Příjmení a jméno: Datum: 26. května 2021

Login studenta:

Zkoušková písemka z předmětu „Úvod do teoretické informatiky“

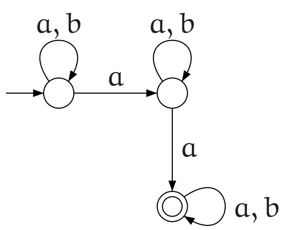
Doba trvání: **90 minut** Max. zisk: **70 bodů**

Část 1: Jazyky a automaty

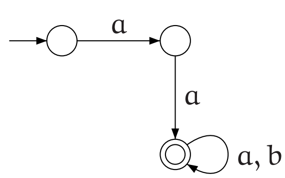
Příklad [1] (8 bodů):

Ke každému z následujících nedeterministických konečných automatů přiřaďte regulární výrazy (z níže uvedeného seznamu regulárních výrazů) popisující jazyk rozpoznávaný daným automatem. Řešení pište do připravené tabulky, kde ke každému automatu napište písmena odpovídajících regulárních výrazů, popřípadě buňku tabulky proškrtněte, pokud automatu neodpovídá žádný výraz. Každému automatu může odpovídat 0 až 6 výrazů, každý výraz může odpovídat 0 až 5 automatům.

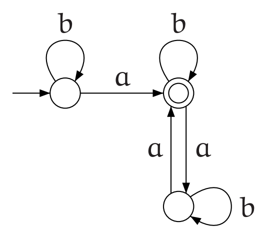
(1)



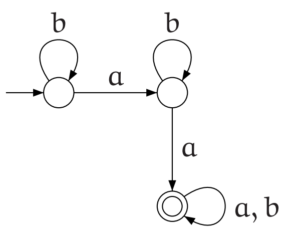
(2)



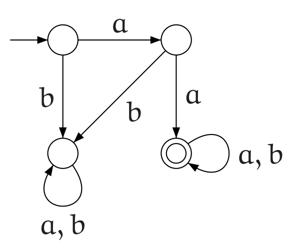
(3)



(4)



(5)



(A) $b^*ab^*(ab^*ab^*)^*$

(B) $b^*ab^*a(a+b)^*$

(C) $aa(b^*a^*)^*$

(D) $(ab^*ab^*)^*ab^*$

(E) $(ab^*a+b)^*ab^*$

(F) $b^*a(a+b)^*ab^*$

Řešení:

NKA	Reg. výrazy
(1)	
(2)	
(3)	
(4)	
(5)	

Příklad [2] (8 bodů): Navrhněte bezkontextovou gramatiku generující následující jazyk L.

$L = L_1 \cdot (L_2)^*$, kde
$L_1 = \{w \in \{a, b\}^* \mid w$ končí sufixem bba nebo $aab\}$ a
$L_2 = \{\, b^n a^m \mid 1 < n \leq m \,\}$.

Příklad [3] (7 bodů): Vezměme si tyto jazyky:

- $L_1 = \mathcal{L}(\mathcal{G})$, kde $\mathcal{G}$ je gramatika: $S \to aS \mid Sa \mid b$
- $L_2 = \{w = a^i b^j a^i \mid i, j \geq 0\}$

Pro každý z jazyků uvedených v bodech a) až d) vypište prvních 5 slov z daného jazyka (tj. uveďte prvních 5 slov v uspořádání, kdy slova řadíme nejprve podle délky a v rámci stejné délky pak lexikograficky, např. prvních 5 slov z jazyka nad abecedou $\{a, b\}$ popsaného regulárním výrazem $(a + b)^*$ jsou slova: ε, a, b, aa, ab). Pokud jazyk má **méně než 5 slov**, uveďte je všechna spolu s poznámkou, že více jich není.

a) $L_1 \cap L_2$

b) $L_2 - L_1$

c) $L_1 - L_2$

d) $\overline{L_2}$

Příklad [4] (6 bodů): Uvažujme následující jazyky ($|w|$ označuje délku slova w, $|w|_a$ označuje počet výskytů symbolu a ve w; w^R značí zrcadlový obraz slova w):

$L_1 = \{w \in \{a\}^* \mid |w| \bmod 3 = 2\}$

$L_2 = \mathcal{L}(\mathcal{G})$, kde $\mathcal{G}$ je bezkontextová gramatika $S \to aSa \mid b$

$L_3 = \{w \in \{a, b, c\}^* \mid |w|_a = |w|_c\}$

$L_4 = \{a^i b^i c^i \mid i \leq 3\}$

$L_5 = \{wcccw \mid w \in \{a, b\}^*\}$

$L_6 = \{w \in \{a, b\}^* \mid w$ obsahuje podslovo aaa nebo podslovo $bbb\}$

Do následující tabulky napište, jaké jsou odpovědi na otázky v řádcích pro jednotlivé jazyky určené sloupcem (místo „ano“ a „ne“ stačí psát „`A`“ a „`N`“).

	L_1	L_2	L_3	L_4	L_5	L_6
Je regulární?						
Je bezkontextový?						

Příklad [5] (6 bodů): Přesně popište, jak lze obecně k libovolné bezkontextové gramatice $\mathcal{G}$ sestrojit (nedeterministický) zásobníkový automat $\mathcal{M}$ takový, že

$$\mathcal{L}(\mathcal{M}) = \mathcal{L}(\mathcal{G}) .$$

Část 2: Vyčíslitelnost a složitost

Příklad [6] (8 bodů): Pro následující algoritmus určete co nejpřesněji jeho časovou složitost v nejhorším případě a vyjádřete ji pomocí asymptotické notace (nejlépe jako jeden odhad vyjádřený pomocí Θ, případně jako dva odhady vyjádřené pomocí O a Ω). Jako velikost vstupu uvažujte hodnotu uloženou v proměnné n.

