IB102 – úkol 1 – řešení **Odevzdání: 27. 9. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

---

1. [2 body] Mějme následující jazyky nad abecedou $\Sigma = \{a, b\}$.

$$L_1 = \{a\} \cdot (\{\varepsilon\} \cup \{a\})$$
$$L_2 = \{b\} \cup (\emptyset \cdot \{b\})$$
$$L_3 = (L_1 \cdot L_2^*) \cap L_1^*$$

Zjistěte, kolik slov obsahuje jazyk L_3, a vypište je. Svou odpověď zdůvodněte.

Řešení: Jazyk L_3 obsahuje právě dvě slova, jsou to slova a a aa.
Zdůvodnění: Nejprve si přepíšeme jazyky L_1 a L_2 do čitelnější podoby:

$$\begin{aligned} L_1 &= \{a\} \cdot (\{\varepsilon\} \cup \{a\}) = \{a\} \cdot \{\varepsilon, a\} = \{a, aa\} \\ L_2 &= \{b\} \cup (\emptyset \cdot \{b\}) \quad = \{b\} \cup \emptyset \quad = \{b\} \end{aligned}$$

Nyní je zřejmé, že jazyk $L_1 \cdot L_2^*$ obsahuje právě všechna slova, která začínají jedním nebo dvěma symboly a, za nimiž následuje libovolný počet symbolů b (včetně nulového počtu), formálně tedy $\{a, aa\} \cdot \{b\}^*$. Jazyk L_1^* pak obsahuje právě všechna slova, která jsou tvořena pouze symboly a (včetně prázdného slova), formálně $L_1^* = \{a\}^*$. Jsou pouze dvě slova, která patří do obou těchto jazyků, a to jsou právě a a aa. Proto platí:

$$L_3 = (L_1 \cdot L_2^*) \cap L_1^* = \{a, aa\}$$

IB102 – úkol 1 – řešení **Odevzdání: 27. 9. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [**2 body**] Nechť L je jazyk nad abecedou $\Sigma = \{a, b\}$ tvořený právě všemi slovy, která mají sudý počet písmen a a zároveň sudý počet písmen b. Zapište jazyk L pomocí jednoprvkových jazyků $\{a\}$ a $\{b\}$ s využitím operací sjednocení ($\cup$), zřetězení ($\cdot$), průniku ($\cap$) a iterace (*). Chcete-li použít jiné operace nebo jazyky, musíte je nejprve definovat pomocí výše uvedených operací a jazyků.

Bonusová varianta za **4 body**: Zapište jazyk L bez použití operace průniku.

Řešení: Pro zpřehlednění zápisu si nejprve definujeme následující pomocné jazyky. Jazyk L_1 bude obsahovat všechna slova se sudým počtem a a jazyk L_2 bude obsahovat všechna slova se sudým počtem b.

$$L_1 = (\{b\}^* \cdot \{a\} \cdot \{b\}^* \cdot \{a\})^* \cdot \{b\}^*$$
$$L_2 = (\{a\}^* \cdot \{b\} \cdot \{a\}^* \cdot \{b\})^* \cdot \{a\}^*$$

Řešení pak můžeme snadno zapsat takto: $L = L_1 \cap L_2$.

Řešení bonusové varianty: Nejprve si definujeme následující pomocné jazyky:

$$L_1 = (\{a\} \cdot \{a\}) \cup (\{b\} \cdot \{b\})$$
$$L_2 = (\{a\} \cdot \{b\}) \cup (\{b\} \cdot \{a\})$$

Řešení pak můžeme zapsat například takto:

$$L = (L_1 \cup L_2 \cdot L_1^* \cdot L_2)^*$$

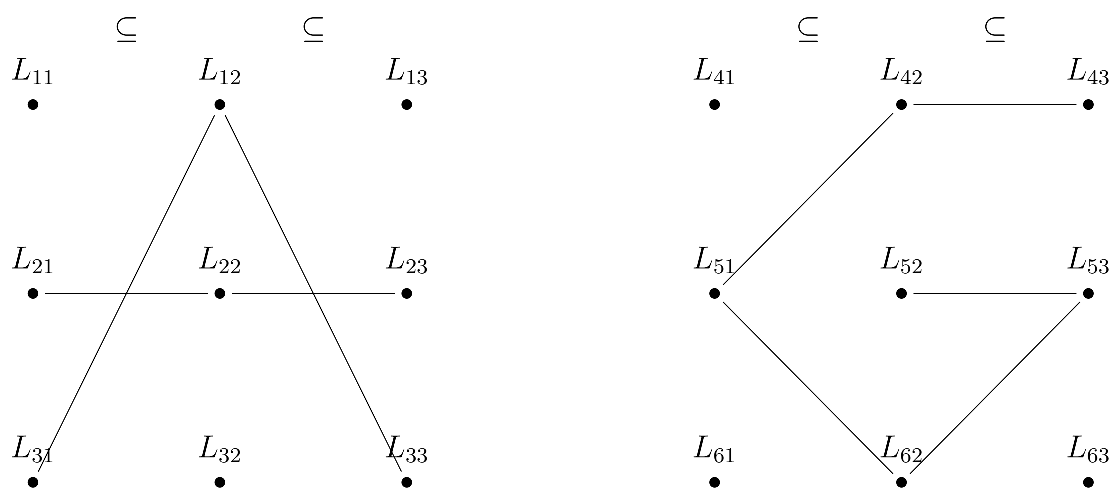
Vysvětlení: Slova, která mají sudý počet a i b, musí být nutně sudé délky. Každé takové slovo se tedy dá rozdělit na posloupnost dvojic písmen. Tyto dvojice jsou dvou typů – buď nemění paritu počtu a a b (to jsou dvojice aa a bb), nebo mění paritu počtu a i paritu počtu b (to jsou dvojice ab a ba). Dvojic prvního typu může být ve slově libovolný počet, dvojic druhého typu musí být nutně ve slově sudý počet. (Paritou míníme sudost/lichost počtu písmen.)

IB102 – úkol 2 – řešení **Odevzdání: 4. 10. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

1. [**3 body**] Na obrázku máte 2 čtverce, v každém z nich 9 jazyků nad abecedou $\{a, b\}$. V sousedních sloupcích spojte čarou ty jazyky, mezi kterými je relace inkluze (tzn. levý jazyk je podmnožinou pravého jazyka anebo jsou si rovny). U jazyků, které se vám nepodaří s ničím spojit, stručně vysvětlete proč.

Poznámka: Zatímco v levém čtverci jsou konkrétní jazyky, v pravém jsou všechny jazyky odvozené od jazyka L, který není specifikován. Do pravého čtverce proto zaznačte inkluze, které platí obecně, tj. pro libovolnou volbu jazyka $L \subseteq \{a, b\}^*$. U jazyků z pravého čtverce, které nejsou s ničím spojeny, by proto mělo vysvětlení využívat protipříklady.



$$L_{11} = \{a, b\}^* \qquad L_{41} = L^R$$
$$L_{21} = \{a\} \qquad L_{51} = L$$
$$L_{31} = \{ab, abb\} \qquad L_{61} = (L^0 \cup \{a, b\})^+$$

$$L_{12} = \{a\}.\{b\}^+ \qquad L_{42} = L.\left(L^R\right)^*$$
$$L_{22} = \{a\}^+ \qquad L_{52} = L^6$$
$$L_{32} = \{b\}^+ \qquad L_{62} = L \cup L^0$$

$$L_{13} = \text{co}-\{a, b\}^* \qquad L_{43} = L \cup L.\left(L^R\right)^+$$
$$L_{23} = \{a\}^* \qquad L_{53} = L^*$$
$$L_{33} = \{a\}^+.\{b\}^+ \qquad L_{63} = \text{co}-L \cap L$$

Řešení: Relace inkluze je zakreslena v obrázku výše. Zdůvodnění s ničím nespojených jazyků jsou následující:

- L_{11} je jazyk všech slov nad abecedou $\{a, b\}^*$. Obsahuje tedy i např. slovo ba, které nepatří do žádného z jazyků L_{12}, L_{22} a L_{32}.
- L_{32} obsahuje např. slovo b, které nepatří do žádného z jazyků L_{13}, L_{23} a L_{33}. Zároveň do L_{32} nepatří ani slovo a, ani slovo ab, není tedy nadmnožinou L_{11}, L_{21} (kvůli a) ani L_{31} (kvůli ab).
- L_{13} je prázdný jazyk. Jazyky L_{12}, L_{22} a L_{32} jsou ale neprázdné, nemohou být proto jeho podmnožinou.
- L_{41} je zrcadlový obraz jazyka L. Zvolíme-li např. $L = \{ab\}$, pak $L_{41} = \{ba\}$. Slovo ba pak ovšem neobsahuje žádný z jazyků $L_{42} = \{ab\}.\{ba\}^*$, $L_{52} = \{abababababab\}$ a $L_{62} = \{\varepsilon, ab\}$.
- L_{61} je (pro libovolnou volbu jazyka L) jazyk všech slov. Stačí tedy zvolit např. $L = \emptyset$, pak zřejmě $L_{42} = \emptyset$, $L_{52} = \emptyset$, $L_{62} = \{\varepsilon\}$. Žádný z těchto jazyků zřejmě není nadmnožinou jazyka všech slov.
- L_{63} je (pro libovolnou volbu jazyka L) prázdný jazyk. Stačí za L zvolit libovolný neprázdný jazyk, pak zřejmě i jazyky L_{42}, L_{52} a L_{62} jsou neprázdné a nejsou tedy podmnožinami prázdného jazyka.

