Úvod do regulárních výrazů

Poslední kapitolou týkající se regulárních jazyků je téma zabývající se tzv. regulárními výrazy. Tento pojem je v praxi velmi známý, určitě se s ním setkáte, pokud budete provádět pokročilejší vyhledávání v „sofistikovaných“ textových editorech. Např. v Microsoft Wordu pracujete s regulárními výrazy, pokud používáte zástupné symboly pro hledání určitých řetězců.

Uživatelé unixových systémů by regulární výrazy měli také znát, pokud používají při vyhledávání „vzory“. Podporu regulárních výrazů má v sobě zabudován příkaz `grep` a jeho varianta `egrep`, případně textové editory `vi`, `ed` či `sed`.

My se budeme regulárními výrazy zabývat „minimalistickou cestou“, čímž míníme, že nezavedeme příliš mnoho nástrojů pro skládání jednodušších výrazů do složitějších. Nejdříve si uvedeme definici množiny regulárních výrazů.

Definice – množina regulárních výrazů

Množinu regulárních výrazů nad abecedou Σ, která je označována $RE(\Sigma)$, definujeme induktivně takto:

1. $\varepsilon,\ \emptyset,\ a$ pro každé $a \in \Sigma$ jsou regulární výrazy nad Σ (tzv. základní regulární výrazy).
2. Jestliže E, F jsou regulární výrazy, pak také $(E.F),\ (E+F),\ (E^*)$ jsou regulární výrazy nad Σ.
3. Každý regulární výraz vznikne konečným opakováním kroků 1, 2.

Poznámka 1.

Předchozí definice se zabývala syntaxí regulárních vyrazů, tj. jak se zapisují. Následující popisuje sémantiku.

Definice – sémantika regulárních výrazů

Každý regulární výraz E nad abecedou Σ popisuje jazyk $L(E) \subseteq \Sigma^*$, a to podle těchto pravidel:

$$\begin{aligned} L(\varepsilon) &= \{\varepsilon\} \\ L(\emptyset) &= \emptyset \\ L(a) &= \{a\} \text{ pro každé } a \in \Sigma \\ L(E.F) &= L(E).L(F) \\ L(E+F) &= L(E) \cup L(F) \\ L(E^*) &= L(E)^* \end{aligned}$$

Poznámka 2.

Pro usnadnění zápisu regulárních výrazů zavedeme prioritu operátorů „.“, „+“ a „*“. Nejvyšší má „*“, poté „.“ a nakonec „+“. Závorky mají funkci metasymbolů, které nám pomáhají upřednostnit operátor s nižší prioritou před tím s vyšší. Např. výrazy $a+b.c,\ (a+b).c$ se vyhodnocují zcela jinak.

Příklad 1.

Pomocí regulárních výrazů vyjádřete jazyky

$$\begin{aligned} L_1 &= \{a,b\}^* \\ L_2 &= \{cuc \mid u \in \{a,b\}^*\} \\ L_3 &= \{w \in \{a,b\}^* \mid \#_b(w) \geq 3\} \\ L_4 &= \{w \in \{a,b\}^* \mid \#_b(w) = 2k+1,\ k \in \mathbb{N}_0\} \\ L_5 &= \{w \in \{a,b\}^* \mid w \text{ obsahuje podslovo } abb\} \end{aligned}$$

Řešení.

1. Jazyk $L_1 = \{a,b\}^*$ je množina všech slov nad abecedou $\{a,b\}$, do které přidáváme prvky opakovaným výběrem jednoho ze symbolů a, b. Čárku mezi nimi můžeme chápat jako výběr jednoho z nich, regulární výraz by měl být $(a+b)^*$. Jistě platí $L((a+b)^*) = \{a,b\}^*$.

2. Pokud jsme vyřešili předchozí jazyk L_1, bude vytvoření výrazu pro L_2 vcelku triviální:

$$c.(a+b)^*.c$$

3. V regulárním výrazu pro jazyk L_3 se musí objevit alespoň tři symboly b. Mezi nimi však může být libovolný počet písmen a, na což nesmíme zapomenout. Výsledek je tedy:

$$a^*.b.a^*.b.a^*.b.(a+b)^*$$

4. Důležitou vlastností jazyka L_4, že se v něm symbol b objeví alespoň jednou, potenciálně obklopen z obou stran libovolným počtem písmen a. V regulárním výrazu se tedy objeví podvýraz $a^*.b.a^*$. K němu poté přiřetězíme iterující výraz, v němž je dvakrát obsaženo písmeno b. Celý výraz vypadá takto:

$$a^*.b.a^*.(b.a^*.b.a^*)^*$$

5. Zapsání regulárního výrazu pro L_5 je triviální, v jeho prostředku se musí objevit řetězec abb, výsledkem je tedy:

$$(a+b)^*.a.b.b.(a+b)^*$$

Příklad 2.

Určete jazyky, které jsou vyjádřeny pomocí následujících regulárních výrazů nad abecedou $\{a, b\}$:

$$\begin{aligned} E_1 &= (a.a + a.b + b.a + b.b)^* \\ E_2 &= a.(a + b)^*.a + b.(a + b)^*.b \\ E_3 &= a.(a + b)^*.b.b.(a + b)^*.a \\ E_4 &= (a + b)^*.b.a.b \\ E_5 &= (a + b).((a + b).(a + b).(a + b))^* \end{aligned}$$

Řešení.

1. Všimněte si, že podstatou výrazu E_1 jsou čtyři potenciální dvojice, které vzniknou kombinací písmen a, b. Při každé iteraci jednu z nich vybereme, navíc provedeme-li opakování n-krát, bude počet symbolů vždy $2n$. Jazyk

$$L(E_1) = \{w \in \{a, b\}^*,\ |w| = 2k,\ k \in \mathbb{N}_0\}.$$

2. Je třeba si vzpomenout, že operátor „+“ má nejnižší prioritu, tj. nejdříve se provede vyhodnocení výrazů $a.(a + b)^*.a$, $b.(a + b)^*.b$ a potom teprve jejich sjednocení pomocí „+“. V případě obou výrazů se jedná o slova začínající a končící stejným symbolem. Jazyk

$$L(E_2) = \{uwu \mid w \in \{a, b\}^*, u \in \{a, b\}\}.$$

3. Výraz E_3 je zvláštní dvěma fakty: jednak začíná i končí symbolem a, a obsahuje podslovo bb. Můžeme tedy psát

$$L(E_3) = \{awa \mid w \in \{a, b\}^*, w \text{ obsahuje podslovo } bb\}.$$

4. Zde asi není třeba váhat, na konci výrazu je $b.a.b$, což znamená, že slovo $w \in L(E_4)$ musí končit řetězcem bab. Před ním je podvýraz $(a+b)^*$, což znamená $\{a, b\}^*$, tedy cokoliv nad abecedou $\{a, b\}$. Můžeme psát

$$L(E_4) = \{w \in \{a, b\}^* \mid w \text{ končí řetězcem } bab\}.$$

5. Výraz E_5 je opakující se posloupností podvýrazů $(a + b)$, přičemž se dá rozdělit do dvou částí: neopakujícího se prvního $(a + b)$, přičemž zbylé tři výskyty $(a + b)$ mohou libovolně krát iterovat za sebou. Pokud víme, že $(a + b)$ znamená výběr jednoho s písmen a, b, dá se snadno rozpoznat, že

$$L(E_5) = \{w \in \{a, b\}^*, |w| = 3k + 1,\ k \in \mathbb{N}_0\}.$$