

### 3. přednáška LS 2006/2007

- Teď Á-čka s B-čkem opustíme a podíváme se, co jsme si přitom připomněli či poprvé pochopili ...
- Vezměme v potaz, že jsou mezi námi i tací, které Á-čko s B-čkem vůbec nezajímá, a jen by teď rádi úspěšně napsali zápočtovou písemku 21.3.2007.
- Dobrá zpráva :-):  
Vzhledem ke starostem některých studentů, že nezískají 100 bodů, když se ze závažného důvodu nebudou moci zúčastnit některé zápočtové písemky, platí toto:  
Zisk alespoň 4 bodů alespoň dvakrát zůstává v platnosti jako nutná podmínka zápočtu. Kdo podmínku splní, obdrží za písemky součet bodů ze tří svých nejlépe hodnocených písemek násobený  $4/3$ .  
Kdo napíše všechny 4 písemky za 4 a více bodů, obdrží blíže nspecifikovaný bonus, ke kterému může být přihlédnuto při závěrečném hodnocení.

- Zápočtové písemky se zaměřují na základní znalosti a dovednosti, které jsou nezbytné k hlubšímu pochopení.
- (Některé 'hlubší' příklady se objeví na zkouškové písemce.)
- Poznámky ke cvičení, samostudiu apod. ....
- Teď provedeme jakýsi '[sebetest](#)' před 1. zápočtovou písemkou. Zhruba se budeme držet následujících bodů ...

- Rozumím detailně např. následujícímu zápisu (tedy všem jeho aspektům?)

$L_0 = \{w \in \{0, 1\}^* \mid \text{jestliže číslo s binárním zápisem } w \text{ je dělitelné pěti, pak } w \text{ neobsahuje podslovo } 010\}$

- Umím tedy v první řadě vysvětlit, co je to (formální) slovo nad danou abecedou, co je to prefix, sufix, podslovo, zřetězení slov, zrcadlový obraz slova, atd. ... ?
- Víím, že slov nad danou konečnou abecedou je spočetně mnoho? Mám např. představu, jak bych napsal program k vygenerování (potenciálně) všech takových slov, např. pro abecedu  $\Sigma = \{a, b\}$  ?
- Umím vysvětlit, co je to (formální) jazyk nad danou abecedou? Víím, že takových jazyků je nespočetně mnoho a nás v praxi zajímá jen 'malý zlomeček' z nich?

- Je mi jasné, jak lze běžně zadat (popsat) i nekonečný jazyk  $L \subseteq \Sigma^*$  ? (popisem podmínky, kterou splňují právě slova v  $L$ ; může to samozřejmě být složitější podmínka konstruovaná z jednodušších např. pomocí logických spojek).

Umím tedy např. bez problémů vypsát prvních dvacet slov z dříve uvedeného jazyka  $L_0$  (v přirozeném pořadí, na němž byl založen onen algoritmus generující všechna slova)?

- Rozumím perfektně operacím s jazyky? (sjednocení, průnik, doplněk, zřetězení, iterace, zrcadlový obraz)? Je mi např. naprosto jasné (tedy umím vysvětlit), že  $\overline{L_1 \cap L_2} = \overline{L_1} \cup \overline{L_2}$ ,  $(L^*)^* = L^*$ ,  $(L_1 L_2)^R = (L_2)^R (L_1)^R$ , apod.?

Rozumím také operaci kvocientu, speciálně (jednoduchému) typu  $w \setminus L$  ?

Co je např.  $110 \setminus L$ , kde  $L = \{w \in \{0, 1\}^* \mid w \text{ neobsahuje podslovo } 010\}$  ?

A co je  $110 \setminus L$ , kde  $L = \{w \in \{0, 1\}^* \mid \text{číslo s binárním zápisem } w \text{ je dělitelné pěti}\}$  ?

- Umím vysvětlit, co je to konečný automat (KA) jakožto rozpoznávač jazyků? Umím to vše popsat i formálně (matematicky)?
- Jakými způsoby umím zadat konkrétní KA ?
- Umím rychle navrhnout KA pro jednoduché jazyky? Jsem si vědom, že přitom (přinejmenším podvědomě) pracuji s kvocienty typu  $a \setminus L$ , tj. s jazyky  $L_q^{out-to-F}$  ?  
Zkusme např.  $L_1 = \{w \in \{0,1\}^* \mid \text{ve } w \text{ je každý výskyt podslova } 11 \text{ bezprostředně následován znakem } 0\}$   
Či např.  $L_2 = \{w \in \{0,1\}^* \mid |w|_1 \text{ je dělitelné pěti } \}$

- Umím pracovat modulárně ? Tj. umím si úkol přirozeně rozdělit podle booleovské struktury podmínky na příslušné množinové operace s jazyky? A umím pak se sestavených (menších) automatů konstruovat odpovídající 'kombinované'? Umím pro to napsat algoritmy? (Např. stručně matematicky?)
- Je mi jasné, že např. při rutinní modulární konstrukci snadno vzniknou nedosažitelné stavy? Umím vysvětlit a aplikovat algoritmus, který je odstraní? Umím přitom převést automat do normovaného tvaru?
- (Jako vždy, nejlépe si to zodpovím, když si vše sám prověřím na příkladech; nestačí pokývat hlavou nad pozorovaným postupem druhého ...)