IB102 – úkol 2 – řešení **Odevzdání: 4. 10. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [**3 body**] Mějme jazyk $L = \{ab, ba, aba\}$ nad abecedou $\Sigma = \{a, b\}$. Zjistěte, kolik slov má následující jazyk:

$$\text{co}-\big((\text{co}-L)^2\big)$$

Odpověď zdůvodněte.

Řešení: Jazyk $\text{co}-((\text{co}-L)^2)$ má právě jedno slovo, a to aba. Ukážeme, že jazyk $(\text{co}-L)^2$ obsahuje všechna slova ze Σ^* kromě slova aba. Nechť tedy $w \in \Sigma^*$. Máme následující čtyři případy:

- w nepatří do L. Pak w patří do $\text{co}-L$. Protože rovněž ε patří do $\text{co}-L$, dohromady tedy $w = w \cdot \varepsilon \in (\text{co}-L)^2$.
- $w = ab$. Slova a a b nepatří do L, patří proto do $\text{co}-L$. Potom $w = a \cdot b \in (\text{co}-L)^2$.
- $w = ba$. Podobně jako v předchozím bodě $w = b \cdot a \in (\text{co}-L)^2$.
- $w = aba$. Není žádná možnost, jak rozdělit w na dvě slova tak, aby obě byla v $\text{co}-L$. To se snadno ověří, neboť možná rozdělení jsou pouze čtyři ($\varepsilon \cdot aba$, $a \cdot ba$, $ab \cdot a$, $aba \cdot \varepsilon$). Proto $w \notin (\text{co}-L)^2$.

Ukázali jsme tedy, že jazyk $(\text{co}-L)^2$ obsahuje všechna slova ze Σ^* kromě slova aba. Proto pro jeho doplněk platí $\text{co}-((\text{co}-L)^2) = \{aba\}$.

IB102 – úkol 3 – řešení **Odevzdání: 11. 10. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

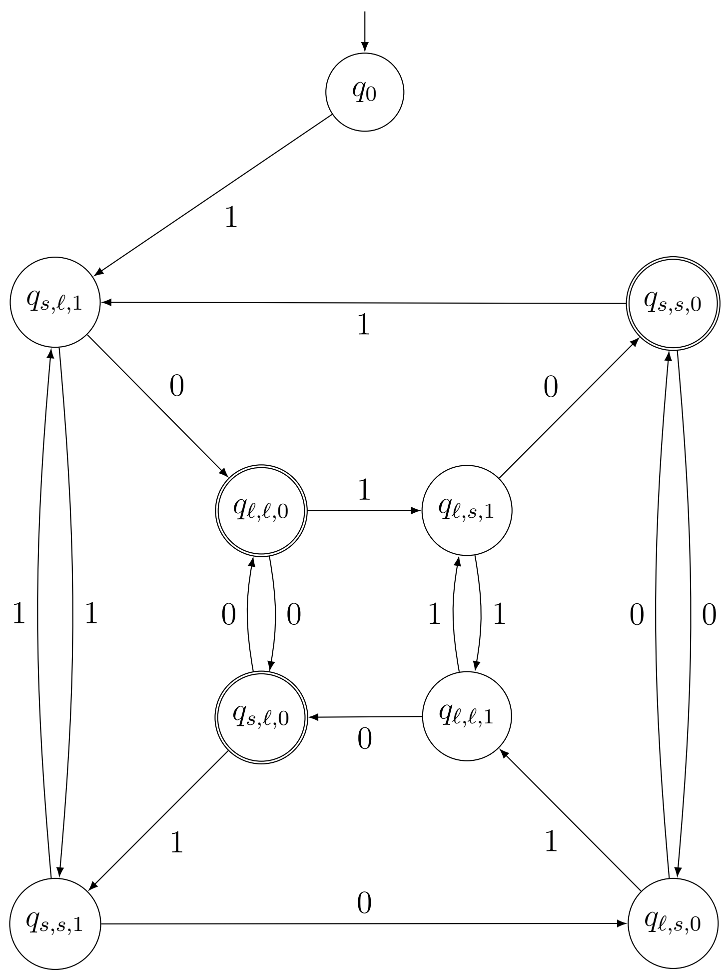
1. [**2 body**] Mějme následující jazyk:

$$L = \{w \in \{0,1\}^* \mid w \text{ je binární zápis sudého čísla a je-li } \#_1(w) \text{ sudý, pak je i } \#_0(w) \text{ sudý}\}$$

přičemž za binární zápis čísla považujeme pouze takový zápis, který neobsahuje zbytečné levostranné nuly, tj. 0110 pro nás není binární zápis čísla, zatímco 110 je.

Sestrojte deterministický konečný automat pro jazyk L.

Řešení: Deterministický konečný automat akceptující jazyk L může vypadat např. takto: Názvy stavů mají mnemotechnický význam, kromě q_0 jsou vždy tvaru $q_{x,y,z}$, kde x označuje paritu (sudost či lichost) počtu znaků 0, y paritu počtu znaků 1 a z označuje naposled čtený symbol.



IB102 – úkol 3 – řešení **Odevzdání: 11. 10. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [**2 body**] Mějme gramatiku $G = (\{S, A, B, C, D\}, \{a, b, c\}, P, S)$, kde

$$
\begin{aligned}
P = \{ \quad S &\rightarrow aA \mid bS \mid cS \mid abB, \\
A &\rightarrow abC \mid bA \mid cA, \\
B &\rightarrow aC \mid bB \mid cB, \\
C &\rightarrow aD \mid bC \mid cC \mid \varepsilon, \\
D &\rightarrow aD \mid bD \mid cD \quad \}
\end{aligned}
$$

Jaký jazyk generuje tato gramatika? Svou odpověď zdůvodněte.

Řešení: $L(G)$ je jazyk všech slov nad abecedou $\{a, b, c\}$, která obsahují podslovo ab a zároveň obsahují právě dva symboly a, množinově se dá zapsat takto

$$L(G) = \{b, c\}^* \cdot \{a\} \cdot \{b, c\}^* \cdot \{ab\} \cdot \{b, c\}^* \cup \{b, c\}^* \cdot \{ab\} \cdot \{b, c\}^* \cdot \{a\} \cdot \{b, c\}^*$$

K tomuto řešení je možno dospět následující úvahou:

- Neterminál D je zbytečný, není z něj možno odvodit žádné slovo.
- Z neterminálu C je proto možné odvodit právě všechna slova ze symbolů b a c, tedy právě slova z $\{b, c\}^*$.
- Z neterminálu B je proto možné odvodit právě slova, která obsahují právě jeden symbol a a libovolný počet symbolů b a c, tedy právě slova z $\{b, c\}^* \cdot \{a\} \cdot \{b, c\}^*$.
- Podobně z neterminálu A je možné odvodit právě slova z $\{b, c\}^* \cdot \{ab\} \cdot \{b, c\}^*$.
- Z předchozích dvou bodů a pravidel pro S pak zřejmě vyplývá řešení.

IB102 – úkol 4 – řešení **Odevzdání: 18. 10. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

1. [**2 body**] Nechť $L = \{w \in \{a,b\}^* \mid \#_a(w) \text{ je sudý} \Rightarrow \#_a(w) < \#_b(w)\}$. Rozhodněte, zda je jazyk L regulární a své tvrzení dokažte.
(K důkazu regularity jazyka stačí napsat příslušnou gramatiku nebo automat.)

Řešení: Jazyk L není regulární. To dokážeme použitím Pumping lemmatu.

