

# Course 3

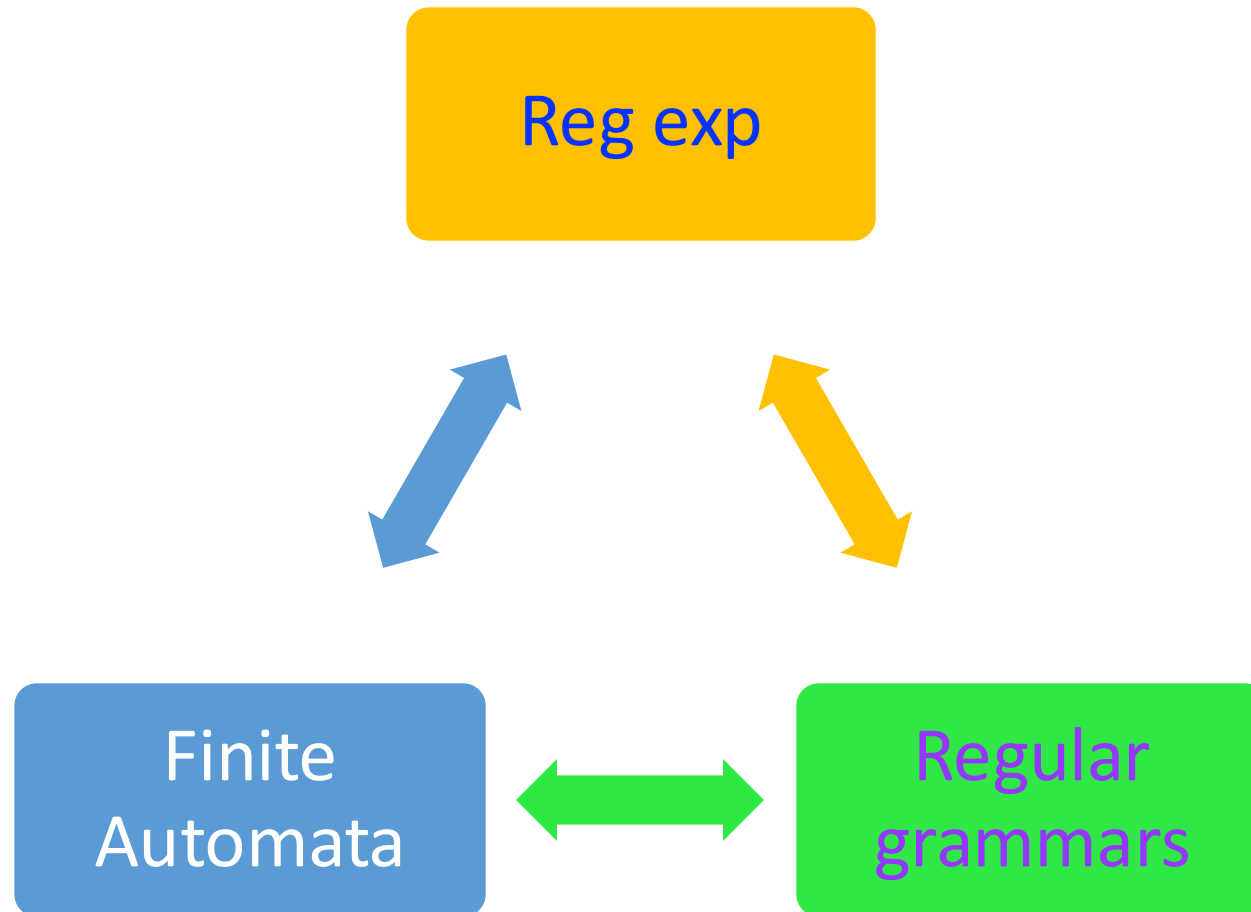
## Formal Languages

- *Basic notions* -

