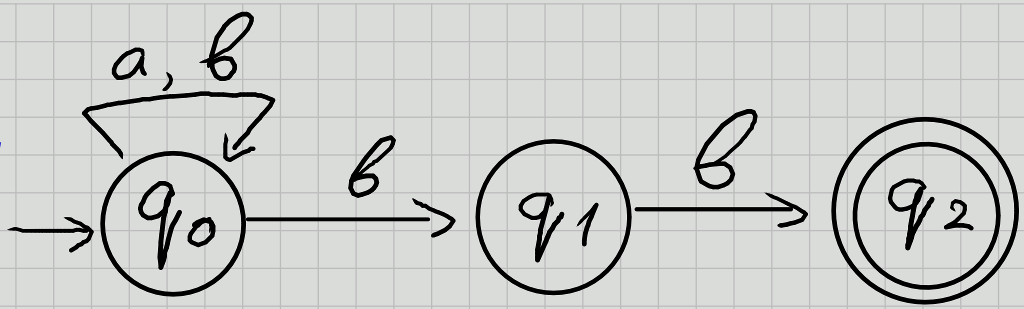
Automaton:



Grammar:

$$S \rightarrow aS \mid bS \mid A$$
$$A \rightarrow bb$$

Automaton:



aabb ✓

bab ✗

abba

Finite Automaton $(Q, \Sigma, q_0, F, \delta)$

$Q = \{q_1, \dots, q_n\}$ – Finite set of all states

$\Sigma = \{a, b, \dots\}$ – Alphabet

δ – Transition Matrix $\delta : Q \times \Sigma \to Q$

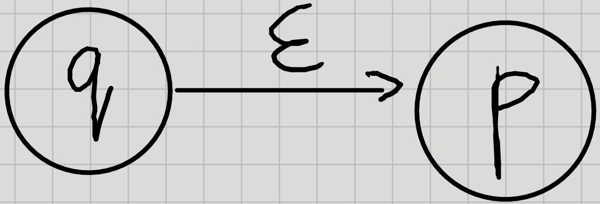
q_0 – starting pos.

F – accept states

Automaton accepts a word: we say automaton accepts a word if there exists a path from "starting state" to some "accepts state" that consumes the word and where each transition qi->qj is accomponied by reading the next symbol of the word.

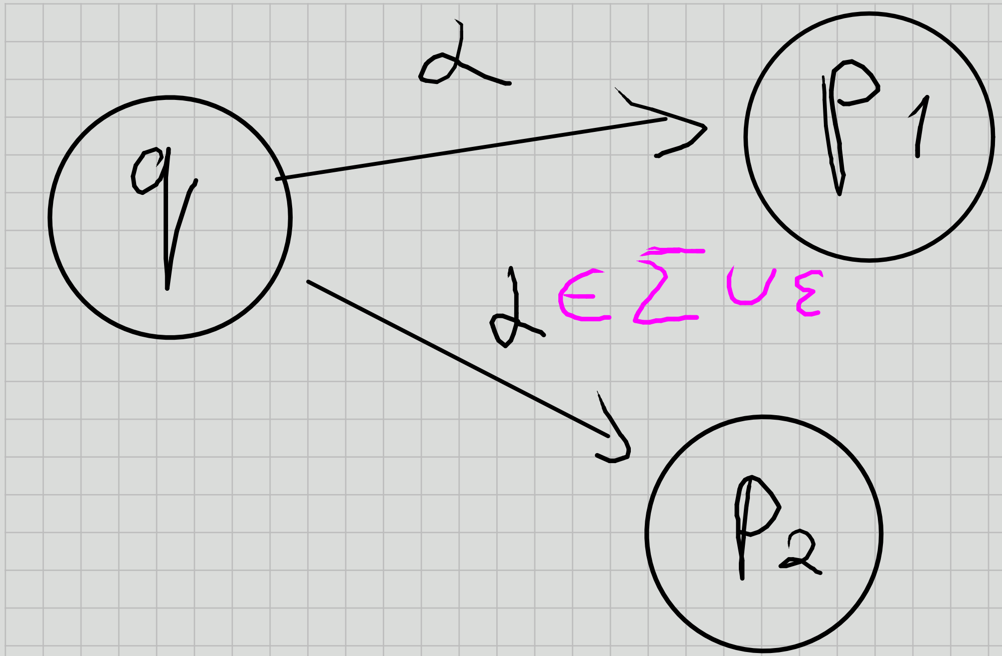
$$\delta(q_i, \alpha) = q_j$$

Empty-word transition: $\delta(q^{\in Q}, \varepsilon) = p^{\in Q}$



Empty word transitions don't require reading any symbols from the word and don't consume any symbols of the word.

Non-deterministic transitions



$$\forall q \in Q, \forall a \in \Sigma$$

$$\boxed{|\delta(q, a)| \leq 1}$$

Automatons can be deterministic and non-deterministic
DFA - Determ. Finite Automaton
NFA - Nondeter. Finite Automaton
FA - Finite Automaton (any)

$$\delta : Q \times \Sigma \to 2^{Q}$$

$L \subset REG$

DFA NFA

$L(M) = \{ w \mid M \text{ accepts } w \}$

$M = DFA \iff L(M) \subset REG$

$M = NFA \iff L(M)$?

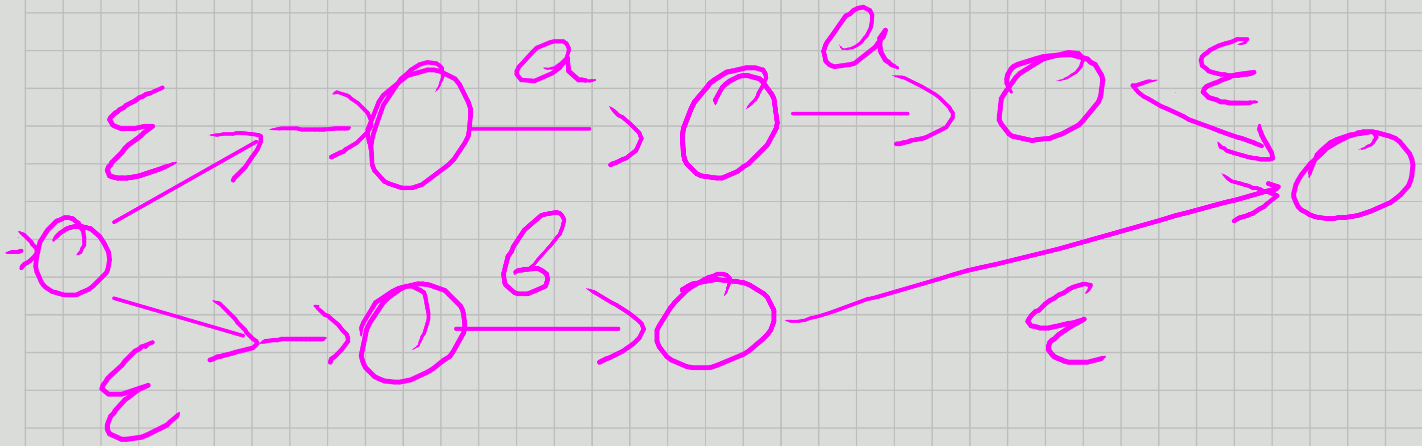
$$A(w) \longrightarrow \{0;1\}$$

$$\{w \mid A(w) = 1\} = L(A)$$

РВ → ДКА (rus)
RE → DFA (eng)

$$\underset{0}{\triangleright}\,(\underset{1}{a}\underset{2}{a}|\underset{3}{b})^*\underset{4}{b}\,(\underset{5}{a}|\underset{6}{b})^*\underset{7}{a}\,\underset{8}{\triangleleft}$$

$$\underset{0}{\triangleright}\,\underset{1}{a}\underset{2}{a}bbb\underset{5}{a}\underset{7}{a}\,\underset{8}{\triangleleft}$$



$\Longleftrightarrow (aa|b)$

firstpos, lastpos, nullable, followpos

$RE = \{w\}$ – finite or infinite

$RE = \underset{0}{a}\underset{1}{a}|\underset{2}{b}$ $\{aa, b\}$ $fp = \{0,2\}$ $lp = \{1,2\}$

firstpos - set of symbols that can be the first symbol of words of the language

$$\{\alpha \mid \alpha x \in L;\ \alpha \in \Sigma,\ x \in \Sigma^*\}$$

lastpos - ...

$$\{\alpha \mid x\alpha \in L;\ \ldots\}$$

nullable - $\varepsilon \overset{?}{\in} L$

$$\begin{cases} 1, & \varepsilon \in L \\ 0, & \text{otherwise} \end{cases}$$

followpos -

$\alpha, \beta \in \Sigma$

$$\text{followpos}[\alpha] = \{\beta \mid x\alpha\beta y \in L;\ x, y \in \Sigma^*\}$$

$RE = (\underset{0}{a}\underset{1}{b}\underset{2}{a} \mid \underset{3}{b}\underset{4}{a}\underset{5}{b} \mid \underset{6}{b}\underset{7}{a}^{*})$ $\Sigma = \{a, b\}$

$fp = \{ id \mid (\{id\}\, w) \in L \}$ $\Sigma_{id} = \{0, 1, 2, \ldots, 7\}$

$L = \{\underset{0\,1\,2}{aba}, \underset{3\,4\,5}{bab}, \underset{6}{b}, \underset{6\,7}{ba}, \underset{6\,7\,7}{baa}, \underset{6\,7\,7\,7}{baaa}, \ldots\}$

$L = \{\boxed{0}1\boxed{2}, \boxed{3}4\boxed{5}, \boxed{\boxed{6}}, \boxed{6}\boxed{7}, \boxed{6}7\boxed{7}, \ldots\}$