- Mějme libovolné n, nadále pevné.
- Zvolíme si slovo $w \in L$ tak, že $|w| \geq n$:

$$w = b^{2n+1}a^{2n}$$

- Všechna možná rozdělení slova $w = xyz, |xy| \leq n, y \neq \varepsilon$ vypadají takto:

$$x = b^k \quad y = b^l \quad z = b^{2n+1-k-l}a^{2n} \quad (k \geq 0, l > 0, k + l \leq n)$$

- Zvolíme si $i = 0$. Potom platí:

$$xy^iz = b^k(b^l)^0b^{2n+1-k-l}a^{2n} = b^{2n+1-l}a^{2n}$$

  Zřejmě $b^{2n+1-l}a^{2n} \notin L$, protože počet symbolů a v tomto slově je sudý a počet b je menší nebo roven počtu a, díky tomu že $l > 0$. Podle PL tedy L není regulární. □

IB102 – úkol 4 – řešení **Odevzdání: 18. 10. 2010**

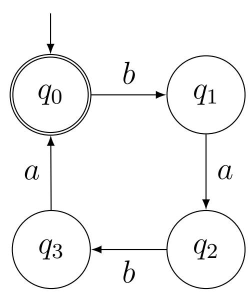
Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [**2 body**] Mějme gramatiku $G = (\{S, X, Y\}, \{a, b, c\}, P, S)$, kde

$$P = \{ \quad \begin{array}{rcl} S & \to & XY, \\ Y & \to & aYa \mid bYb \mid c, \\ Xab & \to & baX, \\ Xc & \to & \varepsilon \quad \} \end{array}$$

Popiše jazyk generovaný gramatikou G a určete, zda je tento jazyk regulární. Své tvrzení dokažte.

Řešení: Jazyk generovaný gramatikou G je $\{baba\}^*$. Tento jazyk je regulární, je pro něj např. možno sestrojit následující konečný automat.



IB102 – úkol 5 – řešení **Odevzdání: 25. 10. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

1. [2 body] Najděte jazyky L_1, L_2, L_3 a L_4 takové, aby byly splněny následující podmínky:

(a) Jazyky L_1, L_2 jsou různé a platí $\sim_{L_1} = \sim_{L_2}$.

(b) Jazyky L_3, L_4 jsou různé, platí $\sim_{L_3} \neq \sim_{L_4}$ a zároveň existuje relace $\sim$ splňující podmínky Nerodovy věty pro oba tyto jazyky.
(*Tj. $\sim$ musí být relace pravé kongruence s konečným indexem taková, že L_3 je sjednocením některých tříd rozkladu podle $\sim$ a zároveň L_4 je sjednocením některých tříd rozkladu podle $\sim$.*)

Své řešení zdůvodněte, tj. zejména popište všechny zmíněné relace, např. tak, že popíšete jejich třídy rozkladu.

Řešení může vypadat například takto:

(a) $L_1 = \emptyset$, $L_2 = \Sigma^*$. Pak zřejmě $\sim_{L_1} = \sim_{L_2} = \Sigma^* \times \Sigma^*$.
Poznámka: Podobně jsme mohli zvolit libovolný jiný jazyk spolu s jeho komplementem. Platí totiž, že $\sim_L = \sim_{\text{co}-L}$. Možných řešení však existuje ještě víc, například jazyky $L_1 = \{w \mid |w| \bmod 3 = 0\}$ a $L_2 = \{w \mid |w| \bmod 3 \neq 2\}$ sice mají neprázdný průnik, ale platí $\sim_{L_1} = \sim_{L_2}$.

(b) $L_3 = \Sigma^*$, $L_4 = \{\varepsilon\}$. Pak $\sim_{L_3} = \Sigma^* \times \Sigma^*$, $\sim_{L_4} = \{(\varepsilon, \varepsilon)\} \cup \Sigma^+ \times \Sigma^+$. Formulováno pomocí tříd rozkladu, $\sim_{L_3}$ má pouze jednu třídu obsahující všechna slova a $\sim_{L_4}$ má dvě třídy, jedna obsahuje pouze ε, druhá všechna ostatní slova. Dále vezmeme $\sim = \sim_{L_4}$. To je zřejmě pravá kongruence s konečným indexem (2) a L_3 i L_4 jsou sjednocením některých tříd rozkladu podle $\sim$.

IB102 – úkol 5 – řešení **Odevzdání: 25. 10. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [2 body] Mějme následující jazyk:

$$L = \{w \in \{0,1\}^* \mid w \text{ je binární zápis čísla } k \text{ takového, že } k \bmod 3 = 1\}$$

přičemž za binární zápis čísla považujeme pouze takový zápis, který neobsahuje zbytečné levostranné nuly, tj. 0110 pro nás není binární zápis čísla, zatímco 110 je.

(a) Určete index $\sim_L$ a popište třídy rozkladu podle $\sim_L$.

(b) Popište relaci pravé kongruence $\sim$ s konečným indexem takovou, že $\sim \neq \sim_L$ a přitom L je sjednocením některých tříd rozkladu podle $\sim$.

Řešení:

(a) Třídy rozkladu podle $\sim_L$ jsou následující:

1. $\{\varepsilon\}$
2. $\{0\} \cdot \{0,1\}^*$
3. $\{w \in \{1\} \cdot \{0,1\}^* \mid w \text{ je binárním zápisem čísla } k \text{ takového, že } k \bmod 3 = 0\}$
4. $\{w \in \{1\} \cdot \{0,1\}^* \mid w \text{ je binárním zápisem čísla } k \text{ takového, že } k \bmod 3 = 1\}$
5. $\{w \in \{1\} \cdot \{0,1\}^* \mid w \text{ je binárním zápisem čísla } k \text{ takového, že } k \bmod 3 = 2\}$

Index $\sim_L$ je tedy 5.

(b) Nechť $\sim$ je relace ekvivalence určená následujícími třídami rozkladu:

1. $\{\varepsilon\}$
2. $\{0\}$
3. $\{0\} \cdot \{0,1\}^+$
4. $\{w \in \{1\} \cdot \{0,1\}^* \mid w \text{ je binárním zápisem čísla } k \text{ takového, že } k \bmod 3 = 0\}$
5. $\{w \in \{1\} \cdot \{0,1\}^* \mid w \text{ je binárním zápisem čísla } k \text{ takového, že } k \bmod 3 = 1\}$
6. $\{w \in \{1\} \cdot \{0,1\}^* \mid w \text{ je binárním zápisem čísla } k \text{ takového, že } k \bmod 3 = 2\}$

Zřejmě platí, že $\sim$ je relací pravé kongruence s konečným indexem a $\sim \neq \sim_L$. L se rovná třídě rozkladu označené číslem 5.

IB102 – úkol 6 – řešení **Odevzdání: 1. 11. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

Bonusový příklad [3 body] Mějme binární operaci $\curlyvee$, kterou lze intuitivně popsat tak, že $u \curlyvee v$ je množina všech slov, která vzniknou tak, že se symboly z u promíchají se symboly z v s tím, že se pořadí symbolů z u a symbolů z v zachová. Například $aba \curlyvee cd = \{abacd, abcad, acbad, cabad, abcda, acbda, cabda, acdba, cadba, cdaba\}$. Formálně se tato operace na slovech dá definovat takto:

$$u \curlyvee v = \{u_1v_1u_2v_2 \cdots u_kv_k \mid k \in \mathbb{N}, u = u_1u_2 \cdots u_k, v = v_1v_2 \cdots v_k, \forall i : u_i, v_i \in \Sigma^*\}$$

Tato operace se dá rozšířit i na jazyky, intuitivně je $L_1 \curlyvee L_2$ jazyk, který vznikne tak, že se operace $\curlyvee$ aplikuje postupně na všechny dvojice slov z jazyků L_1 a L_2 a výsledné množiny se sjednotí. Například $\{ab, ba\} \curlyvee \{b, c\} = \{abb, bab, bba, abc, acb, cab, bac, bca, cba\}$. Formálně definujeme:

$$L_1 \curlyvee L_2 = \bigcup\{u \curlyvee v \mid u \in L_1, v \in L_2\}$$

Mějme dva nedeterministické konečné automaty A_1, A_2. Popište konstrukci automatu A takového, že $L(A) = L(A_1) \curlyvee L(A_2)$.

Řešení: Idea řešení spočívá v konstrukci „asynchronního“ paralelního spojení dvou automatů. Na rozdíl od klasického synchronního paralelního spojení se automaty nepohybují zároveň, ale naopak každý zvlášť. Vždy jeden „táhne“ a druhý „stojí“.

Nechť tedy $A_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ a $A_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$. Nový automat bude $A = (Q_1 \times Q_2, \Sigma, \delta, (q_1, q_2), F_1 \times F_2)$, kde:

$$\delta((p_1, p_2), a) = \delta_1(p_1, a) \times \{p_2\} \cup \{p_1\} \times \delta_2(p_2, a)$$

Důkaz korektnosti této konstrukce ponecháváme čtenáři jako cvičení. (V zadání jsme jej ani nepožadovali.)

