

Týden 1

Po přednášce 14.9. bude text asi mírně modifikován, ale příklady na cvičení se nezmění.
(Navíc dodám na web slidy.)

Přednáška

Na začátku jsme si připomněli základní informace o kursu a podmínkách jeho absolvování, které jsou všechny v Edisonu a na web-stránce předmětu.

Zdůraznil jsem ještě tuto věc: Přednáška se v zásadě bude soustřeďovat na ilustraci základních myšlenek z probírané problematiky v součinnosti s aktivně se účastnícími posluchači. Nepůjde o promítání a recitaci toho, co mají všichni k dispozici v základním studijním textu (na nějž se vždy v popisu průběhu výuky odkazujeme, není-li řečeno jinak) a v příslušných animacích. Je **velmi žádoucí**, aby si posluchači vždy sami prošli příslušné odkazované partie v textu, uvědomili si případné problémy a nejasnosti a připravili se na cvičení, kde je primární prostor pro vyřešení případných problémů a nejasností. (Cvičení nenahrazuje vaše samostudium!)

Konečné automaty

Připomněli jsme si pojem konečného automatu jako (jednoduchého) *modelovacího nástroje* a ukázali souvislost s programy s „malou“ pamětí.

Konečný automat při vyhledávání vzorku v textu

Zkonstruovali jsme konečný automat, který je základem (efektivního) algoritmu pro vyhledání všech výskytů vzorku *abaaba* v (dlouhém) řetězci písmen z abecedy $\{a, b\}$. Řešili jsme tedy „úkol“ U_0 z části 3.1. (studijního textu). Dospěli jsme ke konečnému automatu se stavy U_0, U_1, \dots, U_6 . Při konstrukci se může osvědčit značení

$U_0 \dots |abaaba \dots 0$ (potenciálně jsme právě přečetli “0-tou pozici” vzorku)

$U_1 \dots |a|baaba \dots 0, 1$ (potenciálně jsme právě přečetli 1. nebo 0. pozici vzorku)

...

$U_6 \dots |a|ba|aba| \dots 0, 1, 3, 6$ (přečetli jsme 6., 3., 1., nebo 0. pozici vzorku)

Použili jsme obecný algoritmus zachycený tímto pseudokódem (pro vstup *patt*: array[1..*d*] of char):

$Next[0, patt[1]] := 1; \forall x \in \Sigma \setminus \{patt[1]\} : Next[0, x] := 0; Sec[1] := 0;$
for $i := 1$ to $d-1$ do

$$\begin{aligned} \text{Next}[i, \text{patt}[i+1]] &:= i+1; \\ \forall x \in \Sigma \setminus \{\text{patt}[i+1]\} : \text{Next}[i, x] &:= \text{Next}[\text{Sec}[i], x]; \\ \text{Sec}[i+1] &:= \text{Next}[\text{Sec}[i], \text{patt}[i+1]]; \end{aligned}$$

$$\forall x \in \Sigma : \text{Next}[d, x] := \text{Next}[\text{Sec}[d], x];$$

(Demonstrovali jsme tak de facto variantu algoritmu “Knuth - Morris - Pratt”.)

Připomněli jsme si pak definici konečného automatu jako matematické struktury

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

a definici jazyka rozpoznávaného (přijímaného) automatem A

$$L(A) = \{w \in \Sigma^* \mid \text{slovo } w \text{ je přijímáno } A\} = \{w \in \Sigma^* \mid q_0 \xrightarrow{w} F\}$$

a uvědomili jsme si, že jsme vlastně zkonstruovali automat přijímající jazyk

$$\{w \in \{a, b\}^* \mid w \text{ má sufix } abaaba\}.$$

Modulární postup při návrhu konečných automatů

Zkonstruovali jsme konečný automat rozpoznávající jazyk

$$L_0 = \{w \in \{0, 1\}^* \mid \text{bn}(w) \text{ je dělitelné třemi a } w \text{ neobsahuje podřetězec } 101\},$$

kde $\text{bn}(w)$ znamená číslo, pro něž je w jeho binárním zápisem (např. $\text{bn}(0101) = 5$; definujeme také $\text{bn}(\varepsilon) = 0$).