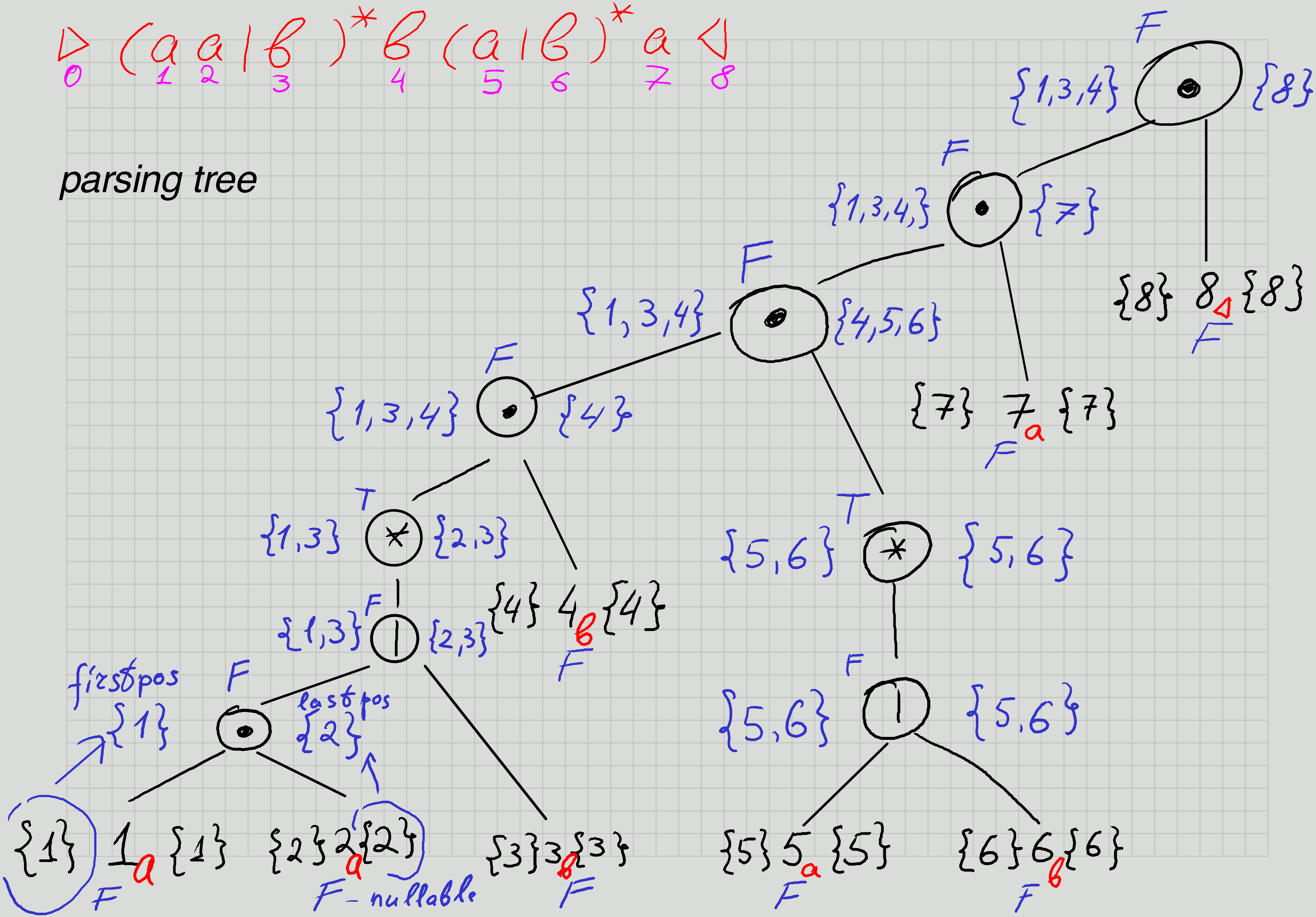
$fp = \{0, 3, 6\}$ $nullable = false$

$lp = \{2, 5, 6, 7\}$

$followpos[i] = \{ j \mid x\{i\}\{j\}y \in L \}$

0	1	3	4	6	◁, 7
1	2	4	5	7	◁, 7
2	◁	5	◁		

parsing tree



followpos table

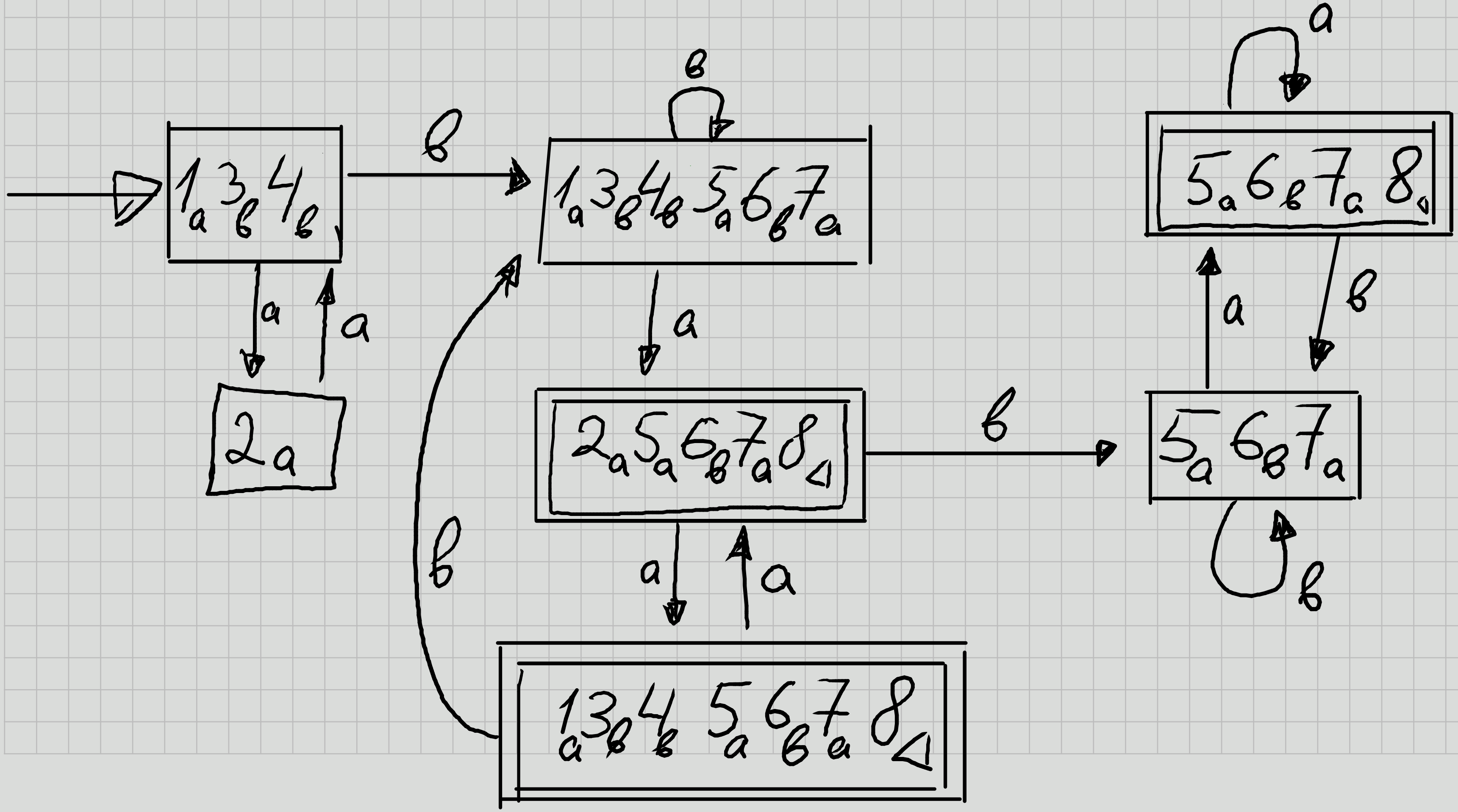
$0_{\triangleright}$	$1_a\ 3_b\ 4_b$
1_a	2_a
2_a	$4_b\ 1_a\ 3_b$
3_b	$4_b\ 1_a\ 3_b$
4_b	$5_a\ 6_b\ 7_a$
5_a	$7_a\ 5_a\ 6_b$
6_b	$7_a\ 5_a\ 6_b$
7_a	$8_{\triangleleft}$

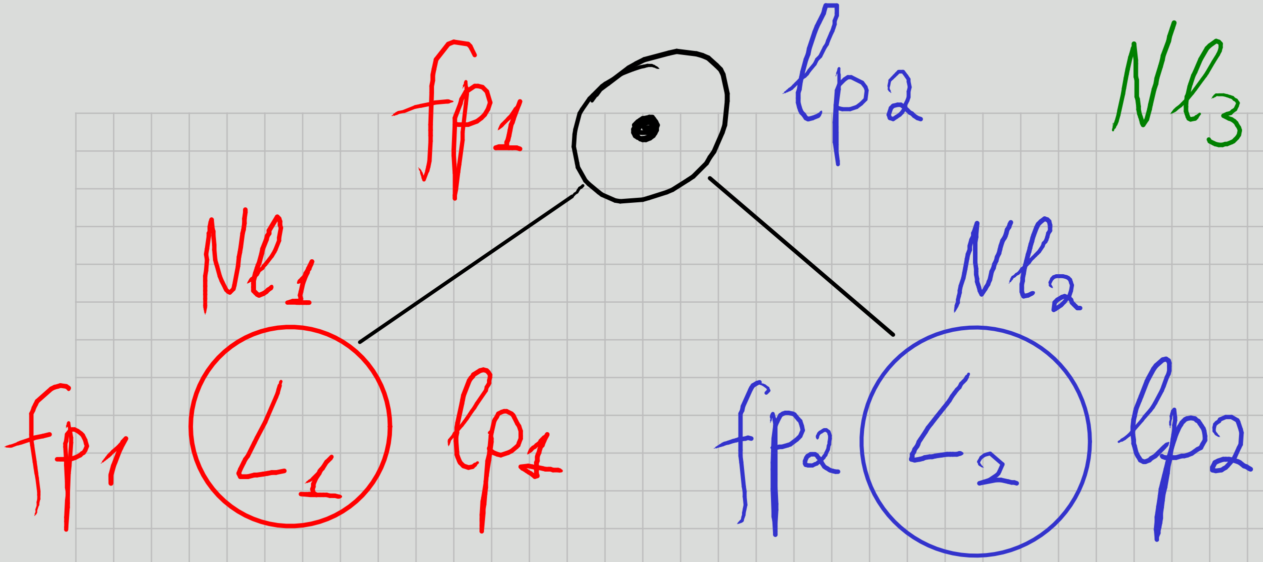
$\underset{0}{\triangleright}\ (\underset{1}{a}\,\underset{2}{a}\,|\,\underset{3}{b})^*\,\underset{4}{b}\,(\underset{5}{a}\,|\,\underset{6}{b})^*\,\underset{7}{a}\,\underset{8}{\triangleleft}$

followpos table

$0_\triangleright$	$1_a\ 3_b\ 4_b$
1_a	2_a
2_a	$4_b\ 1_a\ 3_b$
3_b	$4_b\ 1_a\ 3_b$
4_b	$5_a\ 6_b\ 7_a$
5_a	$7_a\ 5_a\ 6_b$
6_b	$7_a\ 5_a\ 6_b$
7_a	$8_\triangleleft$

aabbbaa





$$Nl_3 = Nl_1 \wedge Nl_2$$

$$fp_3 = \begin{cases} fp_1 , & Nl_1 = F \\ fp_1 \cup fp_2 & T \end{cases}$$

$$lp_3 = \begin{cases} lp_2 , & Nl_2 = F \\ lp_1 \cup lp_2 & T \end{cases}$$

$$L_1 = \{ fp_1 \cdot x \cdot lp_1 \} \quad L_2 = \{ fp_2 \cdot y \cdot lp_2 \}$$

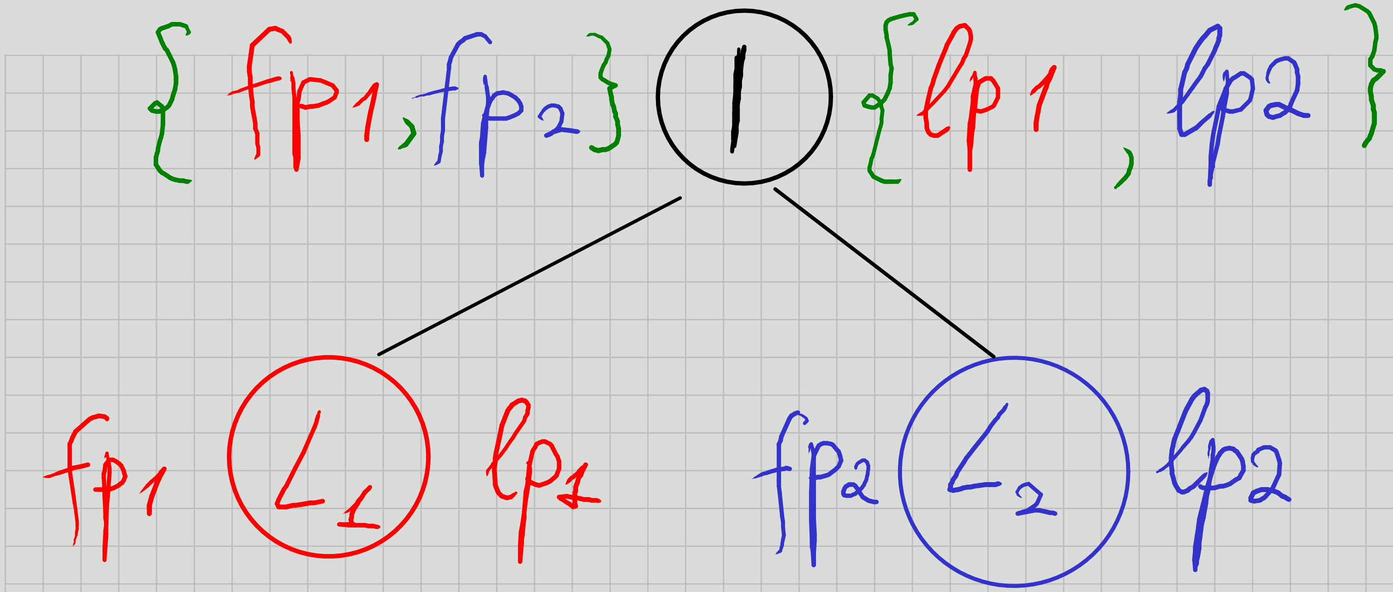
$$L_3 = L_1 \cdot L_2 = \{ \underbrace{fp_1}_{} \cdot x \cdot lp_1 \; fp_2 \cdot y \cdot \underbrace{lp_2}_{} \}$$

$$w_1 w_2 = \varepsilon \iff \begin{cases} w_1 = \varepsilon \\ w_2 = \varepsilon \end{cases}$$

both words have to be empty

$$\varepsilon \in L_1 \quad \hookrightarrow L_1 \cup \{ \varepsilon \} = L_1$$

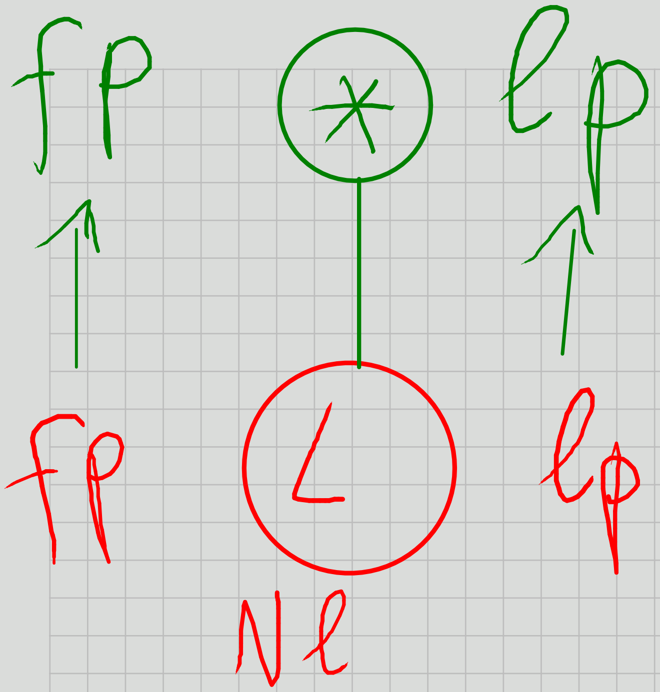
$$L_1 L_2 = (L_1 \cup \{ \varepsilon \}) L_2 = L_1 L_2 \cup \{ \varepsilon \} L_2 = L_1 L_2 \cup L_2$$



$$N\ell_3 = N\ell_1 \vee N\ell_2$$

$L_1 = \{ fp_1 \cdot x \cdot \ell p_1 \}$ $L_2 = \{ fp2 \cdot y \cdot \ell p2 \}$

$L_3 = L_1 | L_2 = \{ fp_1 \cdot x \cdot \ell p_1 , fp2 \cdot y \cdot \ell p2 \}$



$$L^* = \varepsilon \cup L \cup LL \cup LLL$$

$Nl = True$

$$LLL = \{\underbrace{fp\ x\ lp}, \underbrace{fp\ y\ lp}, \underbrace{fp\ z\ lp}\}$$

$L_n = \Sigma^* b \Sigma^n$

$P < 2^n$

$aaa \ldots a$
$baa \ldots a$
$aba \ldots a$

Language that consists of all words that have symbol "b" n positions before the last.