Pro jednoduchost jsme uvažovali, že oba původní automaty mají stejnou vstupní abecedu. Řešení by se dalo zobecnit i na případ, kdy automat A_1 má vstupní abecedu Σ_1 a automat A_2 má vstupní abecedu Σ_2. Výsledný automat by pak měl vstupní abecedu $\Sigma_1 \cup \Sigma_2$.

Poznámka: V anglické literatuře se tato operace většinou nazývá „shuffle“ a označuje se symbolem $\diamond$. V zadání tohoto příkladu jsme schválně zvolili jiné značení, abychom zabránili možnosti dohledat řešení v literatuře, příp. na webu.

IB102 – úkol 7 – řešení **Odevzdání: 8. 11. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

1. [**2 body**] Rozhodněte, zda pro všechny jazyky L, K platí následující implikace. Svá rozhodnutí zdůvodněte.

(a) L^* je regulární $\implies$ L je regulární

(b) $(L \setminus K)^R$ je regulární, K je regulární a $K \subseteq L$ $\implies$ L je regulární

Řešení:

(a) Neplatí. Uvažme například jazyk $L = \{w \in \{a, b\}^* \mid w = w^R\}$. Jazyk obsahuje slova a i b a proto platí $L^* = \{a, b\}^*$, což je regulární jazyk. Jazyk L ale není regulární, což by se dalo dokázat například pomocí Pumping lemmatu s volbou slova $a^n bba^n$.

(b) Platí. Víme, že regulární jazyky jsou uzavřeny na reverzi. Dostáváme tedy, že $L \setminus K = ((L \setminus K)^R)^R$ je regulární. Dále z inkluze $K \subseteq L$ plyne, že $L = K \cup (L \setminus K)$. Jelikož regulární jazyky jsou uzavřené na sjednocení, jazyk L je také regulární.

IB102 – úkol 7 – řešení **Odevzdání: 8. 11. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [**2 body**] Definujme $\mathcal{T}$ jako třídu všech jazyků, jejichž prefixová ekvivalence má index nejvýše 4. Platí tedy, že jazyk L patří do třídy $\mathcal{T}$ právě tehdy, když *index* $\sim_L \leq 4$.

Odpovězte na následující otázky a své odpovědi zdůvodněte.

(a) Je třída $\mathcal{T}$ uzavřená na sjednocení?

(b) Je třída $\mathcal{T}$ uzavřená na průnik?

(c) **BONUS** [**+1 bod**] Je třída $\mathcal{T}$ uzavřená na iteraci?

(d) **BONUS** [**+1 bod**] Je třída $\mathcal{T}$ uzavřená na pozitivní iteraci?

Řešení: Třída $\mathcal{T}$ není uzavřená na sjednocení ani na průnik. K důkazu stačí vzít jazyky $L_1 = \{w \in \Sigma^* \mid |w| \bmod 3 = 0\}$ a $L_2 = \{w \in \Sigma^* \mid |w| \bmod 4 = 0\}$. Snadno se nahlédne, že *index* $\sim_{L_1} = 3$ a *index* $\sim_{L_2} = 4$, tedy L_1, $L_2 \in \mathcal{T}$. Jejich sjednocení a průnik jsou:

$$L_1 \cup L_2 = \{w \in \Sigma^* \mid |w| \bmod 3 = 0 \vee |w| \bmod 4 = 0\}$$
$$L_1 \cap L_2 = \{w \in \Sigma^* \mid |w| \bmod 3 = 0 \wedge |w| \bmod 4 = 0\}$$

Snadno se ověří (např. paralelním synchronním spojením automatů pro L_1 a L_2 a následnou minimalizací), že minimální DFA pro tyto jazyky mají 12 stavů, tedy že *index* $\sim_{L_1 \cup L_2} = 12$ a *index* $\sim_{L_1 \cap L_2} = 12$. Odtud tedy plyne, že $L_1 \cup L_2$, $L_1 \cap L_2 \notin \mathcal{T}$ a třída $\mathcal{T}$ není uzavřená na sjednocení ani na průnik.

Řešení bonusových otázek: Třída $\mathcal{T}$ není uzavřená na iteraci ani na pozitivní iteraci. Vezměme si následující jazyk:

$$L = \{w \in \Sigma^* \mid |w| \bmod 3 = 2\}$$

Zřejmě *index* $\sim_L = 3$. Iterace a pozitivní iterace tohoto jazyka jsou:

$$L^* = \{\varepsilon\} \cup \Sigma^2 \cup (\Sigma^4 \cdot \Sigma^*)$$
$$L^+ = \Sigma^2 \cup (\Sigma^4 \cdot \Sigma^*)$$

L^+ tedy obsahuje právě všechna slova délky 2 a délek od 4 výše, L^* obsahuje totéž, co L^+, a navíc ε. Snadno se ověří (opět např. sestrojením automatu a jeho minimalizací), že *index* $\sim_{L^*} = 5$ a rovněž *index* $\sim_{L^+} = 5$.

Poznámka: Všimněte si, že naše protipříklady fungují nezávisle na konkrétní abecedě Σ.

IB102 – úkol 8 – řešení **Odevzdání: 15. 11. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

---

1. [2 body] Navrhněte algoritmus, který pro zadanou regulární gramatiku rozhodne, zda tato gramatika generuje alespoň jedno slovo obsahující symbol a.

Řešení: Požadovaný algoritmus může vypadat např. takto:

vstup: regulární gramatika $G = (N, \Sigma, P, S)$

1. Nejprve se zbavíme zbytečných neterminálů, ze kterých se nedá odvodit žádné slovo. To uděláme tak, že si spočítáme množinu těch neterminálů, ze kterých se dá odvodit nějaké slovo. To se dá udělat např. tak, že postupně spočítáme množiny N_1, N_2, ... neterminálů, ze kterých se dá odvodit nějaké slovo v 1, 2, ... krocích. Skončíme ve chvíli, kdy N_i je stejná množina jako N_{i+1}. Výslednou množinu nazveme N'.
   $N_1 := \{A \in N \mid A \to x \in P, x \in \Sigma\}$
   $i := 0$
   repeat
   $i := i + 1$
   $N_{i+1} := N_i \cup \{A \in N \mid A \to xB \in P, x \in \Sigma, B \in N_i\}$
   until $N_i = N_{i+1}$
   $N' := N_i$

2. Nyní máme zaručeno, že ze všech neterminálů z N' je možno odvodit nějaké slovo. Dále si tedy spočítáme množinu neterminálů, z nichž je možno odvodit slovo obsahující a. To jsou všechny neterminály s pravidly tvaru $A \to a$, $A \to aB$, kde $B \in N'$ (těmto neterminálům A říkejme třeba „základní“) a všechny neterminály, z nichž se dá odvodit větná forma obsahující základní neterminál. Ty napočítáme obdobným způsobem jako v předchozím kroku. Výslednou množinu nazveme N''.
   $N'_1 := \{A \in N' \mid A \to a \in P\} \cup \{A \in N' \mid A \to aB \in P, B \in N'\}$
   $i := 0$
   repeat
   $i := i + 1$
   $N'_{i+1} := N'_i \cup \{A \in N' \mid A \to xB \in P, x \in \Sigma, B \in N'_i\}$
   until $N_i = N'_{i+1}$
   $N'' := N'_i$

3. Gramatika generuje slovo obsahující a právě tehdy, je-li počáteční neterminál S v množině N''.

výstup: odpověď **ANO**, pokud $S \in N''$, jinak odpověď **NE**

IB102 – úkol 9 – řešení **Odevzdání: 29. 11. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

1. [2 body] Mějme následující jazyk:

$$L = \{w \in \{a,b\}^* \mid w = w^R\} \cup \{w \in \{a,b\}^* \mid \#_a(w) \bmod 2 = \#_b(w) \bmod 2\}$$

Sestrojte *jednoznačnou* bezkontextovou gramatiku generující jazyk L. Stručně vysvětlete, proč je Vaše gramatika jednoznačná.

Řešení: Zřejmě umíme sestrojit jednoznačnou gramatiku pro $\{w \in \{a,b\}^* \mid w = w^R\}$ (jazyk všech palindromů) i pro $\{w \in \{a,b\}^* \mid \#_a(w) \bmod 2 = \#_b(w) \bmod 2\}$ (jazyk všech slov se stejnou paritou symbolů a a b). Sjednocení těchto jazyků ale není disjunktní. Budeme se tedy nejprve snažit napsat L jako disjunktní sjednocení dvou jazyků, pro které umíme sestrojit jednoznačné gramatiky.

