Cvičení 1

Příklad 0.1

Zvažte následující algoritmus (zapsaný pseudokódem), který vyhledává výskyty řetězce *abaaba* v textových souborech (obsahujících jen symboly a, b). (Jedná se o původní návrh B-čka z přednášky.)

```
procedure SEARCH (var F: file)
const length = 6            (* delka hledaneho retezce *)
const P = [ a,b,a,a,b,a ]   (* hledany retezec *)
var A: array [ 1..length ] of char
begin
  for i:=1 to length do
    read( A[i], F ); if EOF (* end of file *) then return
  endfor
  while true do
    if EQUAL(P,A) then ``vypis misto vyskytu''
    for i:=1 to length-1 do
      A[i]:=A[i+1]
    endfor
    read( A[length], F ); if EOF then return
  endwhile
end
```

- Je možné se na proceduru `SEARCH` přirozeně dívat jako na konečný automat? Pokud ano, kolik má takový automat stavů?
- Proceduru jistě snadno zobecníte pro případ hledání jakéhokoli zadaného řetězce (jakékoli délky). Jaká je přímočará realizace booleovské funkční (pod)procedury `EQUAL` ?
- Uveďte příklad (vzorku a textu), kdy zobecněná procedura `SEARCH` je očividně neoptimální.
- Připomeňte si návrh Á-čka z přednášky; tedy zkonstruujte konečný automat $A = (Q, \Sigma, \delta, q_0, F)$ rozpoznávající jazyk

  $L_0 = \{ u \in \{a, b\}^* \mid u \text{ má sufix } abaaba \}$.

  (Jak to, že je to vlastně automat přijímající jen slova končící *abaaba*, když nám jde o hledání všech výskytů řetězce *abaaba* ?)
- Připomeňte si, že Á-čko označil stavy automatu A symboly $0, 1, 2, 3, 5, 6$, ale ve svých poznámkách měl napsáno $(0 = \varepsilon), (1 = a), (2 = ab), (3 = aba), (4 = abaa), (5 = abaab), (6 = abaaba)$. K čemu mu to vlastně bylo dobré ? (Odpověď souvisí s dalším úkolem.)

- Pro výše zmíněný automat A charakterizujte množinu slov $L_3^{in-from-q_0} = \{ u \mid \delta^*(0,u) = 3 \}$ (v jiném značení $\{ u \mid 0 \xrightarrow{u} 3 \}$). Pak charakterizujte množinu slov $L_3^{out-to-F} = \{ u \mid \delta^*(3,u) \in F \}$ (v jiném značení $\{ u \mid 3 \xrightarrow{u} F \}$). Jak spolu $L_3^{in-from-q_0}$ a $L_3^{out-to-F}$ souvisejí?

- Připomeňte si definici operace levého kvocientu na jazycích, tedy obecně $L_2 \backslash L_1 = \{ v \mid \exists u \in L_2 : uv \in L_1 \}$. Charakterizujte jazyk $\{aba\} \backslash L_0$ (k našemu dříve uvedenému L_0). Vidíte nějaký vztah mezi L_0, $L_3^{in-from-q_0}$, $L_3^{out-to-F}$?

- Charakterizujte analogicky jazyky $L_i^{in-from-q_0}$ a $L_i^{out-to-F}$ pro $i = 0, 1, \ldots, 6$. Jsou jazyky $L_0^{out-to-F}$, $L_1^{out-to-F}$, $\ldots$, $L_6^{out-to-F}$ vzájemně disjunktní? A jak je to pro jazyky $L_1^{in-from-q_0}$, $L_2^{in-from-q_0}$, $\ldots$, $L_6^{in-from-q_0}$?

- Připomeňte si, co to je rozklad na množině a jak tento pojem souvisí s relací ekvivalence. Který ze systémů $L_0^{out-to-F}$, $L_1^{out-to-F}$, $\ldots$, $L_6^{out-to-F}$ a $L_1^{in-from-q_0}$, $L_2^{in-from-q_0}$, $\ldots$, $L_6^{in-from-q_0}$ je rozkladem na množině $\{0,1\}^*$? Platí analogie i pro obecný konečný automat? Jestli ano, zformulujte ji přesně.

Příklad 0.2

Zjistěte, zda obecně platí: $\{u\} \backslash (\{v\} \backslash L) = \{uv\} \backslash L$.

Příklad 0.3

Uvažujme jazyk
$L_1 = \{ u \in \{0,1\}^* \mid bn(u) \mod 3 = 0 \}$
kde $bn(u)$ je číslo s binárním zápisem u. ($bn(\varepsilon)$ je chápáno jako 0.)
Zkonstruujte jazyky $\{u\} \backslash L_1$ pro všechna slova u s délkou $|u| \leq 3$.
Potom ukažte konečný automat A tž. $L(A) = L_1$.

Příklad 0.4

- Připomeňte si metodu důkazu matematickou indukcí a formu induktivní definice.

- Mějme konečný automat $A = (Q, \Sigma, \delta, q_0, F)$. Připomeňte si typ funkce δ a induktivně definujte přirozené rozšíření $\delta^* : Q \times \Sigma^* \to Q$.

- Dokažte matematickou indukcí podle délky slova, že máme-li automaty

  $A_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$,

  $A_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$

  a automat

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

  - $Q = Q_1 \times Q_2$

– pro každé $q_1 \in Q_1$, $q_2 \in Q_2$, $a \in \Sigma$:
$\delta(\,(q_1, q_2), a\,) = (\,\delta_1(q_1, a), \delta_2(q_2, a)\,)$

tak $(\forall q_1 \in Q_1, q_2 \in Q_2, w \in \Sigma^*)$

$\delta^*(\,(q_1, q_2), w\,) = (\,\delta_1^*(q_1, w), \delta_2^*(q_2, w)\,)$.

Příklad 0.5

- Dokažte mat. indukcí, že žádné slovo u nesplňuje vztah

$$au = ub$$

pro dva různé symboly a, b. (Podle čeho povedete indukci ?)

Příklad 0.6

- Připomeňte si, jak je definována relace (částečného a úplného) uspořádání na množině.
- Popište některá přirozená (úplná či částečná) uspořádání na množině slov dané abecedy a na množině všech jazyků nad danou abecedou. Zamyslete se i nad nabízejícími se kvaziuspořádáními. (Relace kvaziuspořádání nemusí být antisymetrická.)
- Kolik je slov v abecedě $\Sigma = \{a, b\}$ a kolik je jazyků nad abecedou Σ ? Kolik je regulárních jazyků nad abecedou Σ ?

Referát č. 1

Vysvětlete, co znamená tvrzení, že operace levého kvocientu je asociativní. Pak toto tvrzení pečlivě dokažte či vyvraťte.
Dále vysvětlete, proč pro konečný automat $A = (Q, \Sigma, \delta, q_0, F)$ a jazyk L je $L \backslash L(A)$ sjednocením jazyků $L_q^{out-to-F}$ pro vybrané stavy q. Pro které ?

Referát č. 2

- Připomeňte, co je to homomorfismus mezi (relačními a algebraickými) strukturami (uveďte více konkrétních příkladů), a také definujte homomorfismus mezi konečnými automaty. Dále připomeňte, co to znamená, že dvě (relační a algebraické) struktury jsou izomorfní. Pak konkrétně aplikujte na konečné automaty (vysvětlete, co to znamená, že dva automaty jsou izomorfní).
- Popište a na příkladu ilustrujte (rychlý) algoritmus testující, zda dané dva automaty jsou izomorfní.