Odvozené výsledky **zdůvodněte**.

Poznámka: Předpokládejte, že $n \geq 4$, že pole A a B jsou tvořená n prvky a indexována od nuly, a že prvky těchto polí jsou čísla v plovoucí řádové čárce (floating-point).

```
COMPUTE (A, B, n):
    B[0] := (A[0] + A[1]) / 2
    B[1] := (A[0] + A[1] + A[2]) / 3
    for i := 2 to n − 3 do
        x := 0
        for j := i − 2 to i + 2 do
            x := x + A[j]
        B[i] := x / 5
    B[n − 2] := (A[n − 3] + A[n − 2] + A[n − 1]) / 2
    B[n − 1] := (A[n − 2] + A[n − 1]) / 2
```

Příklad [7] (8 bodů):

$$
\begin{array}{ll}
 & R_2 := \textsc{read}\,() \\
L_1: & R_1 := \textsc{read}\,() \\
 & \textbf{if}\ (R_1 \leq 0)\ \textbf{goto}\ L_2 \\
 & R_0 := R_1 - R_2 \\
 & \textbf{if}\ (R_0 \geq 0)\ \textbf{goto}\ L_1 \\
 & R_2 := R_1 \\
 & \textbf{goto}\ L_1 \\
L_2: & \textsc{write}\,(R_2) \\
 & \textbf{halt}
\end{array}
$$

a) Do tabulky uveďte, kolik kroků (tj. provedení instrukcí) vykoná uvedený stroj RAM, pokud dostane jako vstup sekvenci

$$9, 14, 7, 8, 8, 4, 1, 6, 0$$

a jaký výstup vydá pro tento vstup:

počet kroků	výstup

b) Do následující tabulky vypište obsah paměti uvedeného RAMu po provedení 10 kroků a obsah paměti po skončení výpočtu (byla-li vstupem výše uvedená sekvence).

adresa paměti	R_0	R_1	R_2	R_3	R_4	R_5
obsah po 10 krocích						
obsah po skončení						

c) Uveďte kolik kroků maximálně provede výše uvedený stroj RAM, pokud dostane jako vstup posloupnost n čísel, z nichž poslední je číslo 0 (určete tento počet kroků co nejpřesněji, ne jen asymptotický odhad):

d) Uveďte, co bude obecně pro takovou libovolnou posloupnost na vstupu výstupem:

Příklad [8] (5 bodů): V následující tabulce vyznačte, které ze vztahů typu $f \in O(g)$, $f \in \Omega(g)$, $f \in \Theta(g)$, $f \in o(g)$ a $f \in \omega(g)$ platí, a které ne.

Vztahy, které platí, **zakroužkujte**, a vztahy, které neplatí, **přeškrtněte**.

$f_1(n) = (0.0001n)^3$ $\qquad$ $f_2(n) = (n-1)(n-1)(n-1)$ $\qquad$ $f_3(n) = 2^{12} \cdot n^2 \cdot \sqrt{n}$

$f_1 \in O(f_2)$	$f_2 \in O(f_1)$	$f_1 \in O(f_3)$	$f_3 \in O(f_1)$	$f_2 \in O(f_3)$	$f_3 \in O(f_2)$
$f_1 \in \Omega(f_2)$	$f_2 \in \Omega(f_1)$	$f_1 \in \Omega(f_3)$	$f_3 \in \Omega(f_1)$	$f_2 \in \Omega(f_3)$	$f_3 \in \Omega(f_2)$
$f_1 \in \Theta(f_2)$	$f_2 \in \Theta(f_1)$	$f_1 \in \Theta(f_3)$	$f_3 \in \Theta(f_1)$	$f_2 \in \Theta(f_3)$	$f_3 \in \Theta(f_2)$
$f_1 \in o(f_2)$	$f_2 \in o(f_1)$	$f_1 \in o(f_3)$	$f_3 \in o(f_1)$	$f_2 \in o(f_3)$	$f_3 \in o(f_2)$
$f_1 \in \omega(f_2)$	$f_2 \in \omega(f_1)$	$f_1 \in \omega(f_3)$	$f_3 \in \omega(f_1)$	$f_2 \in \omega(f_3)$	$f_3 \in \omega(f_2)$

Příklad [9] (8 bodů): Navrhněte a podrobně popište (nejlépe pomocí pseudokódu) algoritmus řešící následující problém. V případech, kdy není očividné, že vámi navržený algoritmus skutečně řeší daný problém, uveďte nějaké alespoň neformální zdůvodnění korektnosti tohoto algoritmu.

Uveďte, jakou má tento algoritmus časovou a prostorovou složitost. Vaše odhady složitosti **zdůvodněte**.

VSTUP: Slovo w nad abecedou $\{\mathtt{a}, \mathtt{b}\}$.

VÝSTUP: Sekvence $v_1, v_2, \ldots, v_m$ tvořená všemi podslovy slova w.

Poznámka: Na konkrétním pořadí slov na výstupu nezáleží. Slova na výstupu se mohou i opakovat, ale žádné nesmí chybět.

Například pro slovo abb může být výstupem posloupnost ε, a, b, ab, bb, abb.

Příklad [10] (6 bodů): Vezměme si libovolné dvě funkce f a g typu $\mathbb{N} \to \mathbb{N}$. Podrobně a přesně definujte, kdy platí $f \in \Omega(g)$.

Poté na základě výše uvedené definice podrobně zdůvodněte, proč platí

$$0.1n^3 - 10n - 725 \in \Omega(n^2)\,.$$