# Regular languages

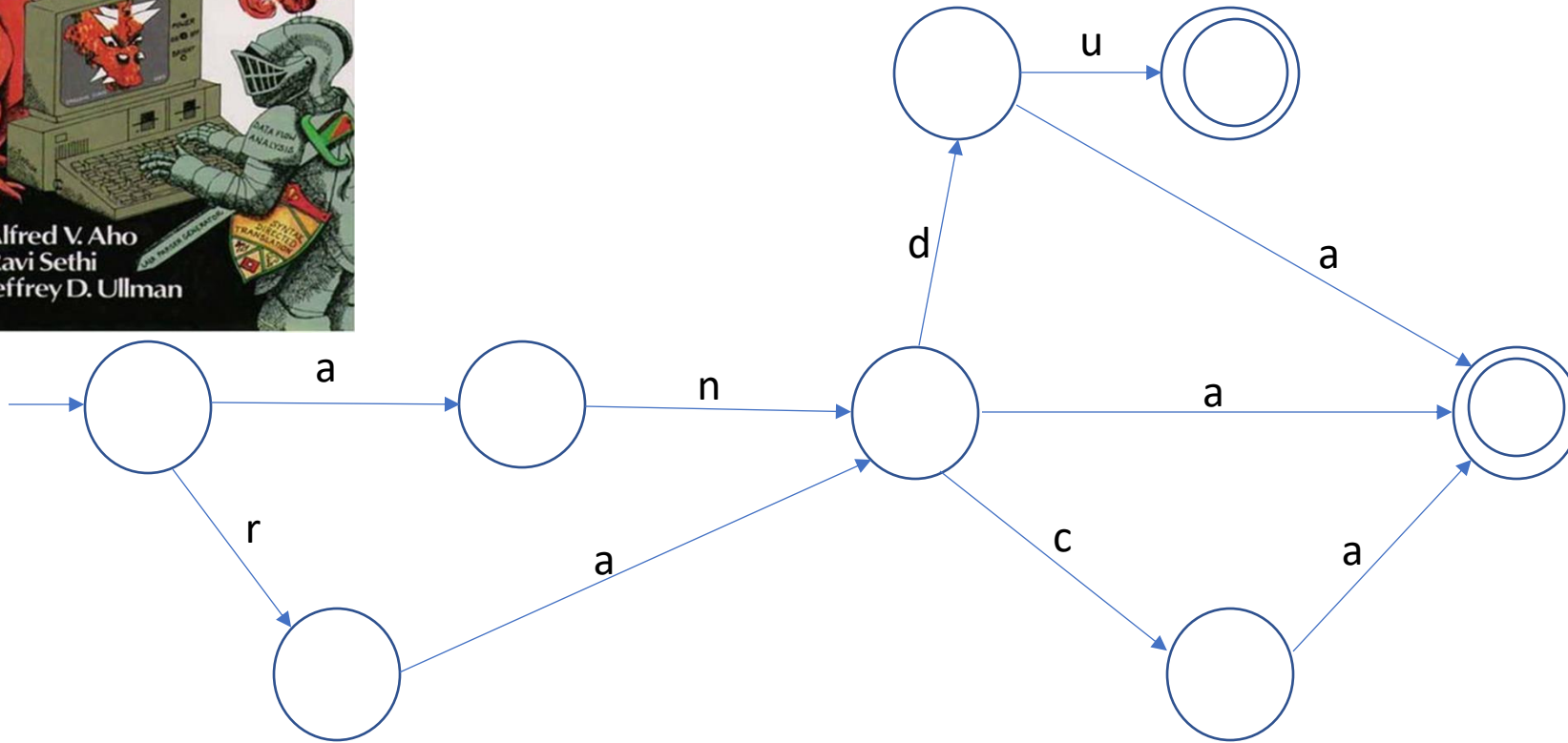
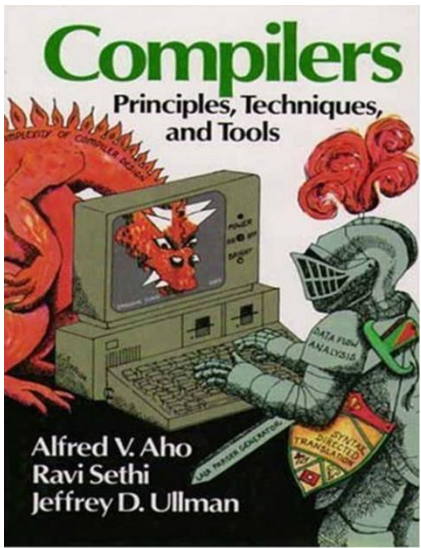
# Why?