Lemma:

DFA that recognizes this language has at least 2^n states.

$$((abb|b)^* | b^*)aa$$

$$\underset{0}{\triangleright}\Big[\big(\underset{1}{a}\,\underset{2}{b}\,\underset{3}{b}\,|\,\underset{4}{b}\big)^*\,\Big|\,\underset{5}{b}^*\Big]\,\underset{6}{a}\,\underset{7}{a}\,\underset{8}{\triangleleft}$$

followpos

0	$1_a\,4_b\,5_b\,6_a$
1_a	2_b
2_b	3_b
3_b	$1_a\,4_b\,6_a$
4_b	$1_a\,4_b\,6_a$
5_b	$5_b\,6_a$
6_a	7_a
7_a	$8_\triangleleft$

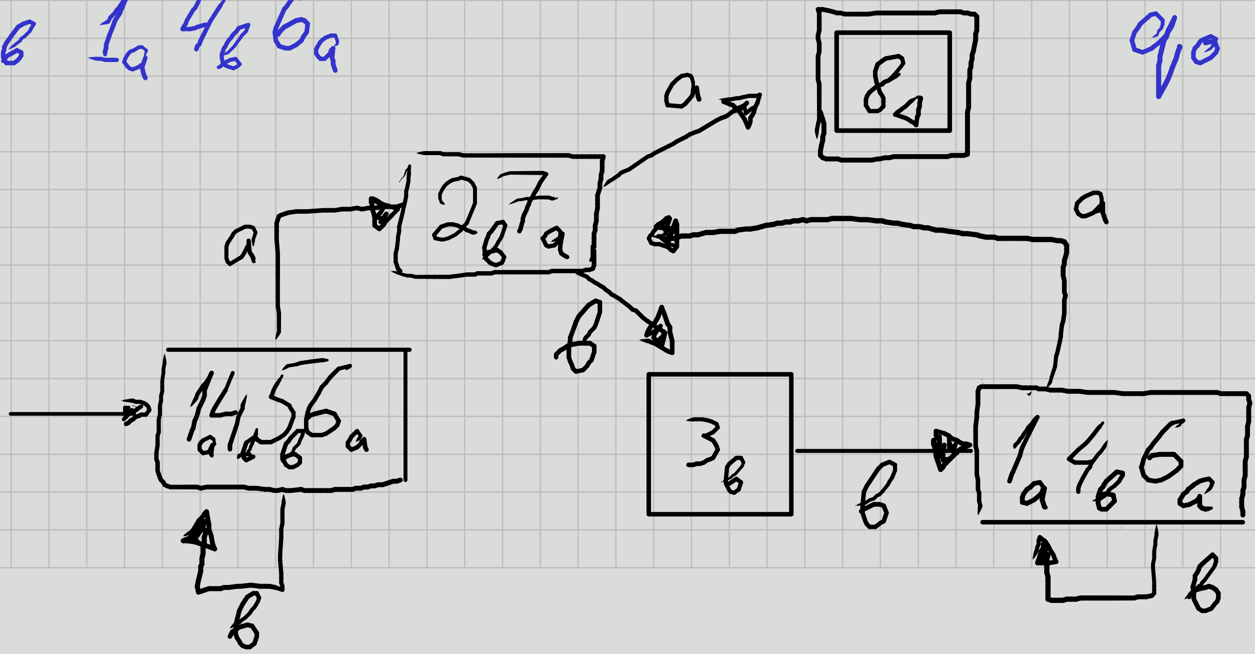
$\longrightarrow abb$

$abbabb$

$q_0 \overset{abb}{\rightsquigarrow} 146$

$q_0 \overset{abbabb}{\rightsquigarrow} 146$

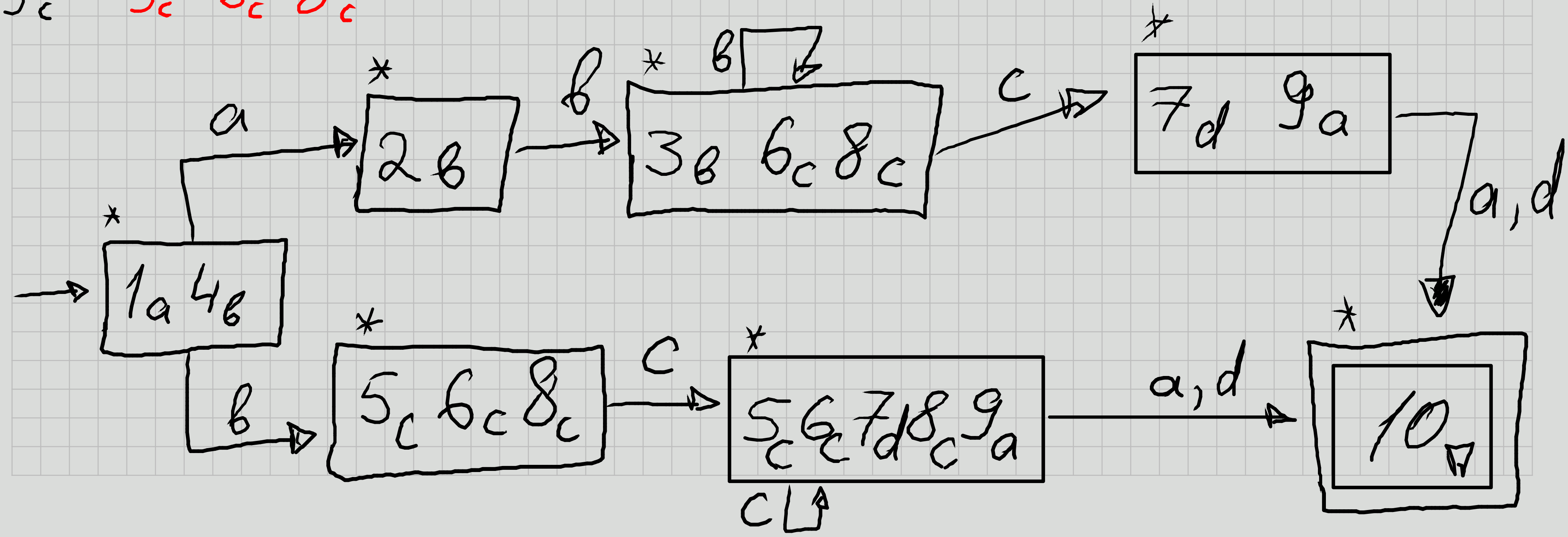
$q \overset{w}{\rightsquigarrow} p$



$0_\triangleright$	1_a	4_b	
1_a	2_b		
2_b	3_b	6_c	8_c
3_b	3_b	6_c	8_c
4_b	5_c	6_c	8_c
5_c	5_c	6_c	8_c

6_c	7_d
7_d	$10_\triangleleft$
8_c	9_a
9_a	$10_\triangleleft$

a b c d



$a^n b^n$ ww ww^R

$|\cdot|_a = |\cdot|_b$ $\cancel{\notin} REG$

---

a^n b^n is not regular.
Proof:

все переходы однозначные

F - set of accepting states

Lets assume the language IS regular. Then there exists DFA that recognizes this language. DFA has finite number of states, lets say it has "p" states.

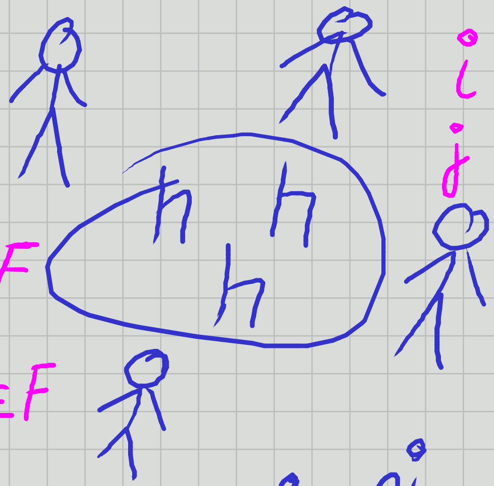
$w_k = a^k, k \geq 1$

$$p+1 \left\{ \begin{array}{l} a \\ aa \\ aaa \\ \vdots \\ a \ldots a_{p+1} \end{array} \right.$$

$q_0 \xrightarrow{w_i} r \xrightarrow{b^i} r' \in F$

$q_0 \xrightarrow{w_j} r \xrightarrow{b^i} r' \notin F$

$i \neq j$ $i < j$



i aa bb
j aaaa bb

$a^i b^i \in L(A)$
$a^j b^i \notin L(A)$

RE → NFA

RE --> DFA
NFA --> DFA
minimization of DFA
NFA --> RE

Grammars
classification (0-3)
NFA <--> Type(3) G
PDA <--> Type(2) G
TM <--> Type(0,1)

Inductive construction of NFA

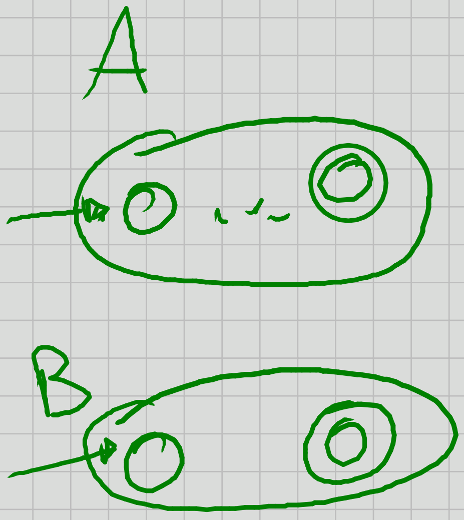
Union, Concatenation, Star

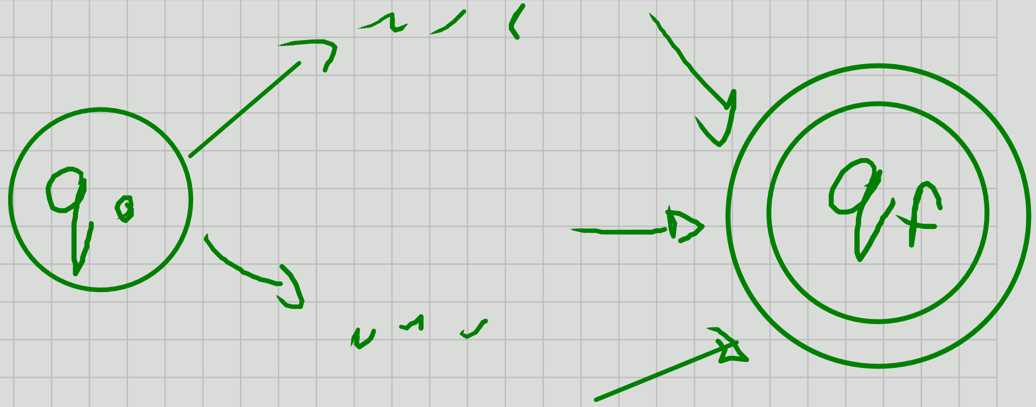
A, B

$A \circ B$

$A | B$

A^*





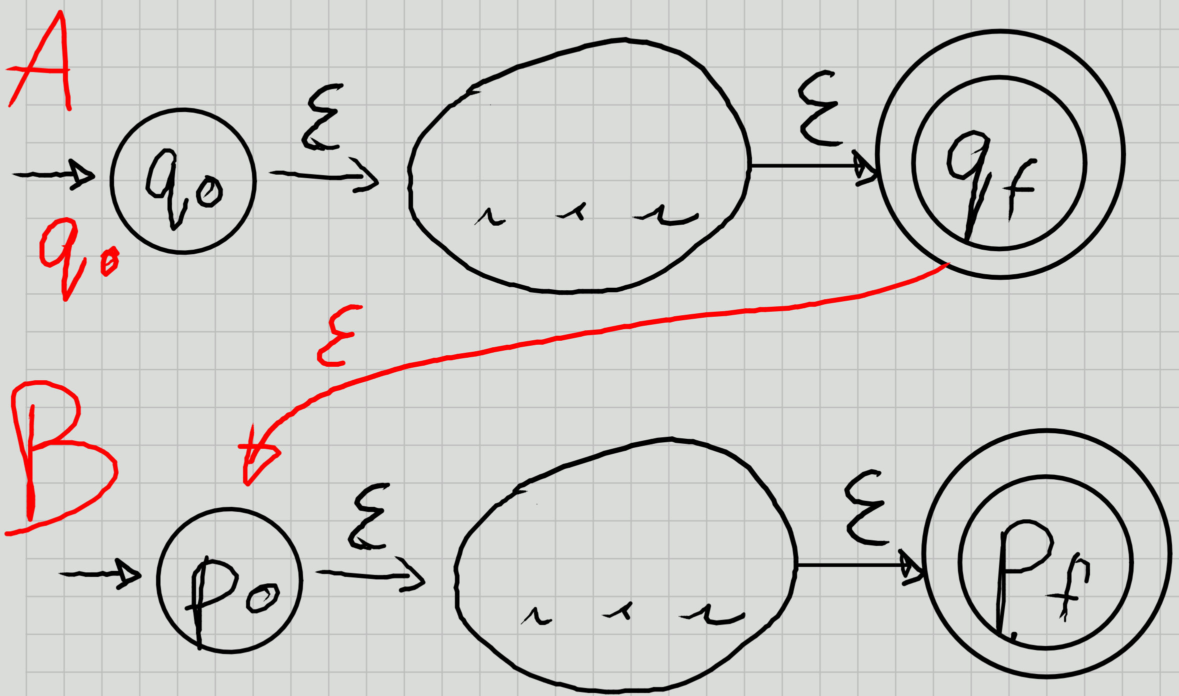
1. |F| = 1

2. input power of "q_0" is zero

3. output power of "q_f" is zero

Concatenation

$C = AB \; ; \; W_C = W_A W_B$



A, B, C - automatons

$L(A), L(B), L(C)$

$L(C) \supset L(A) \circ L(B)$

1) $\forall w \in L(A) \circ L(B) \overset{?}{\Longrightarrow}$

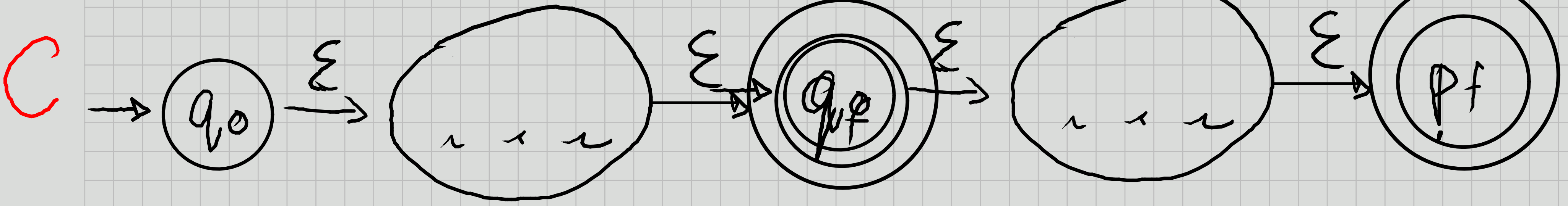
$\underline{w \in C} : \exists\, r_0 r_1 r_2 \ldots r_n :$

$r_i \to r_{i+1} \quad w_j \; \varepsilon$

$r_0 = q_0 \; , \; r_n = p_f$

$w \in L(A) \circ L(B) \Rightarrow w = w_A w_B \, ,$ $w_A \in L(A) \Longrightarrow \exists\, q_0 \ldots q_f : q_0 \xrightarrow{w_1} q_f$