Postupovali jsme tak, že jsme nejdříve zkonstruovali (4-stavový) automat A'_2 pro jazyk

$$L'_2 = \{w \in \{0, 1\}^* \mid w \text{ obsahuje podřetězec } 101\},$$

ten jsme transformovali na (4-stavový) automat A_2 rozpoznávající doplněk jazyka L'_2 , tj. jazyk

$$L_2 = \{0, 1\}^* - L'_2 = \{w \in \{0, 1\}^* \mid w \text{ neobsahuje podřetězec } 101\},$$

pak jsme zkonstruovali (3 stavový) A_1 pro jazyk

$$L_1 = \{w \in \{0, 1\}^* \mid \text{bn}(w) \pmod 3 = 0\}$$

(uvědomili jsme si, že $\text{bn}(uv) = \text{bn}(u) \cdot 2^{|v|} + \text{bn}(v)$ (speciálně $\text{bn}(u0) = \text{bn}(u) \cdot 2$ a $\text{bn}(u1) = \text{bn}(u) \cdot 2 + 1$), takže z dosud přčteného prefixu u si stačí pamatovat hodnotu $\text{bn}(u) \pmod 3$) a na závěr jsme naznačili konstrukci „kartézského součinu“, která k automatům A_1, A_2 vyrobí (12-stavový) automat A rozpoznávající $L(A_1) \cap L(A_2) = L_0$.

Induktivní definice

Pobavili jsme se o induktivních definicích a ilustrovali jsme příkladem zavedení značení (či ternárního predikátu) $q \xrightarrow{w} q'$, jak uvedeno v „matematické poznámce“ na str. 58.

(Levý) kvocient jazyka podle slova

Uvědomili jsme si, že konstrukce konečných automatů úzce souvisí s operací (levého) kvocientu jazyka podle slova. Ilustrovali jsme definici

$$u \setminus L = \{v \mid uv \in L\},$$

uvedli značení (k danému $A = (Q, \Sigma, \delta, q_0, F)$)

$$L_q^{toAcc} = \{v \in \Sigma^* \mid q \xrightarrow{v} F\} \quad (\text{tedy } L(A) = L_{q_0}^{toAcc})$$

a intuitivně jsme pochopili, že

$$\text{když máme } q_0 \xrightarrow{w} q, \text{ tak platí } L_q^{toAcc} = w \setminus L_{q_0}^{toAcc} = w \setminus L(A).$$

Pak jsme vyslovili následující větu (3.18 z části 3.8., str. 87).

Věta. Jazyk $L \subseteq \Sigma^*$ je regulární (tzn. přijímaný konečným automatem) právě tehdy, když je množina kvocientů $\{w \setminus L \mid w \in \Sigma^*\}$ konečná.

Ukázali jsme implikaci zleva-doprava a začali diskutovat opačnou implikaci. Příště dokončíme, také s využitím věty pro důkaz, že jazyk $\{a^n b^n \mid n \geq 1\}$ regulární není.

Partie textu k prostudování

Posluchači by si měli projít Kapitulu 2 (Formální jazyky) a sekce 3.1., 3.2., 3.3. (Tzn. udělat si přinejmenším dobrou první představu a zamyslet se nad příklady, speciálně těmi plánovanými na cvičení, ať se mohou na cvičení aktivně účastnit a případné problémy si tam objasnit.)

Cvičení

Příklad 1.1

Vypište prvních deset slov z jazyka $L = \{w \in \{a, b\}^* \mid \text{každý výskyt podřetězce } aa \text{ je ve } w \text{ ihned následován znakem } b\}$. Odkazujeme se k uspořádání $<_L$, tedy slova máme uspořádána (vzestupně) podle délky a v rámci stejné délky abecedně, přičemž předpokládáme abecední uspořádání $a < b$.

Příklad 1.2

Definujte operaci zřetězení jazyků $L_1 \cdot L_2$.

