# BI-AAG cvičení 2 - Seznam pojmů

Bc. Eliška Šestáková

2.10.2014

**Regulární jazyky** – jakési „nejjednodušší“ formální jazyky. Jedná se o nejmenší množinu jazyků, která obsahuje všechny konečné jazyky a jazyky vzniklé pomocí operaci:

- sjednocení,
- součinu,
- iterace.

Platí tedy, že každý konečný jazyk je regulární (ne naopak).

Formální jazyk je regulární, právě když:

- je akceptovaný nějakým deterministickým konečným automatem,
- je akceptovaný nějakým nedeterministickým konečným automatem,
- může být popsán regulárním výrazem nebo
- může být vygenerován regulární gramatikou

Tyto jednotlivé formy (RG, RV, KA) lze mezi sebou vzájemně převádět.

*Příklady regulárních jazyků:*

- $L = \{w : w \in \{0,1\}^*, w$ je binární zápis sudého čísla$\}$ (tj. množina všech řetězců (vět) $w$ nad abecedou $\{0,1\}$ taková, že $w$ je binární zápis sudého čísla.
- $L = \{w\ :\ w \in \{0,1\}^*,\ w$ obsahuje podřetězec 011 $\}$
- $L = \{w\ :\ w \in \{a,b\}^*,$ třetí symbol od konce je symbol b $\}$

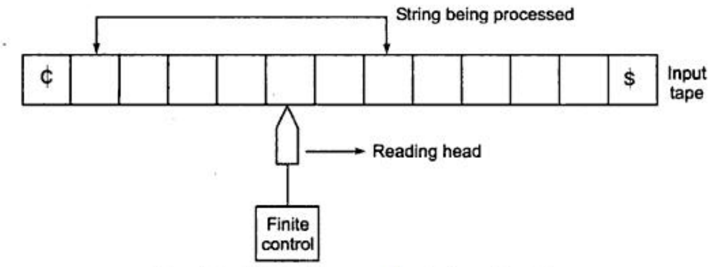
*Příklad jazyka, který není regulární:*

- $L = \{a^n b^n\ :\ n \geq 0\}$ – intuitivně nemůže být rozpoznán konečným automatem, neboť ten má konečnou paměť (konečný počet stavů) a nemůže si tedy zapamatovat libovolný počet symbolů $a$ (aby poté četl stejný počet symbolů $b$).

**Regulární gramatika** – (uspořádaná) čtveřice: $G = (N, T, P, S)$, kde pravidla P jsou ve tvaru ve tvaru:

$$N \rightarrow T \,|\, TN$$

Případně $S \rightarrow \epsilon$, pokud se $S$ neobjevuje na pravé straně žádného pravidla.




*Znázornění konečného automatu* – Automat čte znaky ze vstupní pásky (read-only) pomocí hlavy, která se může pohybovat pouze doprava. Tělo automatu obsahuje vlastní řízení (konečná množina stavů, aktuální stav, přechodová funkce).

**Deterministický konečný automat** (DKA) – (uspořádaná) pětice

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

- Q je konečná množina (vnitřních) stavů,
- T je konečná vstupní abeceda,
- $\delta$ je zobrazení z $Q \times T \to Q$, které nazýváme přechodovou funkcí,
- $q_0 \in Q$ je počáteční stav,
- $F \subseteq Q$ je množina koncových stavů.

**Nedeterministický konečný automat** (NKA) – (uspořádaná) pětice

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

- Q je konečná množina (vnitřních) stavů,
- T je konečná vstupní abeceda,
- $\delta$ je zobrazení z $Q \times T \to 2^Q$ ($2^Q$ je množina podmnožin množiny Q), které nazýváme přechodovou funkcí,
- $q_0 \in Q$ je počáteční stav,
- $F \subseteq Q$ je množina koncových stavů.

**Znázornění KA** - liší se způsobem zápisu přechodové funkce $\delta$. Mějme například následující regulární jazyk:

$$L = \{w : w \in \{0,1\}^*, w \text{ končí na } 10\}$$

- **„formální“ zápis**

$$M = (\{S, A, B\}, \{0,1\}, \delta, S, \{B\})$$

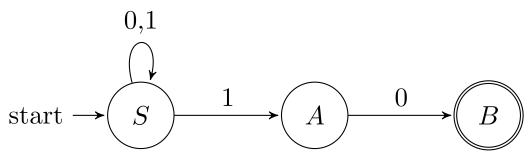
kde $\delta$ :
$\delta(S,0) = \{S\}$
$\delta(S,1) = \{S, A\}$
$\delta(A,0) = \{B\}$

*pozn. u DKA by zápis vypadal takto $\delta(A,0) = B$ (bez množinových závorek, neboť přecházíme vždy právě do jednoho stavu.)*

- **diagram**

$$M = (\{S, A, B\}, \{0, 1\}, \delta, S, \{B\})$$

kde $\delta$ :




Dvojité kolečko označuje přijímající stavy (v našem případě pouze jeden, $B$), počáteční stav je označen šipkou, někdy s připsaným textem, např. START (zde $S$).