$w_B \in L(B) \Longrightarrow p_0 \xrightarrow{w_B} p_f$



2) $L(C) \subset L(A) \circ L(B)$

$\forall w \in C \overset{?}{\Longrightarrow} w \in L(A) \circ L(B)$

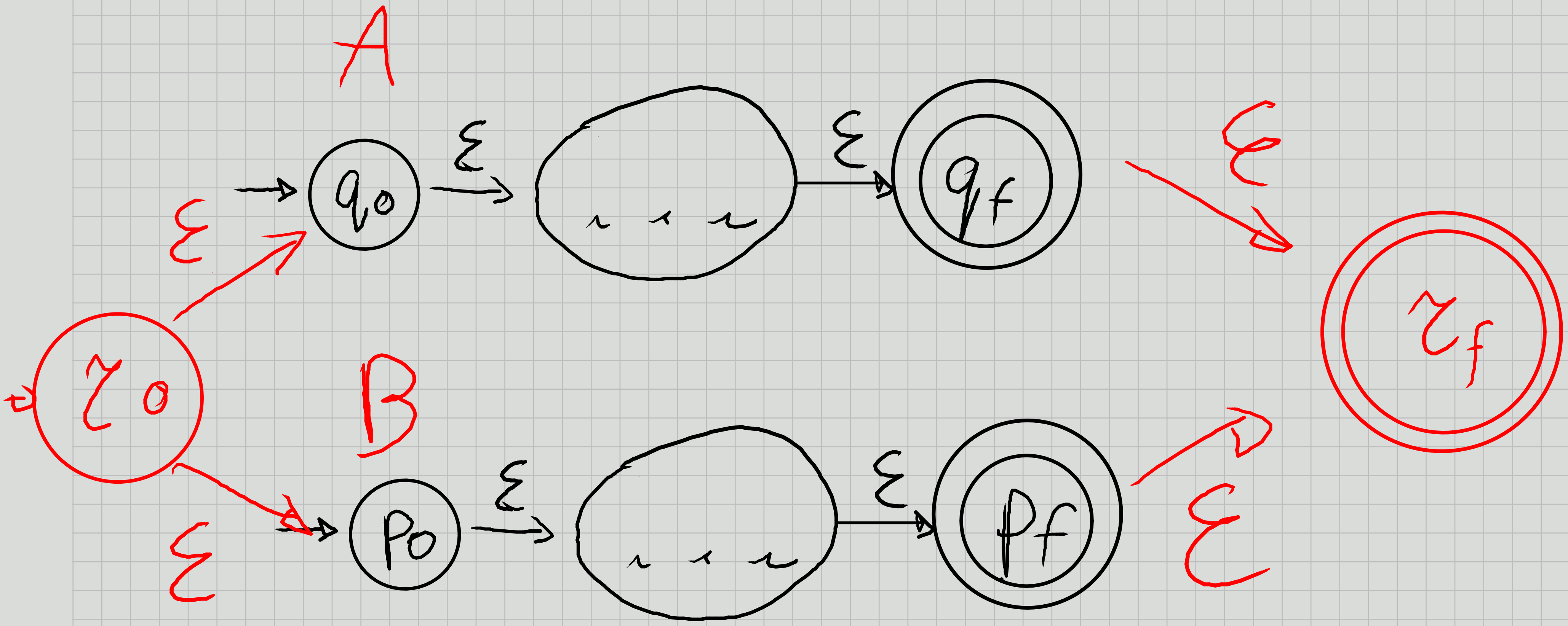
$w \in C \Rightarrow q_0 \xrightarrow{w} p_f \Longleftrightarrow q_0 \to r_1 \to r_2 \to \ldots \to r_n \to p_f$

We know that automatons A and B do not intersact $\exists\, i : r_i = q_f \, , \, r_{i+1} = p_0$

$q_0 \rightsquigarrow q_f \xrightarrow{\varepsilon} p_0 \rightsquigarrow p_f$

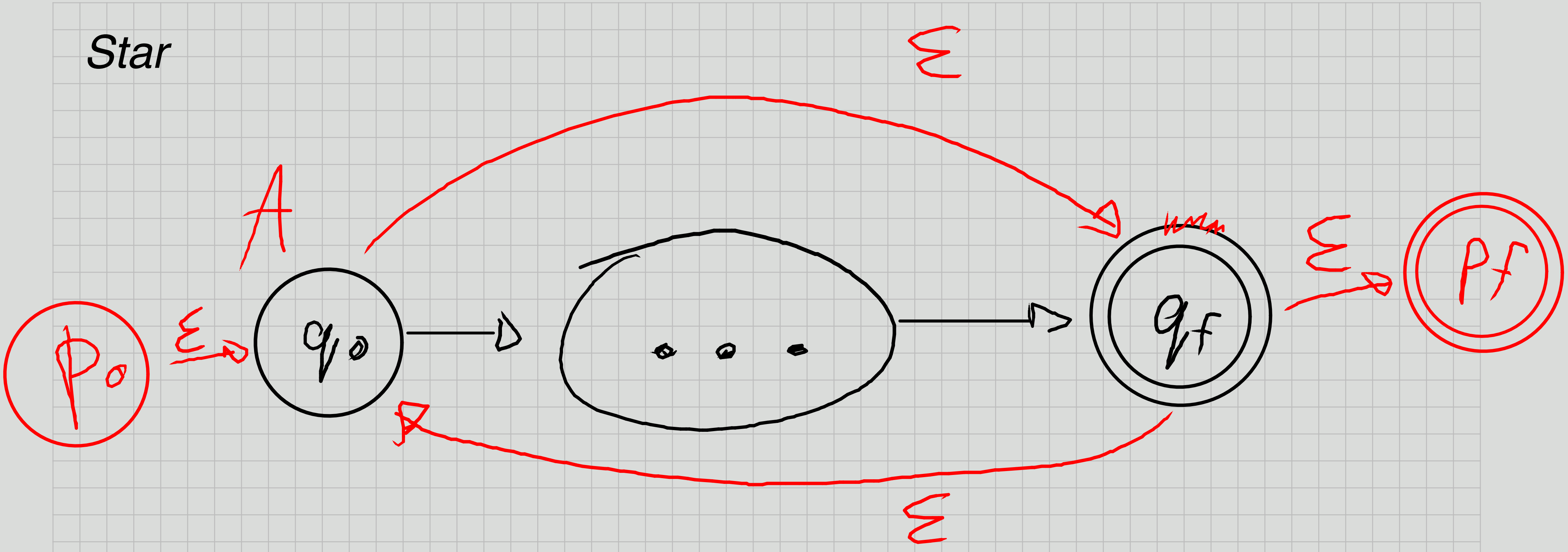
Union

$C = A | B$



$r_0 \xrightarrow{\varepsilon} q_0 \rightarrow q_1 \rightarrow \dots \rightarrow q_f \xrightarrow{\varepsilon} r_f$

Star



$$\varepsilon \in A^* \Rightarrow \exists\, q_0 \xrightarrow{\varepsilon} \dots \xrightarrow{\varepsilon} q_f$$

Empty word is recognized by automaton => there exists a path from starting state to accepting state where all transitions are empty word transitions

NFA → DFA

$$\varepsilon\text{-closure}(q) = \{\, p \mid q \overset{\varepsilon}{\rightsquigarrow} p \,\}$$

$$\varepsilon\text{-cl}(q)$$

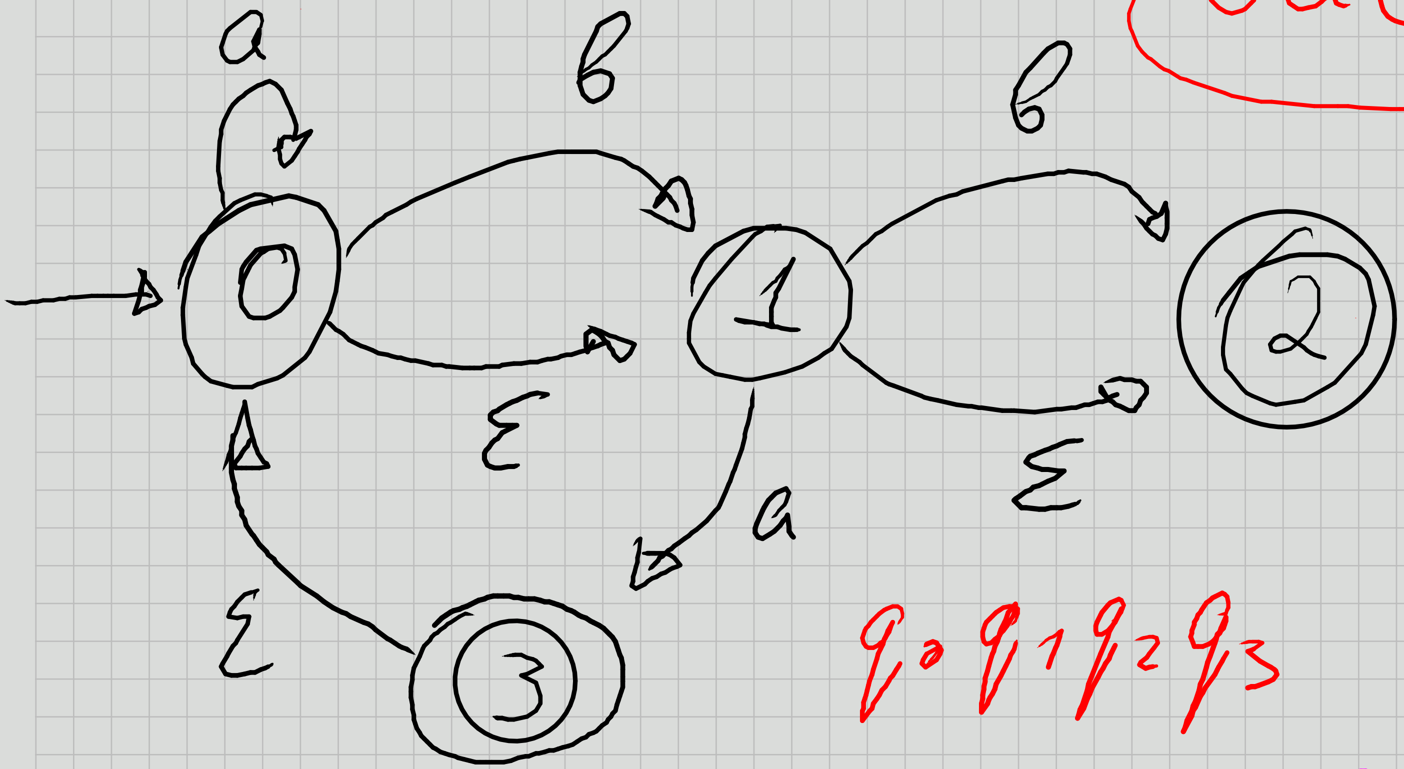
$$\varepsilon\text{-cl}(Q) = \bigcup_{q \in Q} \varepsilon\text{-cl}(q)$$

$Q_{DFA} = 2^{Q_{NFA}}$ - powerset

$Q_0 = \varepsilon\text{-cl}(q_0)$ $\quad F_{DFA} = \{\, Q_i \mid Q_i \cap F_{NFA} \neq \emptyset \,\}$

$$Q_i \xrightarrow{\alpha} Q_j = \varepsilon\text{-cl}\left(\{\, p \mid q \xrightarrow{\alpha} p,\ q \in Q_i \,\}\right)$$