Použijeme následující úvahu: Všechny palindromy sudé délky splňují podmínku stejné parity a a b, zatímco žádný palindrom liché délky tuto podmínku nesplňuje. Jazyk L můžeme tedy ekvivalentně napsat takto:

$$L = \{w \in \{a,b\}^* \mid w = w^R, |w| \text{ je lichá}\} \cup \{w \in \{a,b\}^* \mid \#_a(w) \bmod 2 = \#_b(w) \bmod 2\}$$

Hledaná gramatika je tedy např. $G = (\{S, X, Y_{00}, Y_{01}, Y_{10}, Y_{11}\}, \{a,b\}, P, S)$, kde

$$\begin{aligned} P = \{ \quad S &\rightarrow X \mid Y_{00}, \\ X &\rightarrow a \mid b \mid aXa \mid bXb, \\ Y_{00} &\rightarrow aY_{10} \mid bY_{01} \mid \varepsilon, \\ Y_{01} &\rightarrow aY_{11} \mid bY_{00}, \\ Y_{10} &\rightarrow aY_{00} \mid bY_{11}, \\ Y_{11} &\rightarrow aY_{01} \mid bY_{10} \mid \varepsilon \quad \}. \end{aligned}$$

(Zde neterminál X generuje právě všechny palindromy liché délky a neterminál Y_{00} generuje právě všechna slova se stejnou paritou a a b.)

IB102 – úkol 9 – řešení **Odevzdání: 29. 11. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [**3 body**] Mějme gramatiku $G = (\{S, A, B, C, D, E, F\}, \{a, b, c\}, P, S)$, kde

$$\begin{array}{rrcl} P = \{ & S & \rightarrow & AB \mid cC, \\ & A & \rightarrow & \varepsilon \mid aA \mid SEb, \\ & B & \rightarrow & \varepsilon \mid bB \mid AEc, \\ & C & \rightarrow & D \mid AcB, \\ & D & \rightarrow & Bb \mid Abc \mid C, \\ & E & \rightarrow & abE \mid Ec \mid EF, \\ & F & \rightarrow & Ea \mid ba \ \ \}. \end{array}$$

Ke gramatice G sestrojte (použitím algoritmů z přednášky) ekvivalentní gramatiku v Chomského normální formě.

Poznámka: Gramatika v CNF musí být vždy redukovaná.

Řešení: Nejprve z gramatiky odstraníme nepoužitelné symboly (tento krok bychom nemuseli v tuto chvíli dělat, je ale výhodné jej provést, protože se tím zbavíme zbytečných částí gramatiky a dále budeme pracovat s menší gramatikou).

- Normované neterminály jsou: $N_e = \{A, B, F, S, C, D\}$. (V prvním kroku algoritmu to jsou A, B a F, v druhém se přidají S, C a D, ve třetím kroku se nic nepřidá, proto skončíme.) Odstraníme tedy z gramatiky neterminál E a všechna pravidla jej obsahující.

$$\begin{array}{rcl} S & \rightarrow & AB \mid cC \\ A & \rightarrow & \varepsilon \mid aA \\ B & \rightarrow & \varepsilon \mid bB \\ C & \rightarrow & D \mid AcB \\ D & \rightarrow & Bb \mid Abc \mid C \\ F & \rightarrow & ba \end{array}$$

- Dosažitelné symboly jsou: $N' = \{S, A, B, C, D\}$ a $\Sigma' = \{c, a, b\}$. (V nultém kroku algoritmu to je S, v prvním kroku se přidají A, B, c a C, ve druhém kroku se přidají a, b a D, ve třetím kroku se nepřidá nic a proto skončíme.) Odstraníme tedy z gramatiky neterminál F.

$$\begin{array}{rcl} S & \rightarrow & AB \mid cC \\ A & \rightarrow & \varepsilon \mid aA \\ B & \rightarrow & \varepsilon \mid bB \\ C & \rightarrow & D \mid AcB \\ D & \rightarrow & Bb \mid Abc \mid C \end{array}$$

Nyní je třeba odstranit ε-pravidla. Spočítáme si proto nejprve $N_\varepsilon = \{A, B, S\}$ a postupujeme dále podle algoritmu z přednášky (S' je nyní nový počáteční neterminál):

$$\begin{array}{lcl}
S' & \to & S \mid \varepsilon \\
S & \to & AB \mid cC \mid A \mid B \\
A & \to & aA \mid a \\
B & \to & bB \mid b \\
C & \to & D \mid AcB \mid Ac \mid cB \mid c \\
D & \to & Bb \mid Abc \mid C \mid b \mid bc
\end{array}$$

Následně odstraníme jednoduchá pravidla. Nejprve si napočítáme množiny N_X pro všechny neterminály X:

$$\begin{array}{ccc}
N_{S'} = \{S', S, A, B\} & N_A = \{A\} & N_C = \{C, D\} \\
N_S = \{S, A, B\} & N_B = \{B\} & N_D = \{D, C\}
\end{array}$$

A opět pokračujeme dále dle algoritmu z přednášky:

$$\begin{array}{lcl}
S' & \to & \varepsilon \mid AB \mid cC \mid aA \mid a \mid bB \mid b \\
S & \to & AB \mid cC \mid aA \mid a \mid bB \mid b \\
A & \to & aA \mid a \\
B & \to & bB \mid b \\
C & \to & AcB \mid Ac \mid cB \mid c \mid Bb \mid Abc \mid b \mid bc \\
D & \to & AcB \mid Ac \mid cB \mid c \mid Bb \mid Abc \mid b \mid bc
\end{array}$$

Tím nám mohly vzniknout nedosažitelné neterminály (nenormované neterminály odstraněním jednoduchých pravidel vzniknout nemohou). Spočítáme si proto opět množinu dosažitelných neterminálů: $\{S', A, B, C\}$. Odstraníme z gramatiky neterminály S a D.

$$\begin{array}{lcl}
S' & \to & \varepsilon \mid AB \mid cC \mid aA \mid a \mid bB \mid b \\
A & \to & aA \mid a \\
B & \to & bB \mid b \\
C & \to & AcB \mid Ac \mid cB \mid c \mid Bb \mid Abc \mid b \mid bc
\end{array}$$

Nyní můžeme přejít k samotnému převodu na CNF.
Výsledná gramatika je $G' = (\{S', A, B, C, \langle cB\rangle, \langle bc\rangle, a', b', c'\}, \{a, b, c\}, P', S')$, kde

$$\begin{array}{llcl}
P' = \{ & S' & \to & \varepsilon \mid AB \mid c'C \mid a'A \mid a \mid b'B \mid b, \\
& A & \to & a'A \mid a, \\
& B & \to & b'B \mid b, \\
& C & \to & A\langle cB\rangle \mid Ac' \mid c'B \mid c \mid Bb' \mid A\langle bc\rangle \mid b \mid b'c', \\
& \langle cB\rangle & \to & c'B, \\
& \langle bc\rangle & \to & b'c', \\
& a' & \to & a, \\
& b' & \to & b, \\
& c' & \to & c \ \}.
\end{array}$$

IB102 – úkol 10 – řešení **Odevzdání: 6. 12. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

1. [**2 body**] Mějme gramatiku $G = (\{S, X, Y, Z\}, \{a, b, c\}, P, S)$, kde

$$
\begin{aligned}
P = \{ \quad S &\to XZX, \\
X &\to Xbc \mid Ybc, \\
Y &\to aa \mid Saa, \\
Z &\to ZZb \mid c \quad \}.
\end{aligned}
$$

Ke gramatice G sestrojte (použitím algoritmů z přednášky) ekvivalentní gramatiku v Greibachové normální formě.

Řešení: Gramatika je zřejmě redukovaná, bez jednoduchých a ε-pravidel. Začneme tedy rovnou odstraněním levé rekurze. Zvolíme si např. následující uspořádání:

$$S < X < Y < Z$$

Dále postupujeme podle algoritmu z přednášky.

