TIN - 2. domácí úkol

Kateřina Fortová (xforto00)

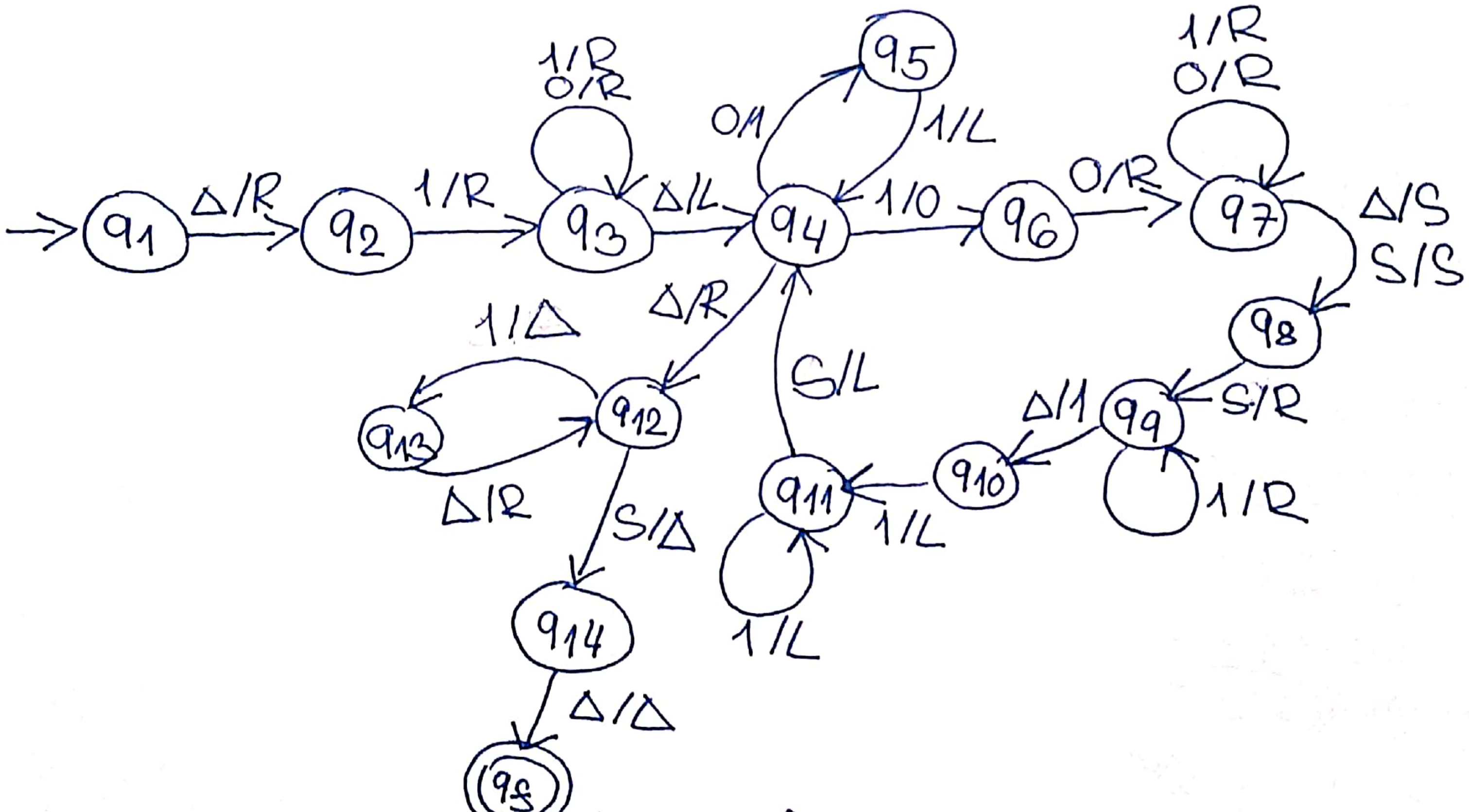
1. jednopáskový TS, $\underline{\Delta} x \Delta^{\omega}$, $x \in \{1\} \cdot \{0,1\}^*$ (vstup), výstup $\underline{\Delta} y \Delta^{\omega}$, $y \in \{1\}^+$, x je zakódování délky řetězce y v binární soustavě

Accept: např. vstup: $\underline{\Delta} 110 \Delta^{\omega}$ výstup: $\underline{\Delta} 111111 \Delta^{\omega}$

vstup: $\underline{\Delta} 11 \Delta^{\omega}$ výstup: $\underline{\Delta} 111 \Delta^{\omega}$