bab



$\varepsilon\text{-}cl(0) = \{0,1,2\}$

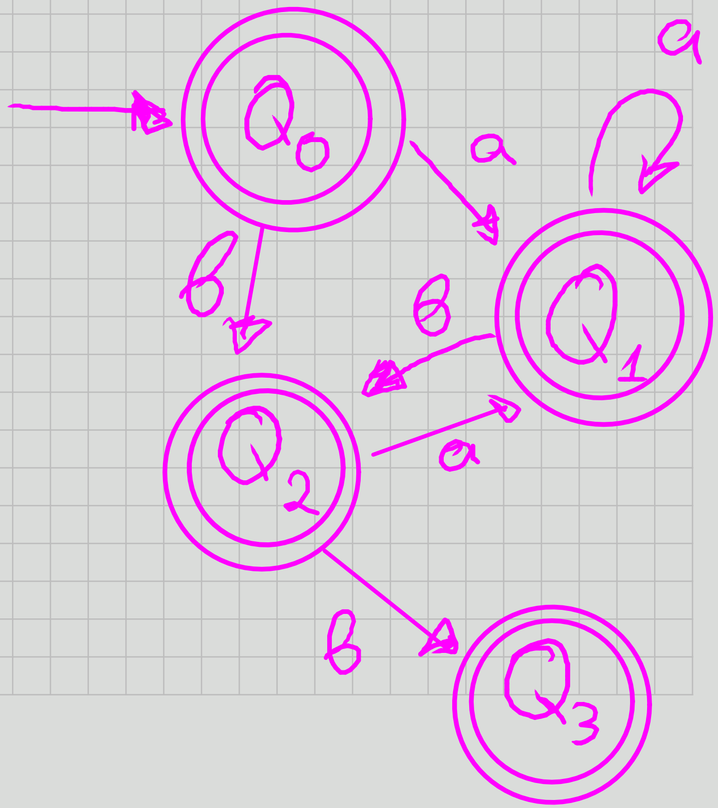
$\varepsilon\text{-}cl(1) = \{1,2\}$

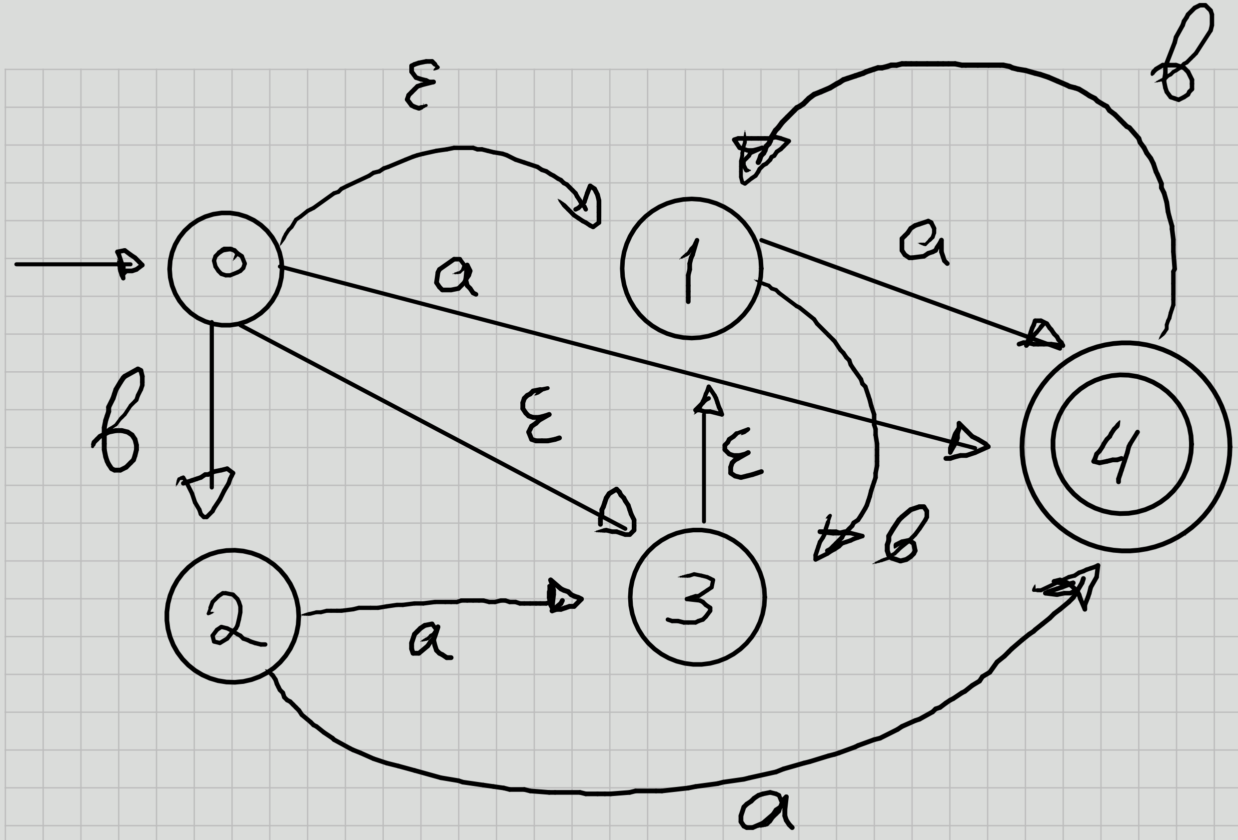
$\varepsilon\text{-}cl(2) = \{2\}$

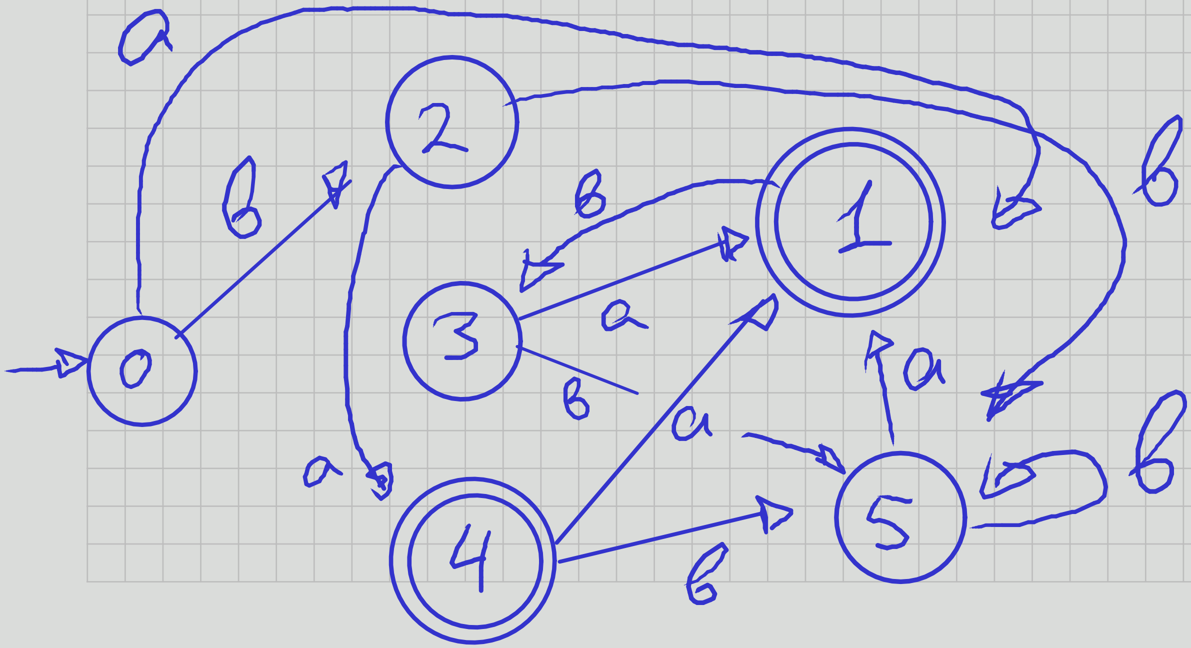
$\varepsilon\text{-}cl(3) = \{3,0,1,2\}$

$q_0\ q_1\ q_2\ q_3$

DFA state	NFA state	a	b
Q_0^*	$q_0^*\ q_1^*\ q_2^*$	Q_1	Q_2
Q_1^*	$q_0^*\ q_3^*\ q_1^*\ q_2^*$	Q_1	Q_2
Q_2^*	$q_1^*\ q_2^*$	Q_1	Q_3
Q_3^*	q_2^*	—	—





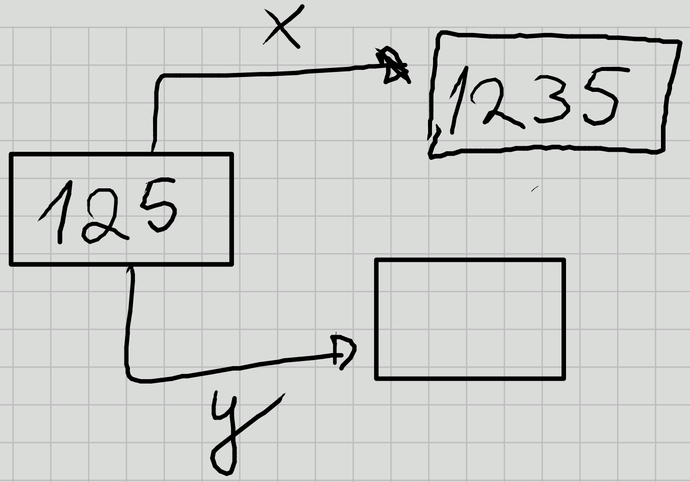


DFA state	NFA state	a	b
Q_0	$0^* 1^* 3^*$	Q_1	Q_2
Q_1^*	4^*	—	Q_3
Q_2	$2^* 3^* 1^*$	Q_4	Q_5
Q_3	1^*	Q_1	Q_5
Q_4^*	$3^* 4^* 1^*$	Q_1	Q_5
Q_5	$3^* 1^*$	Q_1	Q_5

2 4

0 1

5 3



	x	y	z
125	1235	67	—
1235	1235	67	47
47	7	—	4
67	7	—	6
7	7	—	—
4	—	—	4

▷ $x^*(x\,z^+ \mid y\,z^*)(x\,y)^+$ ◁

0 1 2 3 4 5 6 7 8

followpos

$0_\triangleright$	$1_x\ 2_x\ 4_y$
1_x	$1_x\ 2_x\ 4_y$
2_x	3_z
3_z	$3_z\ 6_x$
4_y	$5_z\ 6_x$
5_z	$5_z\ 6_x$
6_x	7_y
7_y	$6_x\ 8_\triangleleft$

δ

	x	y	z
$1_x 2_x 4_y$	$1_x 2_x 3_z 4_y$	$5_z 6_x$	—
$1_x 2_x 3_z 4_y$	$1_x 2_x 3_z 4_y$	$5_z 6_x$	3 6
$5_z\ 6_x$	7	—	5 6
$3_z 6_x$	7	—	3 6
7_y	—	6 8	—
$6_x 8_\triangleleft$	7	—	—

1. Построить регулярное выражение (РВ) для языка из слов, содержащих в качестве подслова слово aab

examples of words: aaabbb, aabb

baaba

baabbba

$$\boxed{\Sigma^* aab \Sigma^*}$$

2. Построить РВ для языка, слова которого не содержат подслово ab.

examples of words: ~~baba~~ ~~bbab~~ baaa

bbba~~b~~

bbbbaaa a^*

b^*

aaaaa

a^*

$$\boxed{b^* a^*}$$

3. Построить регулярное выражение для языка из слов, содержащих в качестве подслова ровно одно слово ab

$$ba\underbrace{ab}bbb \qquad aa\underbrace{ab}aaaaaa$$

We need to have "ab" inside our word. What can we add before that and after that?

$$\begin{tikzcd} \underbrace{\text{"not ab"}} \arrow[d] & \underbrace{ab} \arrow[d] & \underbrace{\text{"not ab"}} \arrow[d] \\ b^*a^* & ab & b^*a^* \end{tikzcd}$$

$$\boxed{b^*\underbrace{a^*a}\underbrace{b\,b^*}a^*}$$

$$a^*a = aa^* = a^+$$

$$\boxed{b^*a^+b^+a^*}$$

4. Построить РВ для языка всех слов чётной длины

$\Sigma = a, b$

ε, aa, bb, ab, ba

$l = 0$ $\quad$ $l = 2$

$l = 4$: $\underbrace{aa}\underbrace{aa}$, $\underbrace{aa}\underbrace{bb}$

$\forall w \in L \Rightarrow w \circ w_2 \in L$, w_2 - word with length $= 2$

$4 = 3 + 1$

$\forall x, y \in L \Rightarrow x \circ y \in L$

$3 = 2 + 1$

$\mathbb{N}$ $0, 1$ $\quad$ $2 = 1 + 1$

inductive generation of language

Set {aa, bb, ab, ba} is a generating set for the language

$$\begin{tikzcd} & waa \\ w \in L \,; \quad w \arrow[ur] \arrow[r] \arrow[dr] \arrow[ddr] & wab \\ & wba \\ & wbb \end{tikzcd}$$

$(ba|bb|aa|ab)^*$

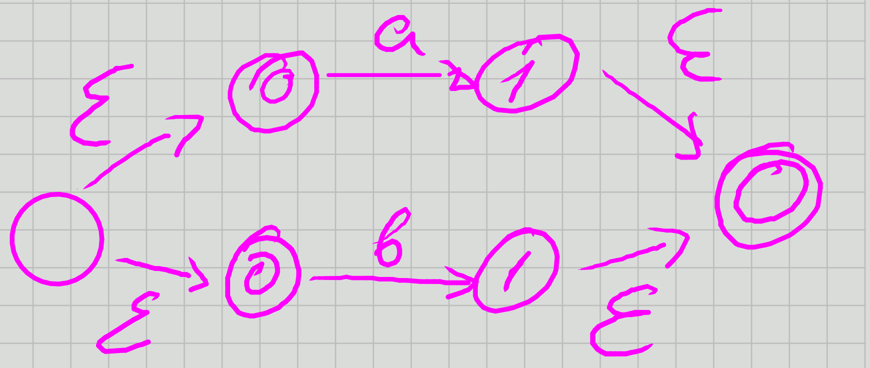
$[(a|b)(a|b)]^* = (\Sigma\Sigma)^* = (\Sigma^2)^*$

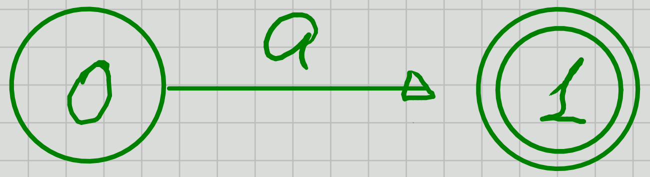
5. Постройте НКА А по РВ (a(alb))*b.