1. Neterminál S neobsahuje přímou levou rekurzi, jeho pravidla tedy ponecháme beze změny.
2. Neterminál X neobsahuje pravidla, jejichž pravá strana by začínala S, ale obsahuje přímou levou rekurzi. Tu odstraníme:

$$
\begin{aligned}
X &\to Ybc \mid YbcX' \\
X' &\to bc \mid bcX'
\end{aligned}
$$

3. Neterminál Y obsahuje pravidlo $Y \to Saa$, to odstraníme a dostaneme:

$$Y \to aa \mid XZXaa$$

Dále odstraníme pravidlo $Y \to XZXaa$:

$$Y \to aa \mid YbcZXaa \mid YbcX'ZXaa$$

Nakonec odstraníme přímou levou rekurzi:

$$
\begin{aligned}
Y &\to aa \mid aaY' \\
Y' &\to bcZXaa \mid bcX'ZXaa \mid bcZXaaY' \mid bcX'ZXaaY'
\end{aligned}
$$

4. Neterminál Z obsahuje pouze přímou levou rekurzi, tu odstraníme:

$$
\begin{aligned}
Z &\to c \mid cZ' \\
Z' &\to Zb \mid ZbZ'
\end{aligned}
$$

Jako výsledek odstranění levé rekurze dostáváme gramatiku s těmito pravidly:

$$
\begin{array}{lcl}
S & \rightarrow & XZX \\
X & \rightarrow & Ybc \mid YbcX' \\
X' & \rightarrow & bc \mid bcX' \\
Y & \rightarrow & aa \mid aaY' \\
Y' & \rightarrow & bcZXaa \mid bcX'ZXaa \mid bcZXaaY' \mid bcX'ZXaaY' \\
Z & \rightarrow & c \mid cZ' \\
Z' & \rightarrow & Zb \mid ZbZ'
\end{array}
$$

Nyní můžeme přikročit k samotnému převodu na GNF. Uspořádání, požadované v algoritmu, je například toto:

$$Z' \prec Y' \prec X' \prec S \prec X \prec Y \prec Z$$

Dále pokračujeme v substitucích podle algoritmu (následující řádky jsou uspořádány podle toho, v jakém pořadí algoritmus zpracovává jednotlivé neterminály).

$$
\begin{array}{lcl}
Z & \rightarrow & c \mid cZ' \\
Y & \rightarrow & aa \mid aaY' \\
X & \rightarrow & aabc \mid aaY'bc \mid aabcX' \mid aaY'bcX' \\
S & \rightarrow & aabcZX \mid aaY'bcZX \mid aabcX'ZX \mid aaY'bcX'ZX \\
X' & \rightarrow & bc \mid bcX' \\
Y' & \rightarrow & bcZXaa \mid bcX'ZXaa \mid bcZXaaY' \mid bcX'ZXaaY' \\
Z' & \rightarrow & cb \mid cZ'b \mid cbZ' \mid cZ'bZ'
\end{array}
$$

Nakonec nahradíme terminály na jiných než počátečních pozicích příslušnými neterminály. Výsledná gramatika je tedy $G' = (\{S, X, X', Y, Y', Z, Z', a', b', c'\}, \{a, b, c\}, P', S)$, kde

$$
\begin{array}{rrcl}
P = \{ & S & \rightarrow & aa'b'c'ZX \mid aa'Y'b'c'ZX \mid aa'b'c'X'ZX \mid aa'Y'b'c'X'ZX, \\
& X & \rightarrow & aa'b'c' \mid aa'Y'b'c' \mid aa'b'c'X' \mid aa'Y'b'c'X', \\
& X' & \rightarrow & bc' \mid bc'X', \\
& Y & \rightarrow & aa' \mid aa'Y', \\
& Y' & \rightarrow & bc'ZXa'a' \mid bc'X'ZXa'a' \mid bc'ZXa'a'Y' \mid bc'X'ZXa'a'Y', \\
& Z & \rightarrow & c \mid cZ', \\
& Z' & \rightarrow & cb' \mid cZ'b' \mid cb'Z' \mid cZ'b'Z', \\
& a' & \rightarrow & a, \\
& b' & \rightarrow & b, \\
& c' & \rightarrow & c\ \ \}.
\end{array}
$$

IB102 – úkol 10 – řešení **Odevzdání: 6. 12. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [**3 body**] O následujících jazycích rozhodněte, zda jsou bezkontextové, a své rozhodnutí dokažte. (V případě, že jazyk je bezkontextový, nám bude jako důkaz stačit napsání příslušné bezkontextové gramatiky nebo zásobníkového automatu.)

(a) $L_1 = \{w \mid w \in \{a,b\}^+, \#_a(w) = \#_b(w)\}^2$

(b) $L_2 = \{w^2 \mid w \in \{a,b\}^+, \#_a(w) = \#_b(w)\}$

Řešení: Jazyk L_1 je bezkontextový, jazyk L_2 není bezkontexový.
Ad (a): Stačí si uvědomit, že se jazyk L_1 dá ekvivalentně napsat takto:

$$L_1 = \{u \cdot v \mid u, v \in \{a,b\}^+, \#_a(u) = \#_b(u), \#_a(v) = \#_b(v)\}$$

Jedná se tedy o jazyk, jehož slova jsou vždy zřetězením dvou neprázdných slov, z nichž každé má stejný počet a jako b. Tento jazyk generuje např. gramatika $G = (\{S,X\},\{a,b\},P,S)$, kde

$$\begin{aligned} P = \{ \quad S &\to XX, \\ X &\to XX \mid aXb \mid bXa \mid ab \mid ba \quad \}. \end{aligned}$$

Ad (b): Jazyk L_2 je jazyk všech slov, která se dají rozdělit na dvě *stejné* části, které mají stejný počet a jako b. (Jazyk L_2 se tedy od jazyka L_1 značně liší, např. $abba \in L_1$, zatímco $abba \notin L_2$.) Že je jazyk L_2 neregulární, dokážeme pomocí Pumping lemmatu pro bezkontextové jazyky.

- Nechť n je libovolné, nadále pevné.
- Zvolíme slovo $z \in L$, $|z| > n$ takto: $z = a^n b^n a^n b^n$.
- Všechna možná rozdělení $z = uvwxy$ taková, že $vx \neq \varepsilon$, $|vwx| \leq n$ jsou jednoho z těchto druhů:
  - $vwx \in \{a\}^+$. Volbou $i = 0$ pak slovo uv^0wx^0y je buď tvaru $a^m b^n a^n b^n$ nebo tvaru $a^n b^n a^m b^n$, kde $m < n$. Tato slova nepatří do L_2.
  - $vwx \in \{b\}^+$. Podobně jako v předchozím bodě volbou $i = 0$ bude slovo uv^0wx^0y buď tvaru $a^n b^m a^n b^n$ nebo tvaru $a^n b^n a^n b^m$, kde $m < n$. Tato slova opět nepatří do L_2.
  - $vwx \in \{a\}^+ \cdot \{b\}^+$. Pak buď $u \in \{a\}^*$ (vwx je v první polovině slova z) nebo $y \in \{b\}^*$ (vwx je v druhé polovině slova z), obě možnosti zároveň nastat nemohou. Zvolíme opět $i = 0$, v prvním případě pak bude slovo uv^0wx^0y tvaru $a^k b^l a^n b^n$, v druhém případě tvaru $a^n b^n a^k b^l$, kde v obou případech alespoň jedno z l, k je číslo menší než n. Žádné takové slovo nepatří do L_2.

- $vwx \in \{b\}^+ \cdot \{a\}^+$ (tj. vwx je na rozhraní dvou polovin slova z). Znovu volíme $i = 0$. Slovo uv^0wx^0y pak je určitě tvaru $a^nb^ra^sb^n$, kde alespoň jedno z r, s je menší než n, a tudíž nepatří do L_2.

- Jiná rozdělení nejsou. Ukázali jsme tedy, že pro každé rozdělení je možno najít konstantu i takovou, že $uv^iwx^iy \notin L_2$. Podle PL pro bezkontextové jazyky tedy L_2 není bezkontextový. $\square$

IB102 – úkol 11 – řešení **Odevzdání: 13. 12. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

1. [2 body] Mějme následující gramatiku:

$$G = (\{S', S, A, V, P, E, N, C, R\}, \{x, y, 1, 0, <, >, =, \mathbf{if}, \mathbf{then}, ;, +\}, P, S'), \text{ kde}$$

$$\begin{array}{rlll} P = \{ & S' & \to & S';S \mid S, \\ & S & \to & A \mid \mathbf{if}\ C\ \mathbf{then}\ S, \\ & A & \to & VPE, \\ & V & \to & x \mid y, \\ & P & \to & =, \\ & E & \to & V \mid N \mid E+E, \\ & N & \to & 0 \mid 1 \mid 0N \mid 1N, \\ & C & \to & ERE, \\ & R & \to & = \mid < \mid > \ \ \}. \end{array}$$

Pro gramatiku G sestrojte syntaktický analyzátor metodou **shora dolů**. Analyzujte slovo $\mathbf{if}\ x > 1\ \mathbf{then}\ y = 10$.