1. Search engine – success of Google
2. Unix commands
3. Programming languages – new feature



# Regular grammars

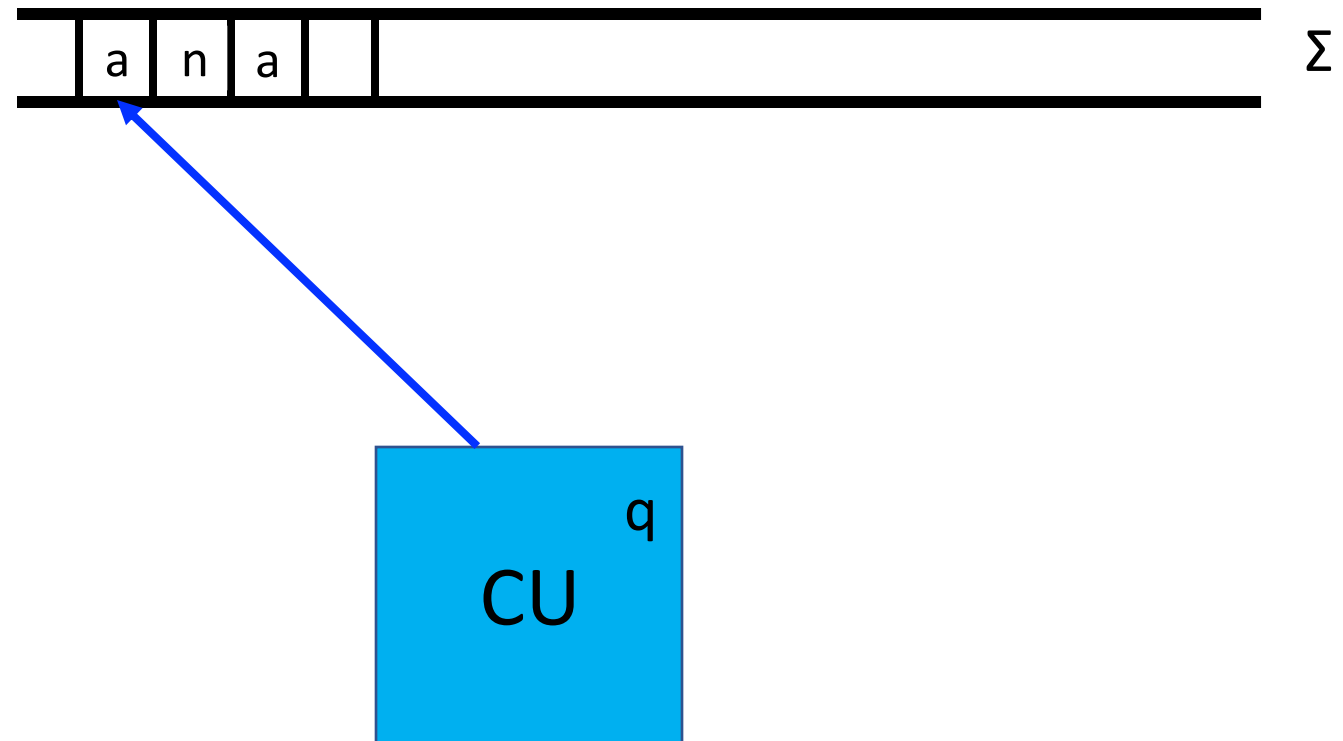
- $G = (N, \Sigma, P, S)$  **right linear grammar** if
$$\forall p \in P: A \rightarrow aB \text{ or } A \rightarrow b, \text{ where } A, B \in N \text{ and } a, b \in \Sigma$$
- $G = (N, \Sigma, P, S)$  **regular grammar** if
  - $G$  is right linear grammar
  - and
  - $A \rightarrow \varepsilon \notin P$ , with the exception that  $S \rightarrow \varepsilon \in P$ , in which case  $S$  does not appear in the rhs (right hand side) of any other production
- $L(G) = \{w \in \Sigma^* \mid S \xRightarrow{*} w\}$  - right linear language



**Problem:** The door to the tower is closed by the **Red Dragon**, using a complicated machinery. Prince Charming has managed to steal the plans and is asking for your help. Can you help him determining all the person names that can unlock the door

# Finite Automata

- Intuitive model



**Definition:** A *finite automaton (FA)* is a 5-tuple

$$M = (Q, \Sigma, \delta, q_0, F)$$

where:

