

Úlohy 2 (24.2.2020)

Termín do najbližších cvičení, najneskôr **3. marec 2020**, na cvičeniach. Každý príklad píše na samostatný papier A4! Nezabudnite sa podpísať a uviesť skupinu kam chodíte na cvičenia (meno cvičiaceho resp. čas cvičenia a miestnosť). Ak máte riešenie príkladu na viacerých papieroch, vhodne ich zopnite.

Vždy uveďte aj zdôvodnenie (dôkaz) vášho riešenia! (Nestačí len áno/nie alebo číslo.)

Opísané riešenia sú za 0b (aj opisované aj opísané, nebudeme zisťovať čo je originál).

Definícia: Nech Σ je abeceda. Majme $A \subseteq \Sigma$. $A^0 = \{\lambda\}$, $A^1 = A$ a pre $n \in \mathbb{N}$ je $A^{n+1} = AA^n$.

$$A^+ = \bigcup_{n \in (\mathbb{N} - \{0\})} A^n \text{ a } A^* = A^+ \cup \{\lambda\}.$$

Definícia: Ľavý kvocient jazykov A a B je definovaný

$$A^{-1}B = \{y \mid \exists x \in A, xy \in B\}$$

Napríklad pre $L = \{a^i b^i \mid i \geq 1\}$ je $\{a^2\}^{-1}L = \{a^i b^{i+2} \mid i \geq 0\}$.

príklad 1.

Určite $L_1^{-1}L_2$ a $L_2^{-1}L_1$ keď

- $L_1 = \{a^i b^i \mid i \geq 0\}$, $L_2 = \{a^i b^j \mid 0 \leq i \leq j\}$
- $L_1 = \{ww \mid w \in \{0, 1\}^*\}$, $L_2 = \{ww^R \mid w \in \{0, 1\}^*\}$
- $c \notin \{0, 1\}$, $L_1 = \{w c w \mid w \in \{0, 1\}^*\}$, $L_2 = \{w c w^R \mid w \in \{0, 1\}^*\}$

príklad 2.

Dokážte alebo vyvráťte:

- L_1 a L_2 sú jazyky. Ak $L_1^* = L_2^*$, tak $L_1 = L_2$.
- Sú jazyky $\{w \in \{a, b\}^* \mid |w| \bmod 2 = 0\}$ a $\{xy \in \{a, b\}^* \mid |x| = |y|\}$ rovnaké?

Pomocou jazykov $L_a = \{a\}$, $L_b = \{b\}$, množinových operácií a zretazenia vyjadrite jazyk, ktorý obsahuje práve slová (nezabudnite zdôvodniť/dokázať)

- ktorých dĺžka je deliteľná 3;
- ktoré začínajú a končia rôznym písmenom a majú nepárnu dĺžku.

príklad 3.

Tento príklad pošlite e-mailom na winczer@fmph.uniba.sk do predmetu dajte uti

Slová môžeme interpretovať rôznym spôsobom, táto úloha je jeden z príkladov ako sa to dá urobiť.

Uvažujme abecedu $\Sigma = \{0, 1\}$ postupnosť slov na abecedou Σ : $w_0 = 0$, $w_i = \rho(w_{i-1})$, $i > 0$, kde $\rho(0) = 01$ a $\rho(1) = 0$. Napríklad: $w_1 = 01$, $w_2 = 010$, $w_3 = 01001$, ...

Slovo $c_0c_1c_2c_3\dots$, kde $c_i \in \Sigma$ budeme interpretovať korytnačkou tak, že budeme postupne čítať jeho symboly z ľava, c_0, c_1, \dots . Keď korytnačka číta symbol 1, posunie sa o krok vpred. Ak číta symbol na pozícii j a $c_j = 0$, otočí sa o 90 stupňov vľavo keď je j párne, a o 90 stupňov vpravo keď je j nepárne.

Definujte v Pythone:

funkciu `slovo(n)`, ktorej výsledok bude slovo w_n (nebojte sa využiť generátor),

funkciu `vykresli(s,krok)`, ktorá vykreslí korytnačkou slovo s , pričom veľkosť kroku korytnačky určí hodnota `krok`.

funkciu `prvychkod(k,m,s)`, ktorej výsledok bude slovo $c_m c_{m+1} \dots c_{m+k-1}$, keď $s = c_0 c_1 c_2 \dots$, $c_i \in \Sigma$. Môžete predpokladať, že $|s| > k + m$.

Váš program by mal pracovať pri štandardných nastaveniach aj pre `vykresli(prvychkod(1000, 1000000, slovo(500)), 1.5)`

Nezabudnite uviesť aj testovacie dáta.