- Je mi jasné, že kromě odstranění nedosažitelných stavů existuje i jiný zdroj možného zmenšení počtu stavů KA (při zachování rozpoznávaného jazyka)?
- Umím v daném automatu (samozřejmě opět algoritmicky) zjistit a 'splácnout' všechny stavy se stejným  $L_q^{out-to-F}$ ? Je mi jasné, že přitom de facto konstruuji zjemňující se ekvivalence  $\sim_0, \sim_1, \sim_2, \dots$  na množině stavů (až dosáhnu 'pevný bod')? Je mi jasné, co např. znamená  $q \not\sim_3 q'$ ?

# Užitečná charakterizace (ne)regulárních jazyků

- Chápu alespoň intuitivně, že jazyk  $L \subseteq \Sigma^*$  je regulární právě tehdy, když množina  $\{w \setminus L \mid w \in \Sigma^*\}$  je konečná?
- Vidím pak, že např. jazyky  $\{a^n b^n \mid n \geq 0\}$ ,  $\{w \in \{a, b\}^* \mid |w|_a = |w|_b\}$  jsou neregulární ?
- A co např. jazyk  $\{w \in \{a, b\}^* \mid \text{počet výskytů } ab \text{ ve } w \text{ je stejný jako počet výskytů } ba\}$  ?
- Vybuodoval jsem si tedy dobrou intuici k rozpoznání toho, zda je zadaný jazyk regulární či neregulární?



- Umím vysvětlit pojem (zobecněného) nedeterministického automatu? Umím vysvětlit, k čemu to může být dobré?
- Umím např. rychle navrhnout NKA rozpoznávající následující jazyk?  
 $L_3 = \{w \in \{0, 1\}^* \mid w \text{ obsahuje } 010 \text{ nebo jeho třetí symbol od konce je } 1\}$
- Umím převést NKA na ekvivalentní DKA ? (Umím tedy také popsat příslušný algoritmus?)
- Jsem si vědom, že obecně může při onom převodu vzrůst počet stavů exponenciálně? Vyhýbám se při konstrukci alespoň případným nedosažitelným stavům?

- Rozumím následujícímu regulárnímu výrazu?

$$(01)^*111 + (00 + 1)^*$$

Umím tedy např. rychle vypsát prvních deset slov z jazyka, který je tím výrazem popsán?

- Je mi jasné, že při úplném vyznačení operací a uzávorkování by výraz vypadal např. takto:

$$((((0 \cdot 1)^* \cdot 1) \cdot (1 \cdot 1)) + ((0 \cdot 0) + 1)^*)$$

A je mi jasné, proč je ta zjednodušená forma ekvivalentní, tj. proč popisuje tentýž jazyk?

- A je mi jasné, že mé intuitivní pochopení pojmu regulárního výrazu a jazyka jím reprezentovaného, je přesně zachyceno následující induktivní definicí?

$RV(\Sigma)$  ... množina *regulárních výrazů* nad  $\Sigma$ ; induktivní definice:  
 $RV(\Sigma)$  je nejmenší množina splňující:

$$\emptyset \in RV(\Sigma), \quad \varepsilon \in RV(\Sigma), \quad a \in RV(\Sigma) \quad (\text{pro } a \in \Sigma);$$

když  $\alpha, \beta \in RV(\Sigma)$ , pak také

$$(\alpha + \beta) \in RV(\Sigma),$$

$$(\alpha \cdot \beta) \in RV(\Sigma),$$

$$(\alpha^*) \in RV(\Sigma).$$

Reg. výraz  $\alpha$  reprezentuje jazyk  $[\alpha]$ :

$$[\emptyset] = \emptyset, \quad [\varepsilon] = \{\varepsilon\}, \quad [a] = \{a\},$$

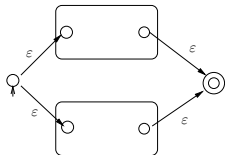
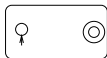
$$[(\alpha + \beta)] = [\alpha] \cup [\beta],$$

$$[(\alpha \cdot \beta)] = [\alpha] \cdot [\beta],$$

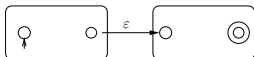
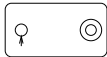
$$[(\alpha^*)] = [\alpha]^*.$$

- Je mi jasné, jak zkonstruuji jednoduché ZNKA s jedním (nepřijímajícím) počátečním stavem a jedním přijímajícím stavem pro jazyky  $\emptyset$ ,  $\{\varepsilon\}$ ,  $\{a\}$  ?
- Rozumím tomu, že konstrukce ZNKA k danému regulárnímu výrazu je dostatečně zachycena následujícím obrázkem? Umím tuto konstrukci tedy snadno aplikovat na konkrétních případech?

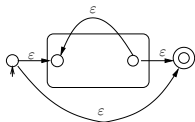
### Sjednocení



### Zřetězení



### Iterace



De facto další modulární konstrukce (vedle průniku, sjednocení, doplňku, ...)

- Je mi jasné, jak k dvěma automatům  $A_1, A_2$  sestrojím automat pro jazyk  $L(A_1) \cdot L(A_2)$  ?
- Je mi jasné, jak k automatu  $A_1$  sestrojím automat  $A_2$  pro jazyk  $L(A_1)^*$  ?
- Je mi jasné, jak k automatu  $A_1$  sestrojím automat  $A_2$  pro jazyk  $L(A_1)^R$  ?
- Chápu tedy také, co to znamená, že třída REG je uzavřena vůči operacím sjednocení, průniku, doplňku, zřetězení, iterace, zrcadlového obrazu ... ?