- $Q$  - finite set of states ( $|Q| < \infty$ )
- $\Sigma$  - finite alphabet ( $|\Sigma| < \infty$ )
- $\delta$  – transition function :  $\delta: Q \times \Sigma \rightarrow P(Q)$
- $q_0$  – initial state  $q_0 \in Q$
- $F \subseteq Q$  – set of final states



## Remarks

1.  $Q \cap \Sigma = \emptyset$
2.  $\delta: Q \times \Sigma \rightarrow P(Q)$  ,  $\epsilon \in \Sigma^0$  - relation  $\delta(q, \epsilon) = p$  **NOT** allowed
3. If  $|\delta(q, a)| \leq 1 \Rightarrow$  deterministic finite automaton (DFA)
4. If  $|\delta(q, a)| > 1$  (more than a state obtained as result)  $\Rightarrow$  nondeterministic finite automaton (NFA)

**Property:** For any NFA  $M$  there exists a DFA  $M'$  equivalent to  $M$

## *Configuration $C=(q,x)$*

where:

- $q$  state
- $x$  unread sequence from input:  $x \in \Sigma^*$

Initial configuration :  $(q_0, w)$  ,  $w$  - whole sequence

Final configuration:  $(q_f, \varepsilon)$  ,  $q_f \in F$ ,  $\varepsilon$  –empty sequence  
(corresponds to accept)

# Relations between configurations

- $\vdash$  **move / transition** (simple, one step)  
 $(q, ax) \vdash (p, x)$  ,  $p \in \delta(q, a)$
- $\vdash^k$  **k move** = a sequence of k simple transitions)  $C_0 \vdash C_1 \vdash \dots \vdash C_k$
- $\vdash^+$  **+ move**  
 $C \vdash^+ C' : \exists k > 0$  such that  $C \vdash^k C'$
- $\vdash^*$  **\* move (star move)**  
 $C \vdash^* C' : \exists k \geq 0$  such that  $C \vdash^k C'$

**Definition** : **Language** accepted by FA  $M = (Q, \Sigma, \delta, q_0, F)$  is:

$$L(M) = \{ w \in \Sigma^* \mid (q_0, w) \vdash^* (q_f, \varepsilon), q_f \in F \}$$

### Remarks

1. 2 finite automata  $M_1$  and  $M_2$  are equivalent if and only if they accept the same language

$$L(M_1) = L(M_2)$$

1.  $\varepsilon \in L(M) \Leftrightarrow q_0 \in F$  (initial state is final state)

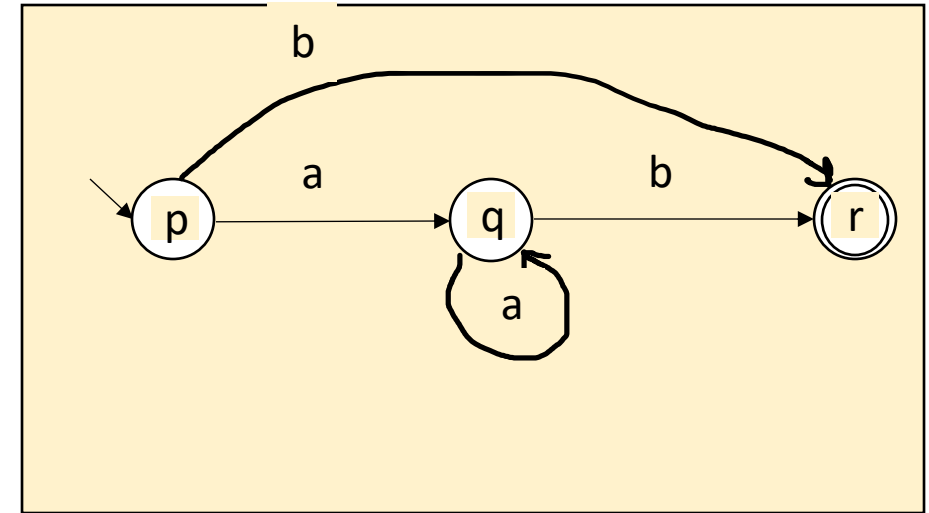
# Representing FA

1. List of all elements
2. Table
3. Graphical representation

$M=(Q,\Sigma,\delta,p,F)$   
 $Q = \{p,q,r\}$   
 $\Sigma = \{a,b\}$   
 $\delta(p,a) = q$   
 $\delta(q,a)=q$   
 $\delta(q,b)=r$   
 $\delta(p,b)=r$   
 $F = \{r\}$