- **tabulka**

$$M = (\{S, A, B\}, \{0, 1\}, \delta, S, \{B\})$$

kde $\delta$ :

| $\delta$ | 0 | 1 |
|---|---|---|
| $\rightarrow$ S | S | S,A |
| A | B | - |
| $\leftrightarrow$ B | - | - |

Do nejlevějšího sloupce tabulky zapisujeme jednotlivé stavy KA, do záhlaví tabulky poté jednotlivé symboly z abecedy (případně $\epsilon$). V tabulce poté jen zapisujeme do kterého stavu (případně stavů u NKA) se dostaneme po čtení příslušného symbolu ze vstupu.

**Konfigurace KA** – Nechť máme konečný automat

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

konfigurací konečného automatu nazveme dvojici:

$$(q, w), \text{ kde } q \in Q,\ w \in T^*$$

Dále máme dvě speciální konfigurace:

- $(q_0, w)$ – počáteční konfigurace
- $(q, \epsilon)$, $q \in F$ – koncová konfigurace

**Přechod DKA** – Přechodem v deterministickém konečném automatu nazveme relaci

$$\vdash_M \in (Q \times T^*) \times (Q \times T^*)$$

Existuje-li v DKA přechod ze stavu $q$ na symbol $a$ do $p$, tedy:

$$\delta(q, a) = p$$

poté existuje následující přechod (pro všechna $w \in T^*$)

$$(q, aw) \vdash_M (p, w)$$

**Přechod NKA** – Přechodem v nedeterministickém konečném automatu nazveme relaci

$$\vdash_M \in (Q \times T^*) \times (Q \times T^*)$$

Existuje-li v DKA přechod ze stavu $q$ na symbol $a$ do $p$, tedy: (povšimněte si rozdílného zápisu oproti předchozí definici):

$$p \in \delta(q, a)$$

poté existuje následující přechod (pro všechna $w \in T^*$)

$$(q, aw) \vdash_M (p, w)$$

**Řetězec přijatý KA** – Říkáme, že řetězec $w$ je přijatý deterministickým konečným automatem, jestliže

$$(q_0, w) \vdash^* (q, \epsilon),\ q \in F$$

(neboli existuje posloupnost přechodů taková, že začnu v počátečním stavu (počáteční konfiguraci) a po přečtení celého řetězce skončím v jednom z koncovým stavů (koncové konfiguraci))

pozn. u NKA se může stát že po přečtení řetězce skončíme v podmnožině množiny stavů, pokud je alespoň jeden z těchto stavů koncový, poté je řetězec přijat NKA

**Jazyk přijímaný KA** – pro jazyk přijímaný konečným automatem platí, že

$$L(M) = \{w : w \in T^*, (q_0, w) \vdash^* (q, \epsilon),\ q \in F\}$$

(neboli jedná se o všechny řetězce $w$ nad abecedou $T$, pro které platí, že jej daný automat přijímá)

**Úplný deterministický automat** – Deterministický konečný automat nazveme úplný, pokud je přechodová funkce $\delta$ definována pro všechny dvojice stavů $q \in Q$ a vstupních symbolů $a \in T$.

(tj. v tabulce nejsou prázdná místa a v diagramu se mi nemůže stát, že nevím kam jít z nějakého stavu pro nějaký symbol)

**Dosažitelný a nedosažitelný stav** – Stav $q \in Q$ nazveme dosažitelný pokud existuje řetězec $w \in T^*$ takový, že existuje posloupnost přechodů, která vede z počátečního stavu $q_0$ do stavu $q$.

Stav, který není dosažitelný se nazývá nedosažitelný.

**Užitečný a zbytečný stav** – Stav $q \in Q$ nazveme užitečný pokud existuje řetězec $w \in T^*$ takový, že existuje posloupnost přechodů, která vede ze stavu $q$ do nějakého koncového stavu.

Stav, který není užitečný se nazývá zbytečný.