Prova Substitutiva de Teoria da Computação — 2017/2
Prof. José de Oliveira Guimarães — DComp — Campus de Sorocaba da UFSCar

Ao final da prova, entregue APENAS a folha de respostas. A folha de questões não será considerada. Utilize quantas folhas de reposta forem necessárias. Não é necessário entregar as questões em ordem. Utilize lápis ou caneta, mas faça letra legível. Na correção, símbolos ou palavras ilegíveis não serão considerados.

Justifique todas as respostas.

Se você estudou com algum colega para esta prova, não se sente ao lado dele pois é possível que acidentalmente vocês produzam respostas semelhantes para alguma questão. Provas de alunos que fizeram a prova próximos uns dos outros com respostas semelhantes caracterizam cópia de questões. Lembro-os de que todos os envolvidos na utilização de métodos ilegais na realização desta prova receberão zero de nota final da disciplina (e não apenas nesta prova).

Coloque o seu nome na folha de resposta, o mais acima possível na folha, seguido do número da sua coluna de carteiras. A primeira carteira é a mais perto da porta e a última a mais perto das janelas. Não precisa colocar o RA.

A menos de menção em contrário, usaremos o alfabeto $\Sigma = \{0, 1\}$ nesta prova.

1. (1,5) Faça um autômato finito que reconheça a linguagem $A = \{ 0^n100^m \mid n \geqslant 0, m \geqslant 0 \}$.

2. (0,5) Cite os três tipos de autômatos vistos na disciplina capazes de reconhecer linguagens regulares.

3. (1,0) Descreva em formato de conjuntos uma linguagem que não é regular mas é livre de contexto. Faça a gramática que gera esta linguagem

4. (0,5) Cite o domínio e o contra-domínio da função δ de transição de um autômato com pilha $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$. Use A_ϵ para $A \cup \{\epsilon\}$.

5. (1,5) Prove que a gramática abaixo é ambígua.

$E \longrightarrow E + E$
$E \longrightarrow E * E$
$E \longrightarrow 0|1$

6. (1,5) Cite a Tese de Church-Turing. Ela pode ser usada como uma definição ou como uma afirmação de que "tal coisa pode ser 'feita' por tal coisa". Assumindo que ela é uma afirmação, como ela poderia ser falsificada? Isto é, como se poderia provar que ela está errada?

7. (1,5) Explique como funciona uma MTND. Na sua resposta, cite a diferença entre o conjunto de instruçõe de uma MTD e uma MTND, faça uma comparaçaõ de quem é mais poderosa (MTND ou MTD ou equivalentes), cite um exemplo de uma MTND (bem simples).

8. (2,0) L e K são linguagens decidíveis. Prove que:

(a) (1,0) $L \cap K$ é decidível;

(b) (1,0) existe uma MT M que enumera L. A prova deve ser pela especificação de M, não pode ser por meios indiretos.

Resumo:

Uma expressão regular sobre um alfabeto Σ é descrita como: a) x é uma e.r. (expressão regular) se $x \in \Sigma$; b) ϵ é uma e.r. c) $\emptyset$ é uma e.r. d) $x?$, $(x \cup y)$, (xy) (concatenação de x e y) e $(x^\star)$ são e.r. se x, y são e.r. Assume-se que a ordem de precedência dos operadores seja, do maior para o menor: $\star$, ?, concatenação e união ($\cup$). Parenteses podem ser removidos se forem redundantes. A concatenação de duas linguagens L e K é $L \circ K = \{vw : v \in L \text{ e } w \in K\}$. E $L^\star = \{w_1 w_2 \ldots w_n | n \geqslant 0\}$.

O lema do bombeamento para linguagens regulares: se A é uma linguagem regular, então existe um inteiro p dependente de A tal que, se $s \in A$ e $|s| \geq p$, então s pode ser dividida em três partes, $s = xyz$, satisfazendo (a) $xy^iz \in A$ para todo $i \geq 0$; (b) $\left|y\right| > 0$ e (c) $\left|xy\right| \leq p$.

Uma máquina de Turing determinística (MT ou MTD) é uma quadrupla (Q, Σ, I, q) na qual Q e Σ são conjuntos chamados de conjuntos de estados e de símbolos, I é um conjunto de instruções, $I \subset Q \times \Sigma \times Q \times \Sigma \times D$, $D = \{-1, 0, 1\}$ e $q \in Q$ é chamado de estado inicial. Há dois estados especiais: q_s e q_n, todos elementos de Q e todos diferentes entre si. Neste texto convenciona-se que o estado inicial será q_0 a menos de menção em contrário. Exige-se que $\{0, 1, \rhd, \sqcup, \square\} \subset \Sigma$. Uma instrução é da forma (q_i, s_j, q_l, s_k, d) na qual $s_k \neq \square$ e $q_i \notin \{q_s, q_n\}$. Se $(q, s, q'_0, s'_0, d_0), (q, s, q'_1, s'_1, d_1) \in I$, então $q'_0 = q'_1$, $s'_0 = s'_1$ e $d_0 = d_1$. Q, Σ e I são conjuntos finitos.

O símbolo $\square$ é o branco utilizado para as células ainda não utilizadas durante a execução e $\sqcup$ é utilizado para separar dados de entrada e saída.

Símbolos: Máquina de Turing Não Determinística (MTND), Máquina de Turing (MT), Máquina de Turing Determinística (MTD). A menos de menção em contrário, uma MT é uma MTD. Todas as linguagens utilizadas são subconjuntos de $\{0, 1\}^\star$.

Uma linguagem L é computável (recursiva) se existe uma MT M de decisão tal que

$$x \in L \text{ sse } M(x) = 1$$

Isto é, M decide L.

Uma linguagem L é ou computacionalmente enumerável, c.e. (recursivamente enumerável, r.e.) se existe uma MT M tal que se $x \in L$ então $M(x) = 1$. Se $x \notin L$, $M(x) \uparrow$. Dizemos que M *aceita* L.

Dizemos que uma MT M enumera os elementos de uma linguagem $L \subset \Sigma^\star$ (ou $A \subset \mathbb{N}$) não vazia se:

(a) M despresa a sua entrada e imprime na fita um elemento de L, seguido de $\sqcup$, seguido de outro elemento de L, seguido de $\sqcup$ e assim por diante;

(b) dado $x \in L$, em algum momento da execução de M x será impresso na fita;

(c) M nunca pára a sua execução.

Note que os elementos impressos na fita por M podem ser repetidos e estar fora de ordem. Em particular, se L (ou A) for finito, haverá repetições.