$M=(Q,\Sigma,\delta,p,F)$   
 $F = \{r\}$

	a	b
p	q	r
q	q	r
r	-	-



$(p,aab) \mid -(q,ab) \mid -(q,b) \mid -(r,\epsilon) \Rightarrow aab \text{ accepted}$   
 $(p,aba) \mid -(q,ba) \mid -(r,a) \Rightarrow aba \text{ not accepted}$

# Remember

- Grammar

$$G=(N,\Sigma,P,S)$$

$$L(G)=\{w \in \Sigma^* \mid S \xRightarrow{*} w\}$$

- Finite automaton

$$M = (Q,\Sigma,\delta,q_0,F)$$

$$L(M)=\{ w \in \Sigma^* \mid (q_0,w) \vdash (q_f,\varepsilon) , q_f \in F \}$$

**Theorem 1:** For any regular grammar  $G=(N, \Sigma, P, S)$  there exists a FA  $M=(Q, \Sigma, \delta, q_0, F)$  such that  $L(G) = L(M)$

Proof: construct  $M$  based on  $G$

$Q = N \cup \{K\}, K \notin N$

$q_0 = S$

$F = \{K\} \cup \{S \mid S \rightarrow \epsilon \in P\}$

$\delta$ : if  $A \rightarrow aB \in P$  then  $\delta(A, a) = B$

if  $A \rightarrow \epsilon \in P$  then  $\delta(A, \epsilon) = K$

---

**Theorem 1:** For any regular grammar  $G=(N, \Sigma, P, S)$  there exists a FA  $M=(Q, \Sigma, \delta, q_0, F)$  such that  $L(G) = L(M)$

Proof: **construct M based on G**

$Q = N \cup \{K\}, K \notin N$

$q_0 = S$

$F = \{K\} \cup \{S \mid \text{if } S \rightarrow \epsilon \in P\}$

$\delta$ : if  $A \rightarrow aB \in P$  then  $\delta(A, a) = B$

if  $A \rightarrow a \in P$  then  $\delta(A, a) = K$

**Prove that  $L(G) = L(M)$  ( $w \in L(G) \Leftrightarrow w \in L(M)$ ):**

$S \xRightarrow{*} w \Leftrightarrow (S, w) \vdash^* (q_f, \epsilon)$

$w = \epsilon$ :  $S \xRightarrow{*} \epsilon \Leftrightarrow (S, \epsilon) \vdash^* (S, \epsilon)$  – true

$w = a_1 a_2 \dots a_n$ :  $S \xRightarrow{*} w \Leftrightarrow (S, w) \vdash^* (K, \epsilon)$

$S \Rightarrow a_1 A_1 \Rightarrow a_1 a_2 A_2 \Rightarrow \dots \Rightarrow a_1 a_2 \dots a_{n-1} A_{n-1} \Rightarrow a_1 a_2 \dots a_{n-1} a_n$

$S \Rightarrow a_1 A_1$  exists if  $S \rightarrow a_1 A_1$  and then  $\delta(S, a_1) = A_1$

$A_1 \rightarrow a_2 A_2 : \delta(A_1, a_2) = A_2 \dots$

$A_{n-1} \rightarrow a_n : \delta(A_{n-1}, a_n) = K$

$(S, a_1 a_2 \dots a_n) \vdash (A_1, a_2 \dots a_n) \vdash (A_2, a_3 \dots a_n) \vdash \dots \vdash (A_{n-1}, a_n) \vdash (K, \epsilon), K \in F$



**Theorem 2:** For any FA  $M=(Q, \Sigma, \delta, q_0, F)$  there exists a right linear grammar  $G=(N, \Sigma, P, S)$  such that  $L(G) = L(M)$

Proof: **construct G based on M**

$N = Q$

$S = q_0$

$P$ : if  $\delta(q, a) = p$  then  $q \rightarrow ap \in P$

if  $p \in F$  then  $q \rightarrow a \in P$

if  $q_0 \in F$  then  $S \rightarrow \epsilon$

---

**Theorem 2:** For any FA  $M=(Q, \Sigma, \delta, q_0, F)$  there exists a right linear grammar  $G=(N, \Sigma, P, S)$  such that  $L(G) = L(M)$