RE → NFA

$$\left[a(a|b)\right]^*b$$

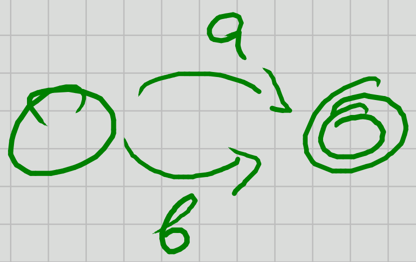
1. A_1 : $\{ a \}$



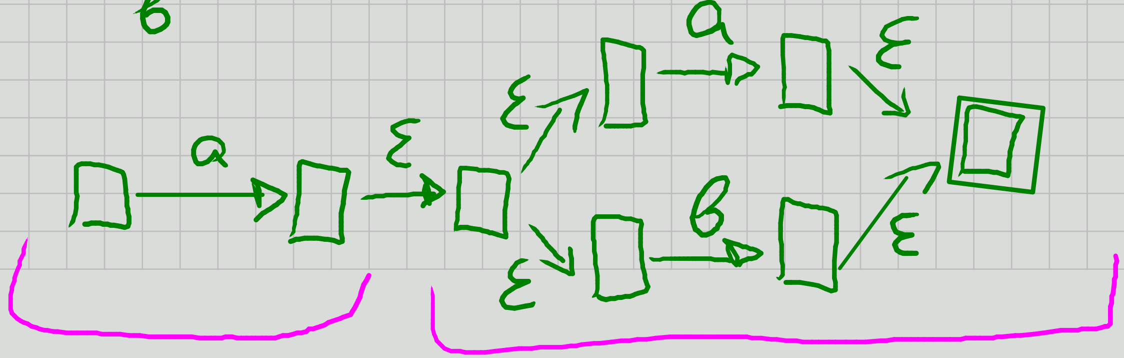


2. A_2 : $\{ b \}$

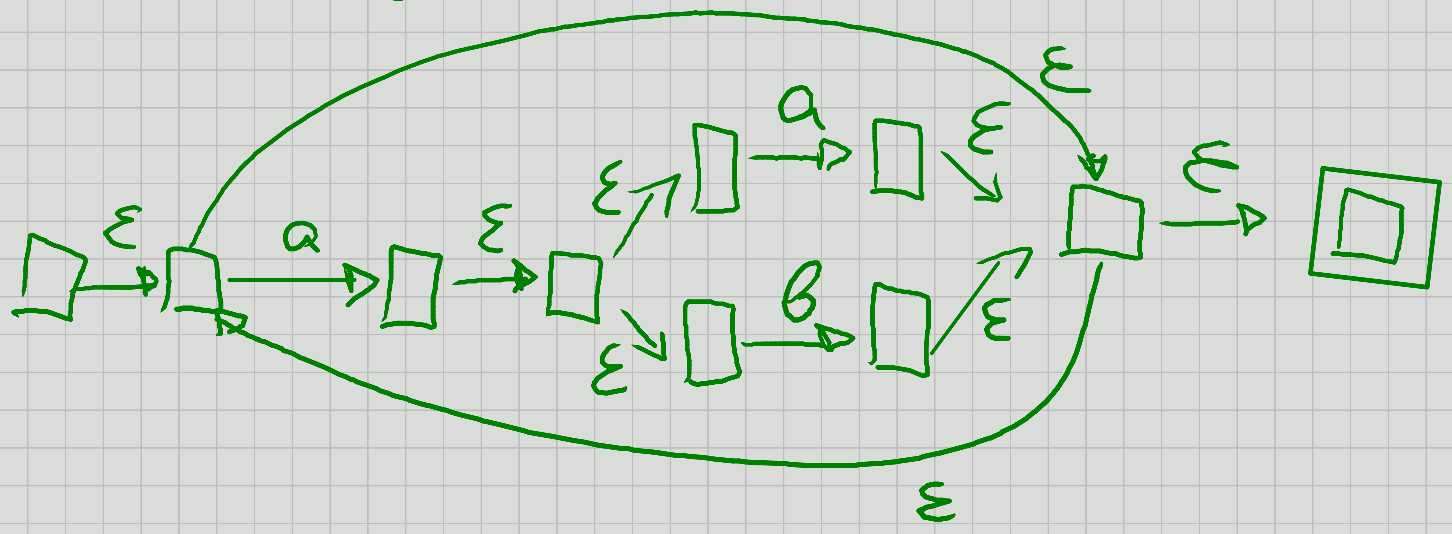
3. $\{ a | b \}$



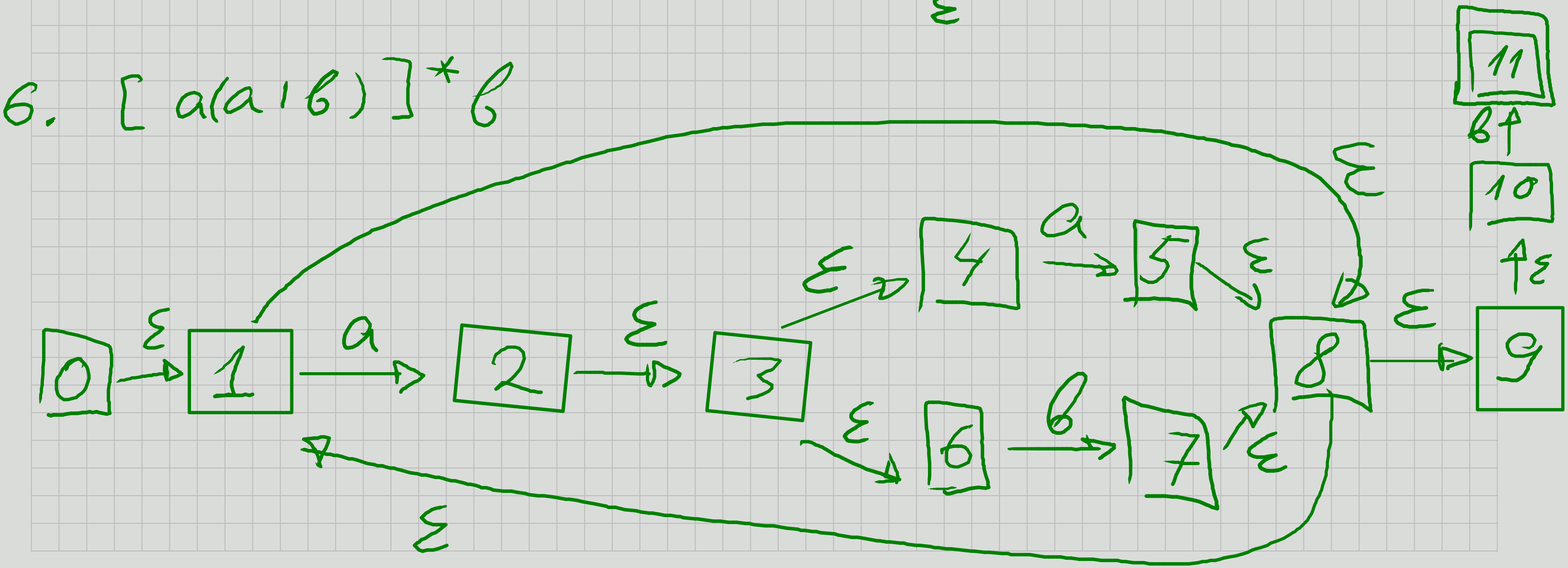
4. $a(a|b)$

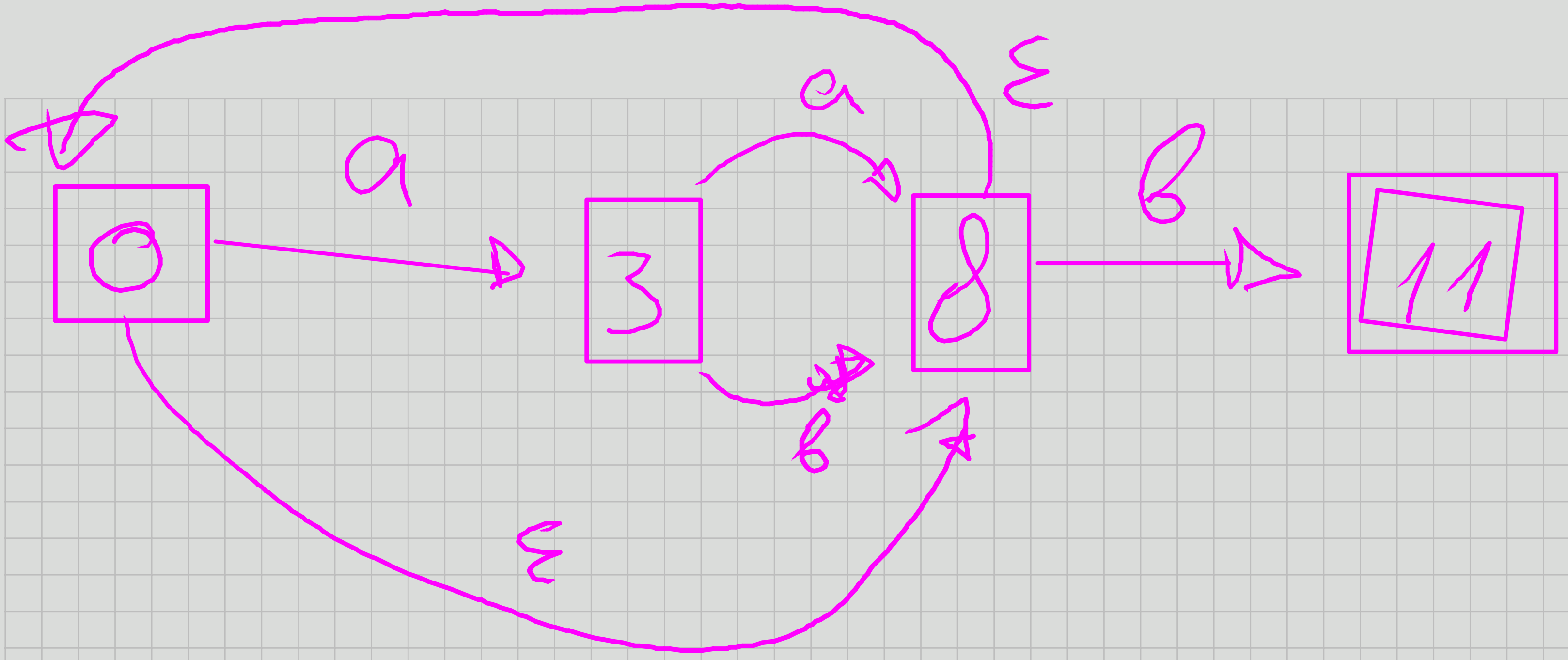


5. $[a(a|b)]^*$



6. $[a(a|b)]^* b$

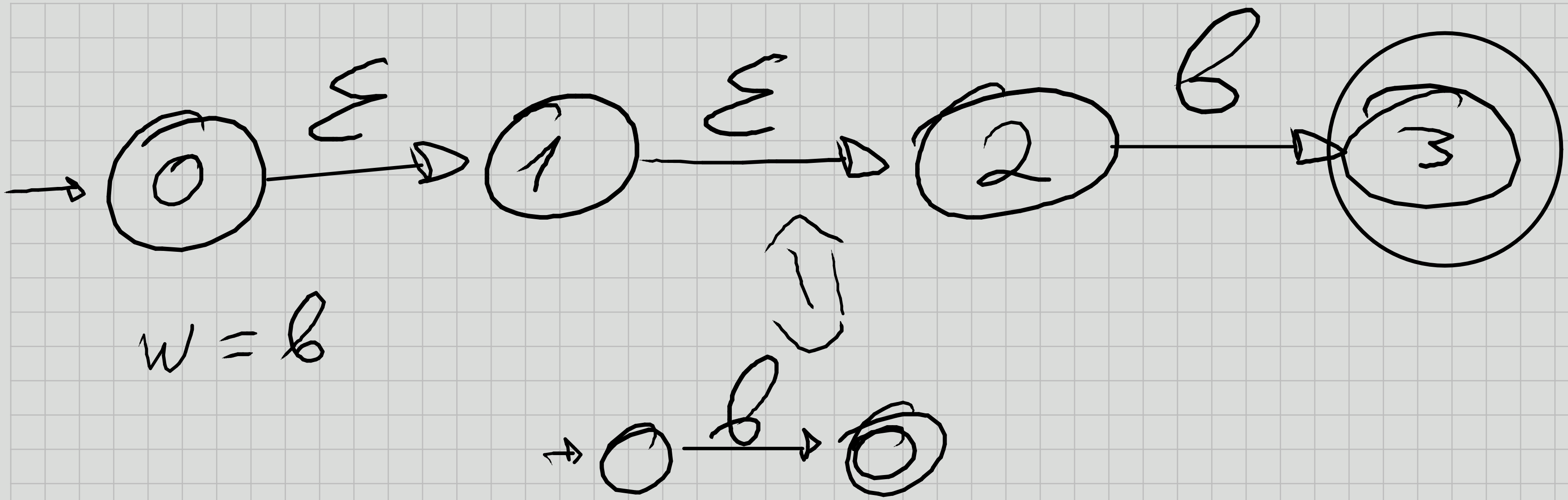



0
a
3
a
b
8
ε
ε
b
11

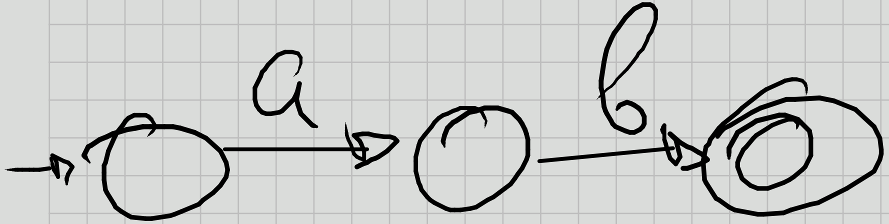
NFA → RE

$$\begin{tikzcd}
RE \arrow[rr] \arrow[dr] & & G \\
& A \arrow[ur] &
\end{tikzcd}$$

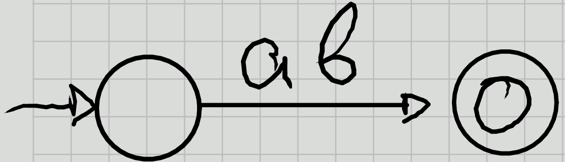
$$\begin{tikzcd}
G \arrow[rr] \arrow[dr] & & RE \\
& A \arrow[ur] &
\end{tikzcd}$$