Připomeňte si, že pro jazyk L lze definovat L^* (iterace jazyka L) např. takto:

$$L^* = \{w \mid \text{existuje } n \text{ a slova } v_1, v_2, \dots, v_n \text{ v } L \text{ tak, že } w = v_1 v_2 \dots v_n\}$$

(z případu $n = 0$ vyplývá, že $\varepsilon \in L^*$).

Příklad 1.3

Charakterizujte slova jazyka $L = L_0 \cup L_1$, kde L_0 je množina všech slov obsahujících více 0 než 1 (tedy $L_0 = \{w \in \{0, 1\}^* \mid |w|_0 > |w|_1\}$) a L_1 je množina všech slov obsahujících více 1 než 0 (tedy $L_1 = \{w \in \{0, 1\}^* \mid |w|_0 < |w|_1\}$).

Jaká slova patří do iterace jazyka L , tedy do L^* ?

Příklad 1.4

Jaká slova patří do jazyka L_1^R , tj. jazyka, který je zrcadlovým obrazem jazyka $L_1 = \{\varepsilon, a, abb, baaba\}$?

Jaký je jazyk L_2^R pro $L_2 = \{w \mid |w|_a \bmod 2 = 0\}$?

Platí pro tyto jazyky $(L_1 L_2)^R = (L_1)^R (L_2)^R$?

Platí $(L_1 L_2)^R = (L_2)^R (L_1)^R$? Platí tento vztah obecně pro jakékoli jazyky L_1, L_2 ?

Příklad 1.5

Navrhněte konečný automat A tak, že $L(A) = L$, kde $L = \{w \in \{a, b\}^* \mid \text{každý výskyt podslova } aa \text{ je ve } w \text{ ihned následován znakem } b\}$.

Charakterizujte jazyky $a \setminus L, b \setminus L$.

Příklad 1.6

Zkonstruujte modulárně konečný automat přijímající jazyk

$L = \{w \in \{0, 1\}^* \mid w \text{ obsahuje podslovo } 010 \text{ nebo } |w|_1 \text{ je sudé}\}$.

Později se zamysleme, zda půjde počet stavů výsledného automatu zmenšit.

Příklad 1.7

Zjistěte, zda obecně platí některý ze vztahů

$$(uv) \setminus L = u \setminus (v \setminus L),$$

$$(uv) \setminus L = v \setminus (u \setminus L).$$

Dokažte svá zjištění.

Příklad 1.8

Vybudujte si intuici o (ne)regulárních jazycích promyšleným zodpovězením otázek na s. 88 a 89. Ty se týkají (ne)regularity následujících jazyků. (Zápisem w^R značíme zrcadlový obraz [reversal] slova w ; induktivně lze definovat takto: $\varepsilon^R = \varepsilon$; $(au)^R = u^R a$.)

$$\{w \in \{a, b\}^*; |w|_a \bmod 2 = 0\}$$

$$\{w \in \{a, b\}^*; w \text{ začíná nebo končí dvojicí stejných písmen}\}$$

$$\{w \in \{a, b\}^*; |w|_a < |w|_b\}$$

$\{w \in \{a, b, c\}^*; \text{jestliže } w \text{ neobsahuje podřetězec } abc, \text{ pak končí } bca\}$

$\{w \in \{a, b\}^*; |w|_a > |w|_b \text{ nebo } w \text{ končí } baa\}$

$\{w \in \{a, b\}^*; |w|_a > |w|_b \text{ nebo } |w|_b \geq 2\}$

$\{u; \text{ex. } w \in \{a, b\}^* \text{ tak, že } u = ww^R\}$, stručněji také psáno $\{ww^R; w \in \{a, b\}^*\}$,

$\{w \in \{a, b\}^*; w = w^R\}$

$\{w \in \{a\}^*; w = w^R\}$

$\{ww; w \in \{a, b\}^*\}$

$\{ww; w \in \{a\}^*\}$

$\{w \in \{a, b\}^*; \text{rozdíl počtů znaků } a \text{ a znaků } b \text{ ve } w \text{ je větší než } 100\}$

$\{w \in \{a, b\}^*; \text{součin } |w|_a \text{ a } |w|_b \text{ je větší nebo roven } 100\}$

$\{w \in \{a\}^*; |w| \text{ je prvočíslo } \}$