Proof: **construct G based on M**

$N = Q$

$S = q_0$

$P$ : if  $\delta(q,a) = p$  then  $q \rightarrow ap \in P$

if  $p \in F$  then  $q \rightarrow a \in P$

if  $q_0 \in F$  then  $S \rightarrow \epsilon$

**Prove that  $L(M) = L(G)$  ( $w \in L(M) \Leftrightarrow w \in L(G)$ ):**

$P(i): q \xRightarrow{i+1} x \Leftrightarrow (q,x) \vdash^i (q_f, \epsilon), q_f \in F$  -prove by induction

Apply  $P : q_0 \xRightarrow{i+1} w \Leftrightarrow (q_0, w) \vdash^i (q_f, \epsilon), q_f \in F$

If  $i=0: q \Rightarrow x \Leftrightarrow (q,x) \vdash^0 (q_f, \epsilon) (x = \epsilon, q = q_f) q \Rightarrow \epsilon \Leftrightarrow q_0 \rightarrow \epsilon, q_0 \in F$

Assume  $\forall k \leq i$   $P$  is true

$q \xRightarrow{i+1} x \Leftrightarrow (q,x) \vdash^i (q_f, \epsilon)$

For  $q \in N$  apply " $\Rightarrow$ " :  $q \Rightarrow ap \xRightarrow{i} ax$

If  $q \Rightarrow ap$  then  $\delta(q,a) = p$  ; if  $p \xRightarrow{i} ax$  then  $(p,x) \vdash^{i-1} (q_f, \epsilon), q_f \in F$

THEN  $(q, ax) \vdash^i (q_f, \epsilon), q_f \in F$

# Regular sets

**Definition:** Let  $\Sigma$  be a finite alphabet. We define regular sets over  $\Sigma$  recursively in the following way:

1.  $\emptyset$  is a regular set over  $\Sigma$  (empty set)
2.  $\{\epsilon\}$  is a regular set over  $\Sigma$
3.  $\{a\}$  is a regular set over  $\Sigma$ ,  $\forall a \in \Sigma$
4. If  $P$ ,  $Q$  are regular sets over  $\Sigma$ , then  $P \cup Q$ ,  $PQ$ ,  $P^*$  are regular sets over  $\Sigma$
5. Nothing else is a regular set over  $\Sigma$

# Regular expressions

**Definition:** Let  $\Sigma$  be a finite alphabet. We define regular expressions over  $\Sigma$  recursively in the following way:

1.  $\emptyset$  is a regular expression denoting the regular set  $\emptyset$  (empty set)
2.  $\epsilon$  is a regular expression denoting the regular set  $\{\epsilon\}$
3.  $a$  is a regular expression denoting the regular set  $\{a\}$ ,  $\forall a \in \Sigma$
4. If  $p, q$  are regular expression denoting the regular sets  $P, Q$  then:
  - $p+q$  is a regular expression denoting the regular set  $P \cup Q$ ,
  - $pq$  is a regular expression denoting the regular set  $PQ$ ,
  - $p^*$  is a regular expression denoting the regular set  $P^*$
5. Nothing else is a regular expression

# Remarks:

## Examples

1.  $p^+ = pp^*$
2. Use paranthesis to avoid ambiguity
3. Priority of operations: \*, concat, + (from high to low)
4. For each regular set we can find at least one regular exp to denote it (there is an infinity of reg exp denoting them)
5. For each regular exp, we can construct the corresponding regular set
6. 2 regular expressions are **equivalent** iff they denote the same regular set

# Algebraic properties of regular exp

Let  $\alpha$ ,  $\beta$ ,  $\gamma$  be regular expressions.

