Segunda Prova 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.

1. (2,0) Considere as duas variações de MT dadas abaixo. Para cada uma delas, responda **SIM** se a variação tem o mesmo poder que as MT e **NÃO** se tem poder menor do que as máquinas de Turing convencionais. Após responder a esta pergunta (apenas **SIM** ou **NÃO**), justifique. A justificativa pode ser informal, mas deve ser rigorosa.

(a) (1,0) a fita é infinita apenas à direita. No início da execução da máquina a cabeça de leitura e gravação está na célula mais à esquerda da fita. Durante a execução, a cabeça de leitura e gravação não pode ir para a esquerda. Ou ela permanece na mesma célula ou vai para a direita;

(b) (1,0) o alfabeto é exatamente o conjunto $\{1, \triangleright, \sqcup, \square\}$. Então zero (0) não pode aparecer na entrada e não pode ser escrito. Os símbolos $\triangleright$, $\sqcup$ e $\square$ desempenham papel igual nesta variação e nas MT convencionais.

2. (2,5) Faça o diagrama de uma MT que prova que a linguagem $\{ w : |w| \text{ é par } \}$ é computacionalmente enumerável. Alternativamente, faça o diagrama de uma MT que prova que esta linguagem é computável (valerá 2,0 pontos). O professor dará um exemplo de diagrama no quadro verde.

3. (3,0) Sobre codificação, responda:

(a) (1,0) o que é uma codificação de uma MT?

(b) (1,0) cite uma aplicação de codificação. Dê detalhes de como a codificação é usada;

(c) (1,0) codifique a instrução $(q_1, 0, q_2, 1, 0)$ de uma MT tal que $Q = \{q_0, q_1, \ldots q_{15}\}$, $\Sigma = \{0, 1, \triangleright, \sqcup, \square\}$. Você pode inventar a sua codificação, mas deixe claro os passos que você seguiu.

Faça UMA e apenas UMA das duas questões abaixo

4. (2,5) Prove que, se as linguagens K e K^c são computacionalmente enumeráveis, K é computável.

5. (2,5) Prove que existe uma MT R_M dependente de M que enumera a linguagem $K = \{x | M(x) \downarrow\}$. Note que M é fixado para R_M. Veja a definição de "máquina enumera uma linguagem" no resumo ao final da prova.

Resumo:

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.