Reject: např. vstup: $\underline{\Delta} 0111 \Delta^{\omega}$

vstup: $\underline{\Delta} 00111 \Delta^{\omega}$



vysvětlení činnosti některých stavů:

$q_2 \xrightarrow{1/R} q_3$ - kontrola 1 na začátku vstupního řetězce

$q_7 \xrightarrow[S/S]{\Delta/S} q_8$ - vložení oddělovače S mezi vstup a finální výstup

$q_9 \xrightarrow{\Delta/1} q_{10}$ - zapsání 1 na místo Δ jako součást délky řetězce

dolní větev do q_f - vymazání vstupního řetězce, který je upraven na základě stavů $q_4 - q_7$, dále je nutné odstranit oddělovač S a tak tedy je přepsán celý upravený vstup a oddělovač pomocí blanků, přijmeme výstup ukázka nejdůležitějších kroků běhu pro vstup 11:

$\underline{\Delta}11\Delta\Delta^{\omega}$

$\vdots$

$\Delta 1\underline{0}\Delta\Delta^{\omega}$

$\Delta 10\underline{S}\Delta^{\omega}$

$\Delta 10S\underline{1}\Delta^{\omega}$

$\Delta\underline{0}1S1\Delta^{\omega}$

$\Delta 01S1\underline{1}\Delta^{\omega}$

$\Delta 00S11\underline{1}\Delta^{\omega}$

$\underline{\Delta}11S111\Delta^{\omega}$

$\Delta\underline{S}111\Delta^{\omega}$

$\underline{\Delta}111\Delta^{\omega}$ accept

ukázka nejdůležitějších kroků běhu pro vstup 10:

$\underline{\Delta}10\Delta\Delta^{\omega}$

$\Delta 1\underline{1}\Delta\Delta^{\omega}$

$\Delta\underline{0}1\Delta\Delta^{\omega}$

$\Delta 01\underline{S}\Delta^{\omega}$

$\Delta 01S\underline{1}\Delta^{\omega}$

$\Delta 0\underline{0}S1\Delta^{\omega}$

$\Delta 00S1\underline{1}\Delta^{\omega}$

$\underline{\Delta}11S11\Delta^{\omega}$

$\Delta\underline{S}11\Delta^{\omega}$

$\underline{\Delta}11\Delta^{\omega}$ accept

simulace pro 110 (uvedeném v zadání) by probíhala obdobně, zde však pro jednoduchost a menší počet kroků ukázány vstupy 10 a 11.

2. $L(M=A) = \{\langle M\rangle\#\langle A\rangle : M$ je TS, A je NKA, $L(M)=L(A)\}$

$\Sigma = \{0,1\}$

Důkaz:

Zkoumaný problém budeme reprezentovat jazykem výše zmíněným $L(M=A)$. Každý NKA může přijímat pouze jazyky ze třídy $\mathcal{L}_3$. Pokud má platit $L(M)=L(A)$, tak TS M musí také přijmout jazyk třídy $\mathcal{L}_3$.

Navrhneme redukci z CO-HP, tedy $\sigma: \{0,1,\#\}^* \to \{0,1\}^*$. Tato redukce přiřadí vstupu $x \in \{0,1,\#\}^*$ kód $\langle M_x\rangle\#\langle A_x\rangle$ (dvojice TS a NKA):

-> M_x je TS, který nemá fixně zvolený přijímaný jazyk.

A_x je NKA, pro který fixně zvolíme přijímaný jazyk. Tedy $L(A_x)=\emptyset$, $L(A_x) \in \mathcal{L}_3$ a je proto přijímáno NKA A_x.

1. M_x posoudí, zda $z \in \{0^n1^n \mid n \geq 0\}$ a výsledek bude zapamatován v konečném řízení.
2. M_x smaže vstup a zapíše sám x (řetězec pro který byl zkonstruovaný)
3. M_x ověří, zda x má správnou strukturu, pokud NE, tak je-li $z \in \{0^n1^n \mid n \geq 0\}$ přijme, jinak odmítne. (tedy pokud x je "garbage", je rozpoznán jazyk $\{0^n1^n \mid n \geq 0\}$)
4. Jinak bude $x = \langle M\rangle\#\langle w\rangle$ a M_x odsimuluje běh M na w. Pokud cyklí, tak cyklí (simulace nedoběhne).
5. Pokud simulace doběhne, tak $z \in \{0^n1^n \mid n \geq 0\}$ tak přijme, jinak odmítne.

CO-HP - NE - M na w zastaví $\to$ NE ($L(M_x) \neq L(A_x)$)
ANO - M na w nezastaví $\to$ ANO ($L(M_x) = L(A_x)$)

Lze snadno navrhnout TS M_σ implementující redukci σ.