1.  $\alpha + \beta = \beta + \alpha$

2.  $\Phi^* = \varepsilon$

3.  $\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma$

4.  $\alpha(\beta\gamma) = (\alpha\beta)\gamma$

5.  $\alpha(\beta + \gamma) = \alpha\beta + \alpha\gamma$

6.  $(\alpha + \beta)\gamma = \alpha\gamma + \beta\gamma$

7.  $\alpha \varepsilon = \varepsilon \alpha = \alpha$

8.  $\Phi\alpha = \alpha\Phi = \Phi$

9.  $\alpha^* = \alpha + \alpha^*$

10.  $(\alpha^*)^* = \alpha^*$

11.  $\alpha + \alpha = \alpha$

12.  $\alpha + \Phi = \alpha$

# Reg exp equations

- Normal form:  $\mathbf{X = aX + b}$

where a,b – reg exp

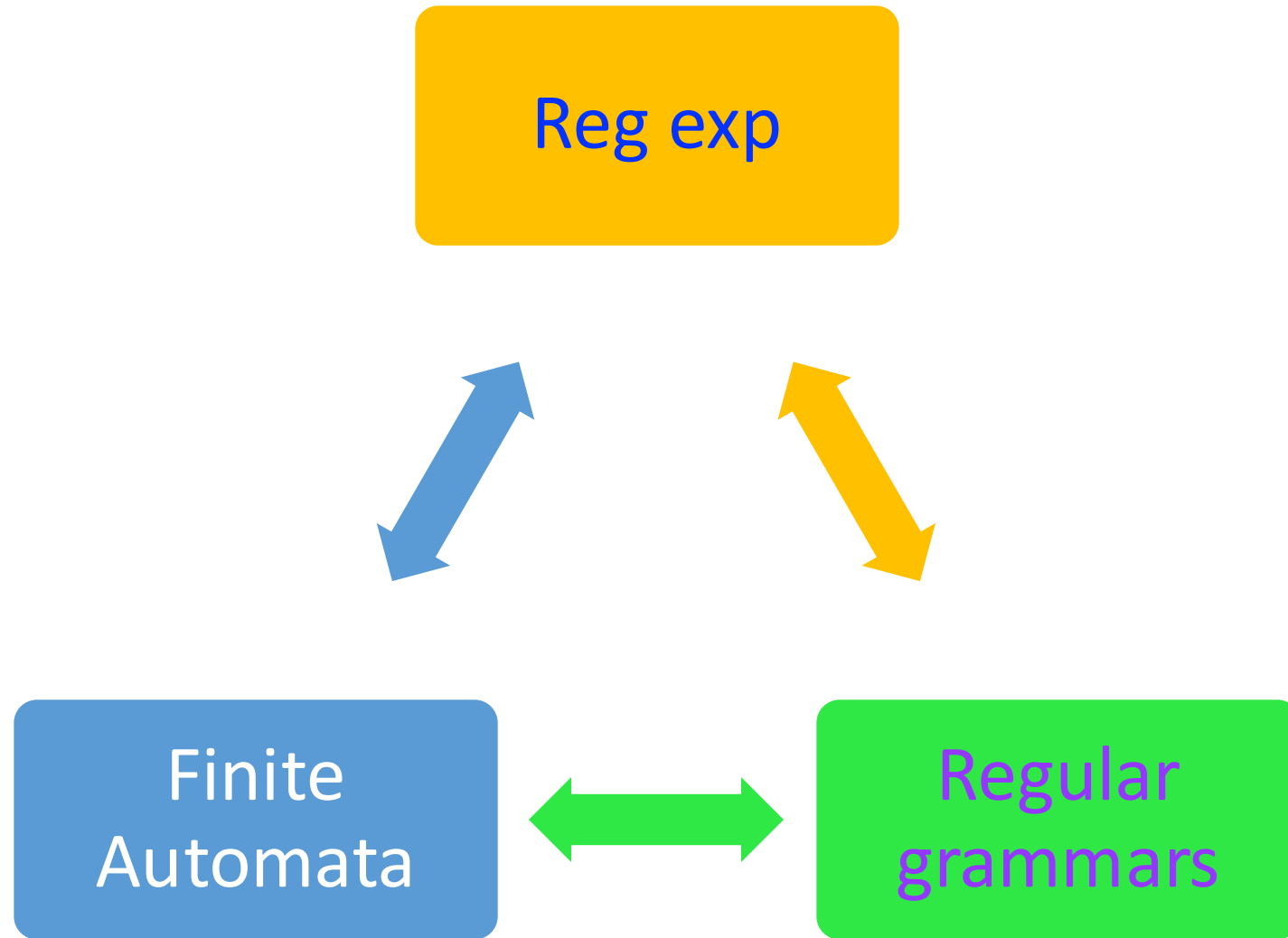
$$a a^* b + b = (a a^* + \epsilon) b = a^* b$$

- Solution:  $\mathbf{X = a^* b}$

- System of reg exp equations:

$$\begin{cases} X = a_1 X + a_2 Y + a_3 \\ Y = b_1 X + b_2 Y + b_3 \end{cases}$$

- Solution: Gauss method (replace  $X_i$  and solve  $X_n$ )