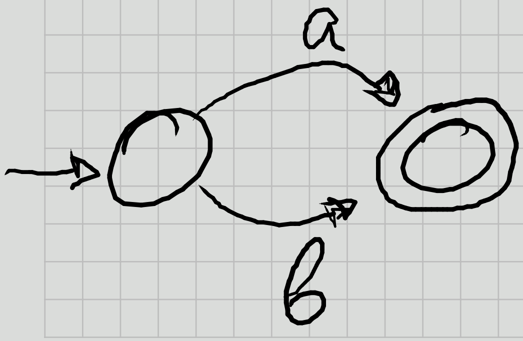
$w = b$



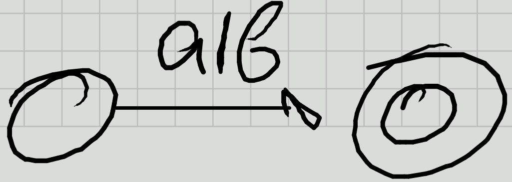
$w = ab$ $L = \{ ab \}$ $r = ab$

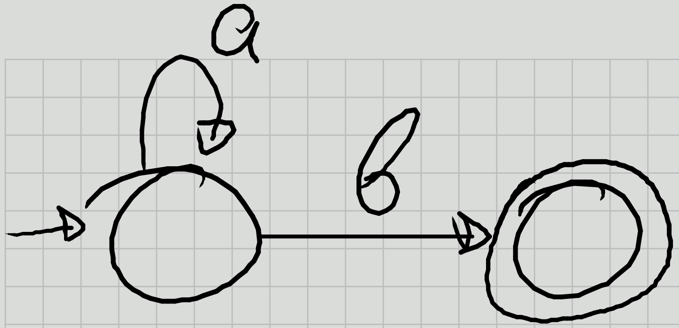


$\delta(p, w)$ $w = r$

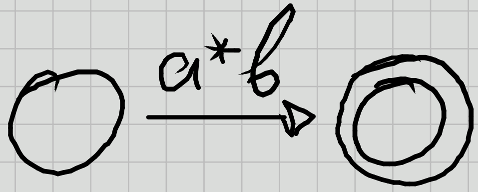


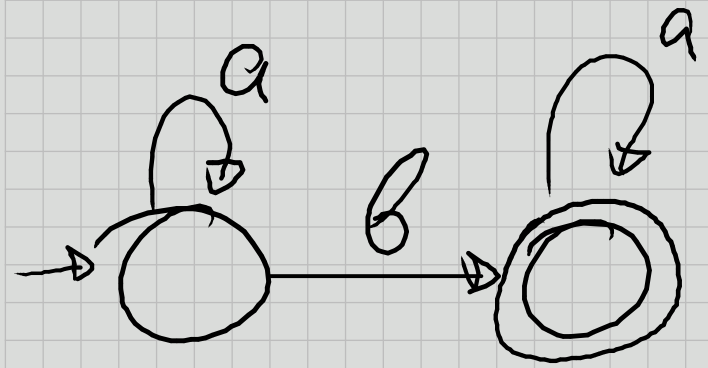
$r = a|b$



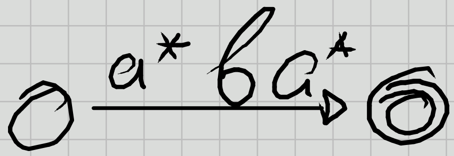


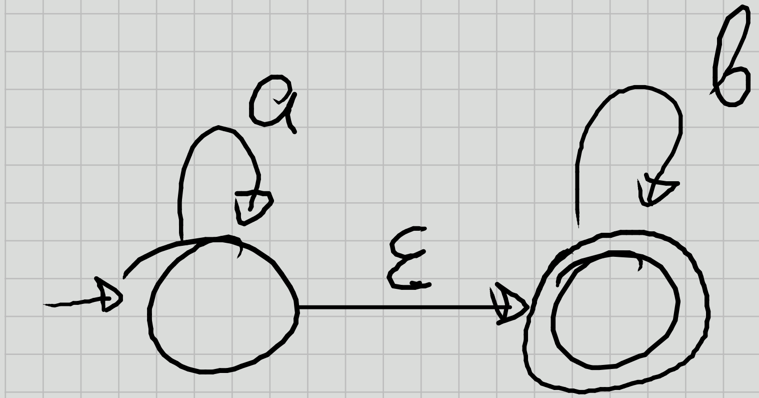
$r = a^* b$



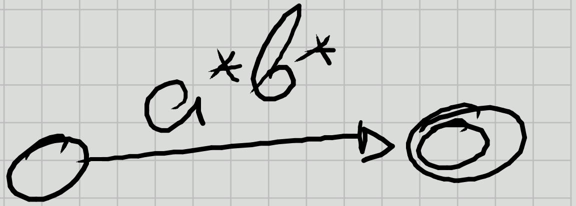


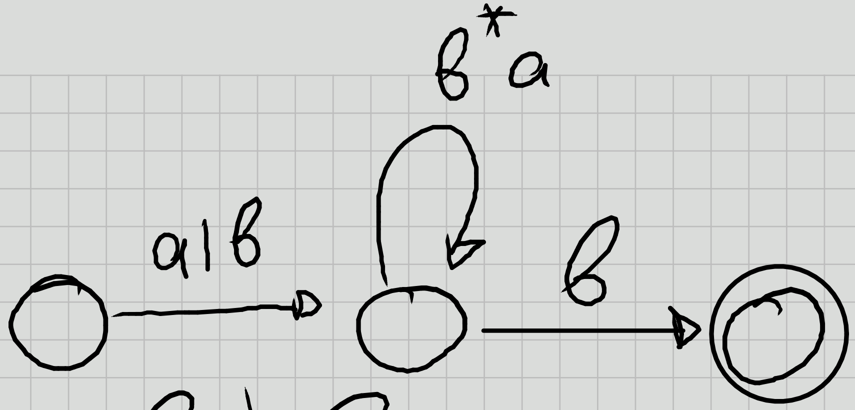
$r = a^* b a^*$





$r = a^* b^*$





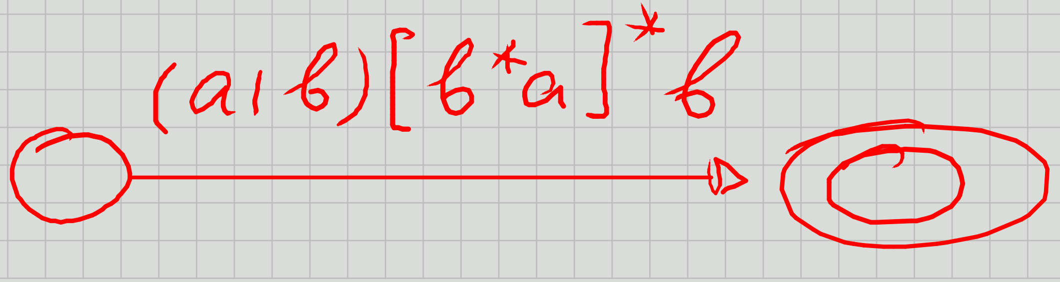
$$r = (a|b)[b^*a]^*b$$

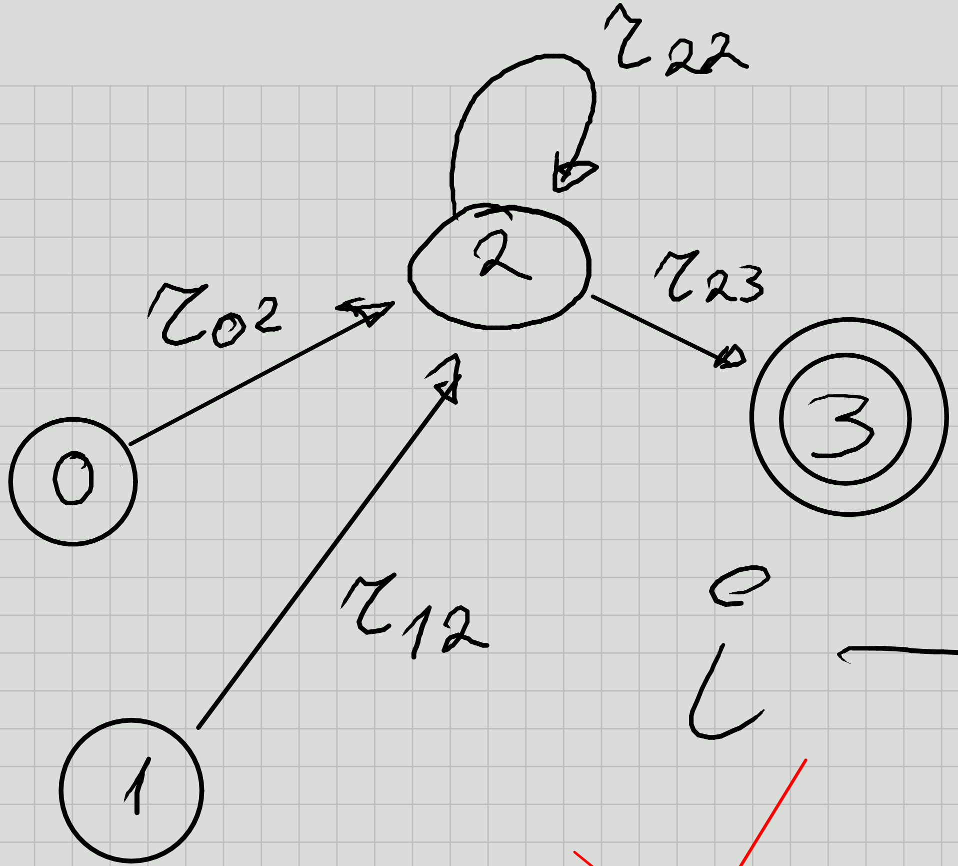
$$(a|b)b$$

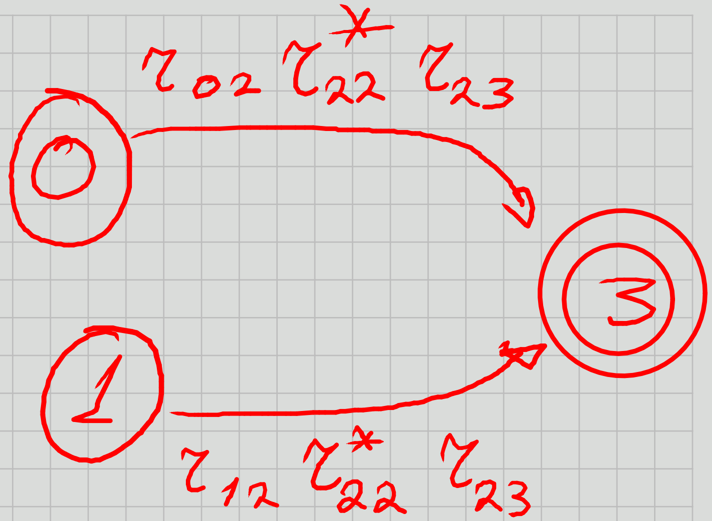
$$(a|b)(b^*a)b$$

$$(a|b)(b^*a)(b^*a)(b^*a)b$$

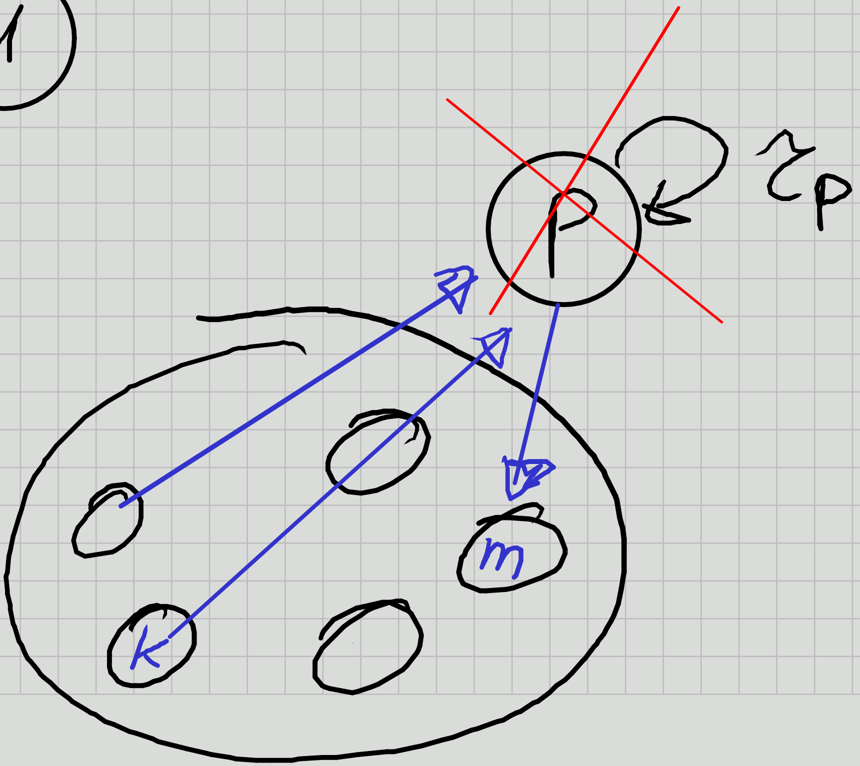
$$\vdots$$





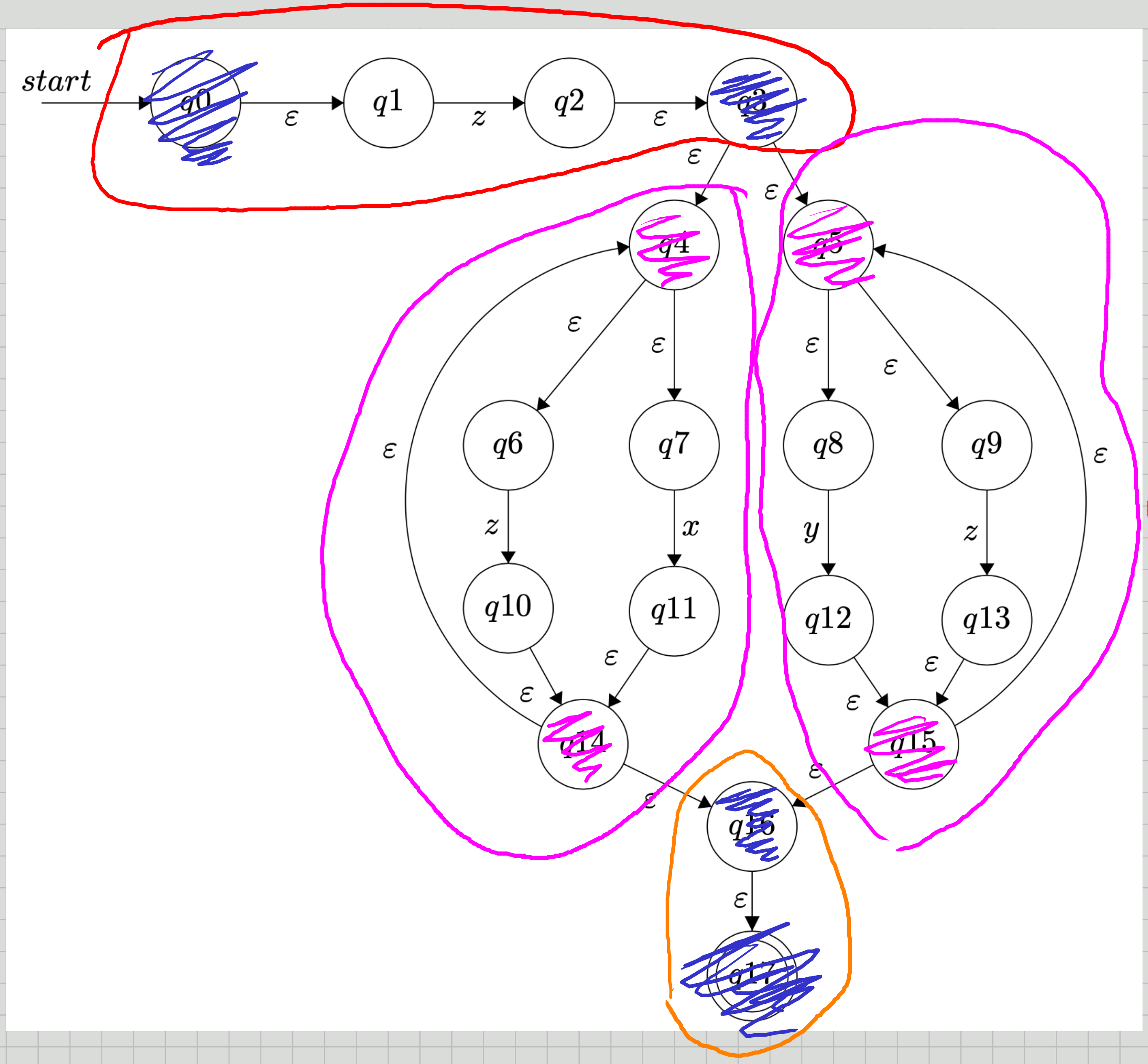


$$i \longrightarrow j$$



∃ path

$$\begin{tikzcd} P_k \arrow[r, "\tau_k"] & P \arrow[r, "\tau_m"] \arrow[d, Leftrightarrow] & P_m \\ P_k \arrow[rr, "\tau_k \tau_p^* \tau_m"] & {} & P_m \end{tikzcd}$$