Řešení: Syntaktický analyzátor metodou shora dolů vypadá takto:

$$\begin{aligned} A_{td} = (&\{q\}, \{x, y, 1, 0, <, >, =, \mathbf{if}, \mathbf{then}, ;, +\}, \\ &\{x, y, 1, 0, <, >, =, \mathbf{if}, \mathbf{then}, ;, +, S', S, A, V, P, E, N, C, R\}, \delta, q, S', \emptyset), \text{ kde} \end{aligned}$$

$$\begin{array}{rr} \delta(q, \varepsilon, S') = \{(q, S';S), (q, S)\} & \delta(q, x, x) = \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, S) = \{(q, A), (q, \mathbf{if}\ C\ \mathbf{then}\ S)\} & \delta(q, y, y) = \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, A) = \{(q, VPE)\} & \delta(q, 1, 1) = \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, V) = \{(q, x), (q, y)\} & \delta(q, 0, 0) = \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, P) = \{(q, =)\} & \delta(q, <, <) = \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, E) = \{(q, V), (q, N), (q, E+E)\} & \delta(q, >, >) = \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, N) = \{(q, 0), (q, 1), (q, 0N), (q, 1N)\} & \delta(q, =, =) = \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, C) = \{(q, ERE)\} & \delta(q, \mathbf{if}, \mathbf{if}) = \{(q, \varepsilon)\} \\ \delta(q, \varepsilon, R) = \{(q, =), (q, <), (q, >)\} & \delta(q, \mathbf{then}, \mathbf{then}) = \{(q, \varepsilon)\} \\ \delta(q, ;, ;) = \{(q, \varepsilon)\} & \delta(q, +, +) = \{(q, \varepsilon)\} \end{array}$$

Syntaktickou analýzou slova $\mathbf{if}\ x > 1\ \mathbf{then}\ y = 10$ je pak následující výpočet:

$$\begin{aligned} (q, \mathbf{if}\ x > 1\ \mathbf{then}\ y = 10, S') &\vdash^{\varepsilon} (q, \mathbf{if}\ x > 1\ \mathbf{then}\ y = 10, S) \\ &\vdash^{\varepsilon} (q, \mathbf{if}\ x > 1\ \mathbf{then}\ y = 10, \mathbf{if}\ C\ \mathbf{then}\ S) \\ &\vdash^{\mathbf{if}} (q, x > 1\ \mathbf{then}\ y = 10, C\ \mathbf{then}\ S) \end{aligned}$$

$\xrightarrow{\varepsilon} (q, x > 1 \textbf{ then } y = 10, ERE \textbf{ then } S)$

$\xrightarrow{\varepsilon} (q, x > 1 \textbf{ then } y = 10, VRE \textbf{ then } S)$

$\xrightarrow{\varepsilon} (q, x > 1 \textbf{ then } y = 10, xRE \textbf{ then } S)$

$\xrightarrow{x} (q, > 1 \textbf{ then } y = 10, RE \textbf{ then } S)$

$\xrightarrow{\varepsilon} (q, > 1 \textbf{ then } y = 10, > E \textbf{ then } S)$

$\xrightarrow{>} (q, 1 \textbf{ then } y = 10, E \textbf{ then } S)$

$\xrightarrow{\varepsilon} (q, 1 \textbf{ then } y = 10, N \textbf{ then } S)$

$\xrightarrow{\varepsilon} (q, 1 \textbf{ then } y = 10, 1 \textbf{ then } S)$

$\xrightarrow{1} (q, \textbf{then } y = 10, \textbf{then } S)$

$\xrightarrow{\textbf{then}} (q, y = 10, S)$

$\xrightarrow{\varepsilon} (q, y = 10, A)$

$\xrightarrow{\varepsilon} (q, y = 10, VPE)$

$\xrightarrow{\varepsilon} (q, y = 10, yPE)$

$\xrightarrow{y} (q, = 10, PE)$

$\xrightarrow{\varepsilon} (q, = 10, = E)$

$\xrightarrow{=} (q, 10, E)$

$\xrightarrow{\varepsilon} (q, 10, N)$

$\xrightarrow{\varepsilon} (q, 10, 1N)$

$\xrightarrow{1} (q, 0, N)$

$\xrightarrow{\varepsilon} (q, 0, 0)$

$\xrightarrow{0} (q, \varepsilon, \varepsilon)$

IB102 – úkol 11 – řešení **Odevzdání: 13. 12. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

2. [**2 body**] Mějme následující gramatiku:

$$G = (\{S', S, A, V, P, E, N, C, R\}, \{x, y, 1, 0, <, >, =, \mathbf{if}, \mathbf{then}, ;, +\}, P, S'), \text{ kde}$$

$$\begin{array}{rlcl} P = \{ & S' & \to & S'; S \mid S, \\ & S & \to & A \mid \mathbf{if}\ C\ \mathbf{then}\ S, \\ & A & \to & VPE, \\ & V & \to & x \mid y, \\ & P & \to & =, \\ & E & \to & V \mid N \mid E + E, \\ & N & \to & 0 \mid 1 \mid 0N \mid 1N, \\ & C & \to & ERE, \\ & R & \to & = \mid < \mid > \ \ \}. \end{array}$$

Pro gramatiku G sestrojte syntaktický analyzátor metodou **zdola nahoru**. Analyzujte slovo **if** $x > 1$ **then** $y = 10$.

Řešení: Syntaktický analyzátor metodou zdola nahoru vypadá takto:

$$\begin{array}{l} A_{bu} = (\{q, r\}, \{x, y, 1, 0, <, >, =, \mathbf{if}, \mathbf{then}, ;, +\}, \\ \qquad \{x, y, 1, 0, <, >, =, \mathbf{if}, \mathbf{then}, ;, +, S', S, A, V, P, E, N, C, R, \bot\}, \delta, q, \bot, \{r\}), \text{ kde} \end{array}$$

$$\begin{array}{rr} \delta(q, x, \varepsilon) = \{(q, x)\} & \delta(q, \varepsilon, S'; S) = \{(q, S')\} \\ \delta(q, y, \varepsilon) = \{(q, y)\} & \delta(q, \varepsilon, S) = \{(q, S')\} \\ \delta(q, 1, \varepsilon) = \{(q, 1)\} & \delta(q, \varepsilon, A) = \{(q, S)\} \\ \delta(q, 0, \varepsilon) = \{(q, 0)\} & \delta(q, \varepsilon, \mathbf{if}\ C\ \mathbf{then}\ S) = \{(q, S)\} \\ \delta(q, <, \varepsilon) = \{(q, <)\} & \delta(q, \varepsilon, VPE) = \{(q, A)\} \\ \delta(q, >, \varepsilon) = \{(q, >)\} & \delta(q, \varepsilon, x) = \{(q, V)\} \\ \delta(q, =, \varepsilon) = \{(q, =)\} & \delta(q, \varepsilon, y) = \{(q, V)\} \\ \delta(q, \mathbf{if}, \varepsilon) = \{(q, \mathbf{if})\} & \delta(q, \varepsilon, =) = \{(q, P), (q, R)\} \quad (!) \\ \delta(q, \mathbf{then}, \varepsilon) = \{(q, \mathbf{then})\} & \delta(q, \varepsilon, V) = \{(q, E)\} \\ \delta(q, ;, \varepsilon) = \{(q, ;)\} & \delta(q, \varepsilon, N) = \{(q, E)\} \\ \delta(q, +, \varepsilon) = \{(q, +)\} & \delta(q, \varepsilon, E + E) = \{(q, E)\} \\ \delta(q, \varepsilon, 0) = \{(q, N)\} & \delta(q, \varepsilon, 1) = \{(q, N)\} \\ \delta(q, \varepsilon, 0N) = \{(q, N)\} & \delta(q, \varepsilon, 1N) = \{(q, N)\} \\ \delta(q, \varepsilon, ERE) = \{(q, C)\} & \delta(q, \varepsilon, <) = \{(q, R)\} \\ \delta(q, \varepsilon, >) = \{(q, R)\} & \delta(q, \varepsilon, \bot S') = \{(r, \varepsilon)\} \end{array}$$