**Prop:** *Regular sets are right linear languages*

**Lemma 1:**  $\Phi, \{\epsilon\}, \{a\}, \forall a \in \Sigma$  are right linear languages

Proof: constructive

- i.  $G = (\{S\}, \Sigma, \Phi, S)$  – regular grammar such that  $L(G) = \Phi$
- ii.  $G = (\{S\}, \Sigma, \{S \rightarrow \epsilon\}, S)$  – regular grammar such that  $L(G) = \{\epsilon\}$
- iii.  $G = (\{S\}, \Sigma, \{S \rightarrow a\}, S)$  – regular grammar such that  $L(G) = \{a\}$

**Lemma 2:** If  $L_1$  and  $L_2$  are right linear languages then:  
 $L_1 \cup L_2$ ,  $L_1L_2$  and  $L_1^*$  are right linear languages.

Proof: constructive

$L_1, L_2$  right linear languages  $\Rightarrow \exists G_1, G_2$  such that

$G_1 = (N_1, \Sigma_1, P_1, S_1)$  and  $L_1 = L(G_1)$

$G_2 = (N_2, \Sigma_2, P_2, S_2)$  and  $L_2 = L(G_2)$                       assume  $N_1 \cap N_2 = \emptyset$

i.  $G_3 = (N_3, \Sigma, P_3, S_3)$

$$N_3 = N_1 \cup N_2 \cup \{S_3\}; \Sigma_3 = \Sigma_1 \cup \Sigma_2$$

$$P_3 = P_1 \cup P_2 \cup \{S_3 \rightarrow S_1 \mid S_2\}$$

$$\{S_3 \rightarrow \alpha_1 \mid S_1 \rightarrow \alpha_1 \in P_1\} \cup \{S_3 \rightarrow \alpha_2 \mid S_2 \rightarrow \alpha_2 \in P_2\}$$

$G_3$  – right linear language  
and

$$L(G_3) = L(G_1) \cup L(G_2)$$

**PROOF!!! Homework**

$$\text{ii. } G_4 = (N_4, \Sigma, P_4, S_4)$$

$$N_4 = N_1 \cup N_2; S_4 = S_1; \Sigma_4 = \Sigma_1 \cup \Sigma_2$$

$$P_4 = \{A \rightarrow aB \mid \text{if } A \rightarrow aB \in P_1\} \cup \\ \{A \rightarrow aS_2 \mid \text{if } A \rightarrow a \in P_1\} \cup \\ P_2 \cup \\ \{S_1 \rightarrow \alpha_2 \mid \text{if } S_1 \rightarrow \epsilon \in P_1 \text{ and } S_2 \rightarrow \alpha_2 \in P_2\}$$

$G_4$  – right linear language  
and

$$L(G_4) = L(G_1) L(G_2)$$

**PROOF!!! Homework**

$$\text{iii. } G_5 = (N_5, \Sigma_1, P_5, S_5)$$

//IDEA: concatenate  $L_1$  with itself

$$N_5 = N_1 \cup \{S_5\};$$

$$P_5 = P_1 \cup \{S_5 \rightarrow \epsilon\} \cup \\ \{S_5 \rightarrow \alpha_1 \mid S_1 \rightarrow \alpha_1 \in P_1\} \cup \\ \{A \rightarrow aS_1 \mid \text{if } A \rightarrow a \in P_1\}$$

$G_5$  – right linear language  
and

$$L(G_5) = L(G_1)^*$$

**PROOF!!! Homework**

**Theorem:** *A language is a regular set if and only if it is a right linear language*

Proof:

=> Apply lemma 1 and lemma 2

<= construct a system of regular exp equations where:

- Indeterminants – nonterminals
- Coefficients – terminals
- Equation for A: all the possible rewritings of A

Example:  $G = (\{S, A, B\}, \{0, 1\}, P, S)$

P:  $S \rightarrow 0A \mid 1B \mid \epsilon$

$A \rightarrow 0B \mid 1A$

$B \rightarrow 0S \mid 1$

$$\begin{cases} S = 0A + 1B + \epsilon \\ A = 0B + 1A \\ B = 0S + 1 \end{cases}$$

**Regular exp = solution  
corresponding to S**