Zkoumejme jazyk $L(M_x)$:

a) $L(M_x) = \{0^n1^n \mid n \geq 0\} \notin \mathcal{L}_3 \iff x$ nemá požadovanou strukturu $\langle M\rangle\#\langle w\rangle$ nebo M na w zastaví

b) $L(M_x) = \emptyset \in \mathcal{L}_3 \iff x$ má požadovanou strukturu $x = \langle M\rangle\#\langle w\rangle$ kde M na w nezastaví

Tedy $\forall x \in \{0,1,\#\}^*: \sigma(x) = \langle M_x\rangle\#\langle A_x\rangle \in L_{M=A} \iff L(M_x)=\emptyset \iff x=\langle M\rangle\#\langle w\rangle \wedge M$ na w nezastaví $\iff x \in$ CO-HP

Problém, zda $L(M=A)$ tedy není ani částečně rozhodnutelný. $\square$

Skenováno pomocí CamScanner

3. Operátor paralelní kompozice (shuffle)
např. $\{aa\} || \{bb\} = \{aabb, abab, abba, baab, baba, bbaa)$. Dokažte, že množina RE je uzavřena na $||$.

Důkaz na základě uzávěrových vlastností:
Operátor paralelní kompozice je definovaný dle zadání následovně:

$$||: \Sigma^* \times \Sigma^* \to 2^{\Sigma^*}$$

$w || \varepsilon = \varepsilon || w = \{w\}$ a $aw_1 || bw_2 = \{a\} \cdot (w_1 || bw_2) \cup \{b\}(aw_1 || w_2)$. Rozšíření na jazyky $L_1 || L_2 = \bigcup \{w_1 || w_2 : w_1 \in L_1, w_2 \in L_2\}$.
Základní operace, které shuffle využívá jsou tedy konkatenace a sjednocení. Dle uzávěrových vlastností víme, že třída RE je uzavřena vůči konkatenaci, sjednocení a průniku. Proto musí být RE uzavřený na $||$, protože jak induktivní definice shuffle, tak rozšíření operátoru na jazyky využívá pouze operace uzavřené na RE. □

4. Diagonalizace

množina Expr - všechny výrazy reprezentovaných řetězci generovaných z neterminálu E gramatikou s pravidly:

$$E \to 1 \mid 1+F \qquad \Sigma = \{0, 1, +\}$$
$$F \to 0 \mid 1 \mid F+F$$

příklady generovaných řetězců:

1+0+1+1 1+1+1+0+0
1+1 1
1+0+0 1+1+1

Dokažte že množina všech podmnožin (potenční množina) Expr je nespočetná (2^{Expr})

matice - řádky - spočetná množina - jazyky generované gramatikou uvedenou výše

- sloupce - generované řetězce z gramatiky uvedené výše

Předpokládejme, že potenční množina 2^{Expr} je spočetná. Pak dle definice spočetnosti existuje bijekce $f: \mathbb{N} \leftrightarrow 2^{Expr}$.

Uspořádejme řetězce generované gramatikou do nějaké posloupnosti $w_1, w_2, w_3, \dots$ např. $1, 1+0, 1+1, 1+0+0, 1+0+1, \dots$ pro $\Sigma = \{0, 1, +\}$. Nyní můžeme f zobrazit nekonečnou maticí.

$$\begin{array}{lccccc}
 & w_0 & w_1 & w_2 & \dots & w_i \dots \\
L_0 = f(0) & a_{00} & a_{01} & a_{02} & \dots & a_{0i} \\
L_1 = f(1) & a_{10} & a_{11} & a_{12} & \dots & a_{1i} \\
L_2 = f(2) & a_{20} & a_{21} & a_{22} & \dots & a_{2i} \\
\vdots & & & & &
\end{array}$$

, kde $a_{ij} = \begin{cases} 0, & \text{jestliže } w_j \notin L_i \\ 1, & \text{jestliže } w_j \in L_i \end{cases}$

Vezměme jazyk $\overline{L} = \{w_i \mid a_{ii} = 0\}$. $\overline{L}$ se liší od každého jazyka $L_i = f(i)$, $i \in \mathbb{N}$:

je-li $a_{ii} = 0$, pak w_i patří do jazyka generovaného gramatikou

je-li $a_{ii} = 1$, pak w_i nepatří do jazyka generovaného gramatikou

současně ale $\overline{L} \in 2^{Expr}$, f tudíž není surjektivní, což je spor.

Potenční množina Expr tedy je nespočetná. $\square$