Syntaktickou analýzou slova **if** $x > 1$ **then** $y = 10$ je pak následující výpočet:

$$(q, \mathbf{if}\ x > 1\ \mathbf{then}\ y = 10, \bot) \overset{\mathbf{if}}{\vdash} (q, x > 1\ \mathbf{then}\ y = 10, \bot\mathbf{if})$$

$$\overset{x}{\vdash} (q, > 1\ \mathbf{then}\ y = 10, \bot\mathbf{if}\ x)$$

$$\overset{\varepsilon}{\vdash} (q, > 1\ \mathbf{then}\ y = 10, \bot\mathbf{if}\ V)$$

$$\overset{\varepsilon}{\vdash} (q, > 1\ \mathbf{then}\ y = 10, \bot\mathbf{if}\ E)$$

$$\overset{>}{\vdash} (q, 1\ \mathbf{then}\ y = 10, \bot\mathbf{if}\ E >)$$

$$\overset{\varepsilon}{\vdash} (q, 1\ \mathbf{then}\ y = 10, \bot\mathbf{if}\ ER)$$

$$\overset{1}{\vdash} (q, \mathbf{then}\ y = 10, \bot\mathbf{if}\ ER1)$$

$$\overset{\varepsilon}{\vdash} (q, \mathbf{then}\ y = 10, \bot\mathbf{if}\ ERN)$$

$$\overset{\varepsilon}{\vdash} (q, \mathbf{then}\ y = 10, \bot\mathbf{if}\ ERE)$$

$$\overset{\varepsilon}{\vdash} (q, \mathbf{then}\ y = 10, \bot\mathbf{if}\ C)$$

$$\overset{\mathbf{then}}{\vdash} (q, y = 10, \bot\mathbf{if}\ C\ \mathbf{then})$$

$$\overset{y}{\vdash} (q, = 10, \bot\mathbf{if}\ C\ \mathbf{then}\ y)$$

$$\overset{\varepsilon}{\vdash} (q, = 10, \bot\mathbf{if}\ C\ \mathbf{then}\ V)$$

$$\overset{=}{\vdash} (q, 10, \bot\mathbf{if}\ C\ \mathbf{then}\ V =)$$

$$\overset{\varepsilon}{\vdash} (q, 10, \bot\mathbf{if}\ C\ \mathbf{then}\ VP)$$

$$\overset{1}{\vdash} (q, 0, \bot\mathbf{if}\ C\ \mathbf{then}\ VP1)$$

$$\overset{0}{\vdash} (q, \varepsilon, \bot\mathbf{if}\ C\ \mathbf{then}\ VP10)$$

$$\overset{\varepsilon}{\vdash} (q, \varepsilon, \bot\mathbf{if}\ C\ \mathbf{then}\ VP1N)$$

$$\overset{\varepsilon}{\vdash} (q, \varepsilon, \bot\mathbf{if}\ C\ \mathbf{then}\ VPN)$$

$$\overset{\varepsilon}{\vdash} (q, \varepsilon, \bot\mathbf{if}\ C\ \mathbf{then}\ VPE)$$

$$\overset{\varepsilon}{\vdash} (q, \varepsilon, \bot\mathbf{if}\ C\ \mathbf{then}\ A)$$

$$\overset{\varepsilon}{\vdash} (q, \varepsilon, \bot\mathbf{if}\ C\ \mathbf{then}\ S)$$

$$\overset{\varepsilon}{\vdash} (q, \varepsilon, \bot S)$$

$$\overset{\varepsilon}{\vdash} (q, \varepsilon, \bot S')$$

$$\overset{\varepsilon}{\vdash} (r, \varepsilon, \varepsilon)$$

IB102 – úkol 11 – řešení **Odevzdání: 13. 12. 2010**

Vypracoval: James Bond UČO: 007
Skupina: MI6

3. [2 body] Mějme následující jazyk nad abecedou $\{1, 2, 5, =\}$:

$$L = \{x = y \mid x \in \{1, 2\}^*, y \in \{5\}^*, \#_1(x) + 2 \cdot \#_2(x) = 5 \cdot \#_5(y)\}$$

Jedná se tedy o jazyk všech slov, která jsou tvaru $x = y$, kde x se skládá pouze ze znaků 1 a 2, y jen ze znaků 5 a ciferný součet x a y je stejný. (Všimněte si, že znak $=$ patří mezi znaky abecedy!)

Sestrojte zásobníkový automat akceptující jazyk L. Jasně uveďte, jakým způsobem Váš automat akceptuje (koncovým stavem, prázdným zásobníkem). (Motivace: jde o to, sestrojit automat, do nějž uživatel hází množství jedno- a dvoukorun, pak zmáčkne tlačítko $=$ a následně hází množství pětikorun. Automat má rozhodnout, jestli částky vhozené před a po zmáčknutí tlačítka $=$ byly stejné.)

BONUS [+2 body]: Sestrojte zásobníkový automat s jedním stavem, akceptující jazyk L. (Stačí vyřešit bonusovou variantu, neboť ta v sobě obsahuje i řešení příkladu jako takového. Napíšete-li tedy správný zásobníkový automat pro L s jedním stavem, získáte 4 body.)

Řešení: Idea konstrukce bude taková, že si automat bude počítat ciferný součet dosud přečtených 1 a 2 na zásobníku (pomocí jednoho pomocného symbolu X, jehož počet bude odpovídat cifernému součtu), po přečtení $=$ přejde do nového stavu, kde bude s každým dalším symbolem 5 odečítat pět symbolů X ze zásobníku (k tomu budeme potřebovat pomocné stavy). Automat bude akceptovat koncovým stavem. Hledaný zásobníkový automat se pak dá napsat například takto:

$$A = (\{q_x, q_y, q_4, q_3, q_2, q_1, q_F\}, \{1, 2, 5, =\}, \{X, Z\}, \delta, q_x, Z, \{q_F\}), \text{ kde}$$

$$\begin{aligned}
\delta(q_x, 1, Z) &= \{(q_x, XZ)\} & \delta(q_x, 1, X) &= \{(q_x, XX)\} \\
\delta(q_x, 2, Z) &= \{(q_x, XXZ)\} & \delta(q_x, 2, X) &= \{(q_x, XXX)\} \\
\delta(q_x, =, Z) &= \{(q_y, Z)\} & \delta(q_x, =, X) &= \{(q_y, X)\} \\
\delta(q_y, 5, X) &= \{(q_4, \varepsilon)\} & \delta(q_4, \varepsilon, X) &= \{(q_3, \varepsilon)\} \\
\delta(q_3, \varepsilon, X) &= \{(q_2, \varepsilon)\} & \delta(q_2, \varepsilon, X) &= \{(q_1, \varepsilon)\} \\
\delta(q_1, \varepsilon, X) &= \{(q_y, \varepsilon)\} & \delta(q_y, \varepsilon, Z) &= \{(q_F, \varepsilon)\}
\end{aligned}$$

Řešení bonusové varianty: První idea zásobníkového automatu pro jazyk L s jediným stavem bude taková, že si všechny informace budeme pamatovat na zásobníku. Zásobník bude opět sloužit jako počítadlo, ale místo jednoho symbolu použijeme pět symbolů 1, 2, 3, 4, 5 s tím, že všechny symboly na zásobníku kromě vrcholového budou 5. Při čtení symbolu $=$ pak dovolíme pokračovat pouze tehdy, je-li na vrcholu zásobníku symbol 5 (a tedy hodnota počítadla je dělitelná pěti). Následně budeme s každým čteným symbolem 5 jeden symbol 5 vybírat ze zásobníku.

Předchozí idea má ale drobný nedostatek – potřebujeme pro ni ve skutečnosti dva stavy (je třeba si pamatovat, jestli jsme před symbolem = nebo za ním). Toto se ale dá obejít, a k tomu opět využijeme zásobník: Budeme mít dvě varianty symbolu 5: symboly 5 a V. Symbol 5 se bude vyskytovat pouze na vrcholu zásobníku, a to pouze tehdy, nebyl-li dosud čten symbol =. Ve všech ostatních případech bude symbol 5 nahrazen symbolem V.

Automat bude samozřejmě akceptovat prázdným zásobníkem (máme-li jenom jeden stav, nemůžeme si dovolit akceptovat koncovým stavem).

Formálně se tedy dá takovýto automat zapsat takto:

$$A' = (\{q\}, \{1, 2, 5, =\}, \{1, 2, 3, 4, 5, V, Z\}, \delta, q, Z, \emptyset), \text{ kde}$$

$$\begin{array}{rr}
\delta(q, 1, Z) = \{(q, 1)\} & \delta(q, 2, Z) = \{(q, 2)\} \\
\delta(q, 1, 1) = \{(q, 2)\} & \delta(q, 2, 1) = \{(q, 3)\} \\
\delta(q, 1, 2) = \{(q, 3)\} & \delta(q, 2, 2) = \{(q, 4)\} \\
\delta(q, 1, 3) = \{(q, 4)\} & \delta(q, 2, 3) = \{(q, 5)\} \\
\delta(q, 1, 4) = \{(q, 5)\} & \delta(q, 2, 4) = \{(q, 1V)\} \\
\delta(q, 1, 5) = \{(q, 1V)\} & \delta(q, 2, 5) = \{(q, 2V)\} \\
\delta(q, =, Z) = \{(q, \varepsilon)\} & \delta(q, =, 5) = \{(q, V)\} \\
\delta(q, 5, V) = \{(q, \varepsilon)\} &
\end{array}$$