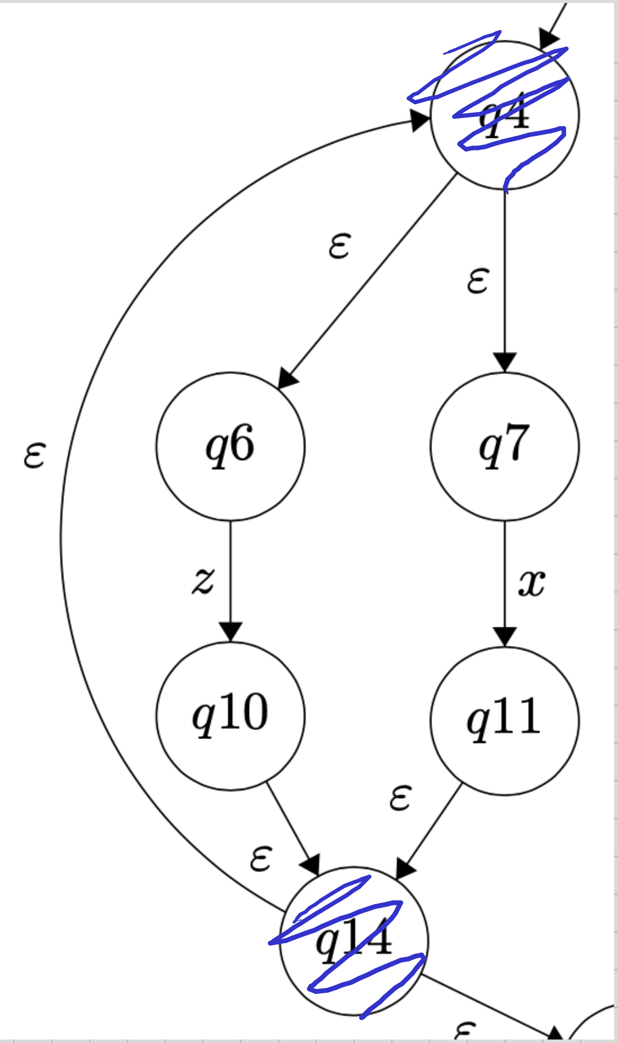
$0 \xrightarrow{\varepsilon} 1 \xrightarrow{z} 2 \xrightarrow{\varepsilon} 3$

$z = \varepsilon z \varepsilon = z$

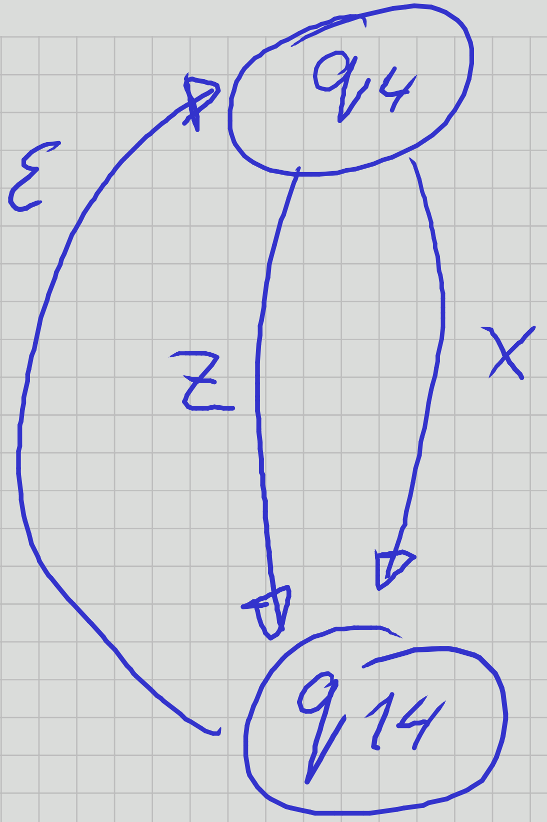
$q_0 \xrightarrow{z} q_3$

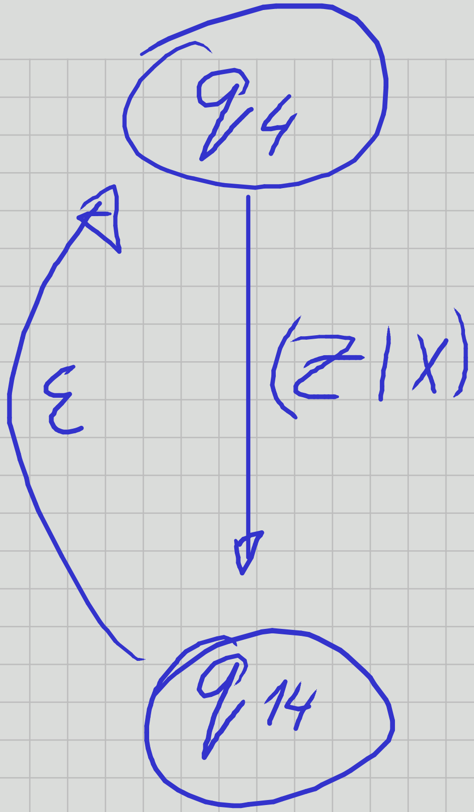
$\xcancel{q_{16}} \xrightarrow{\varepsilon} q_{17}$

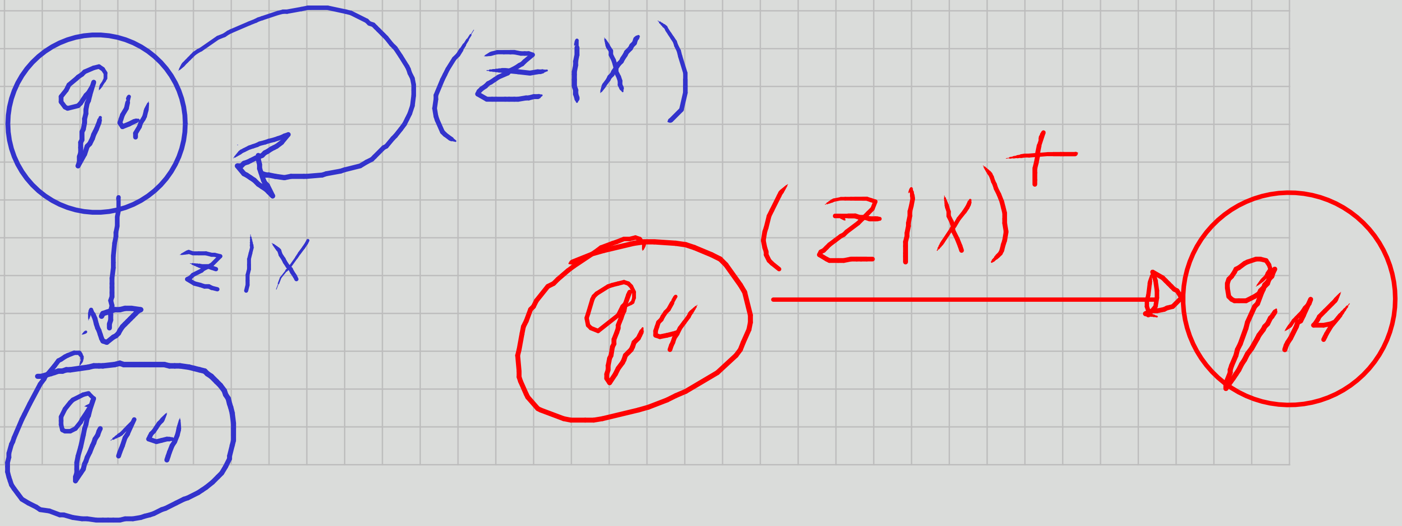
q_{17}

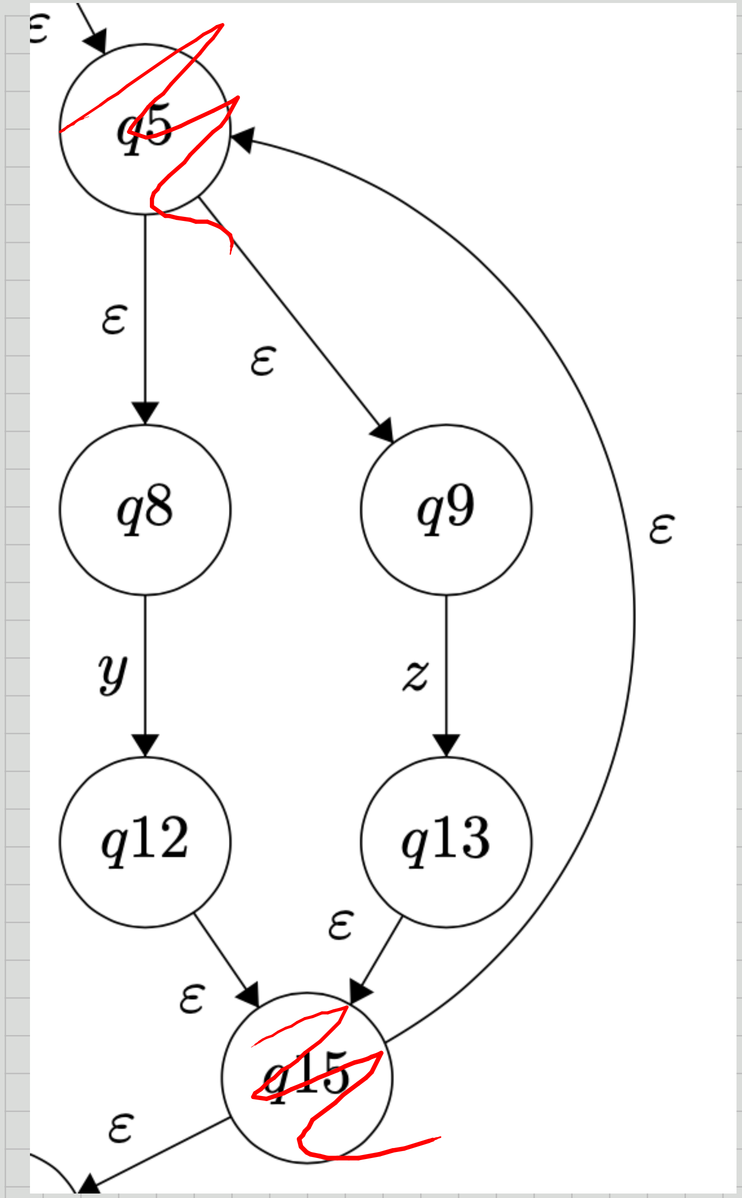

q4
ε
ε
ε
q6
q7
z
x
q10
q11
ε
ε
q14
ε

$r=(z\,|x)^*(z|x)$


q4
ε
z
x
q14

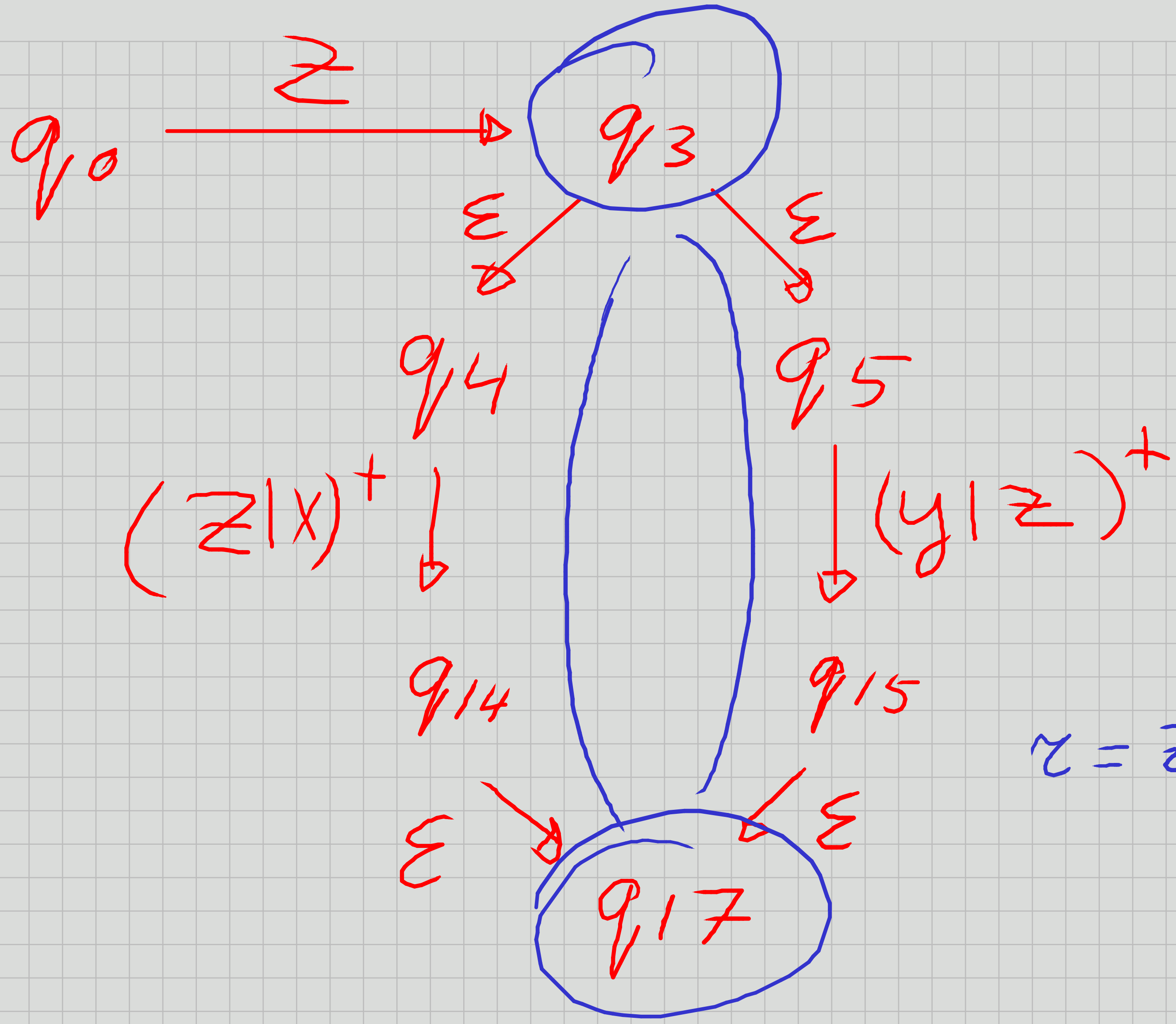

q4
(z|x)
ε
q14


q4
(z|x)
z|x
q14
(z|x)+
q4
q14



$$(y|z)^+$$

$$q5 \xrightarrow{(y|z)^+} q15$$



$$r = z\left[(z|x)^+ | (y|z)^+\right]$$

$$q_0 \xrightarrow{z} q_3 \xrightarrow{(z|x)^+ \,|\, (y|z)^+} q_{17}$$