TEORIA DA COMPUTAÇÃO

Prova 2 - 27/03/2017

Prof. Marcus Ramos

1ª Questão - (1,0 ponto) Descreva as linhas gerais da prova de que Máquinas de Turing sem limitação de fita podem ser simuladas por Máquinas de Turing com fita limitada à esquerda.

A simulação de uma MT com fita ilimitada é feita numa MT' com fita limitada à esquerda e duas trilhas. Na trilha superior representamos a parte da fita de MT que se encontra à direita da cabeça de leitura e escrita, considerada a sua configuração inicial. Na trilha inferior representamos a parte da fita de MT que se encontra à esquerda. Os estados de MT´são os mesmos de MT, porém considerando-se a trilha que está selecionada no momento. Movimentos para a direita em MT' são feitos à direita na trilha superior ou à esquerda na trilha inferior, dependendo da posição da cabeça de leitura e escrita. Da mesma forma, movimentos para a esquerda em MT' são feitos à esquerda na trilha superior ou à direita na trilha inferior em MT. Um cuidado especial deve ser tomado na transição da trilha superior para a trilha inferior e vice-versa, a fim de garantir que a trilha correta esteja selecionada. Dessa forma, a cabeça de leitura e escrita nunca se movimenta à esquerda da posição mais à esquerda.

2ª Questão - (1,0 ponto) Em que consiste a Teoria da Computabilidade, e qual a sua importância prática para o profissional de computação?

A Teoria da Computabilidade investiga a existência de algoritmos para problemas diversos. Como resultado, ela procura determinar os problemas que possuem solução (problemas decidíveis) e aqueles que não possuem solução (indecidíveis). A sua importância prática reside no fato de o profissional poder poupar tempo e esforço se ele puder determinar, com antecedência, se um determinado problema possui ou não solução.

3ª Questão - (1,0 ponto) Seja P um probema de decisão. De que maneira é feita a representação de P na forma de uma linguagem?

Deve-se, inicialmente, codificar todas as instâncias do problema sobre um alfabeto pré-estabelecido. Depois, reune-se todas as instâncias cuja resposta seja afirmativa e temos a linguagem que representa P.

4ª Questão - (1,0 ponto) Considere as linguagens $L_1$, $L_2$, e $L_3$, respectivamente recursiva, recursivamente enumeráveis não-recursivas e não-recursivamente enumeráveis. Quais os tipos (mais restritos) de complemento de $L_1$, complemento de $L_2$ e complemento de $L_3$?

Conforme provado em sala de aula, o complemento de uma linguagem recursiva é sempre uma linguagem recursiva e o complemento de uma linguagem recursivamente enumerável não-recursiva é sempre uma linguagem não-recursivamente enumerável. Já o complemento de uma linguagem não-recursivamente enumerável pode tanto ser uma linguagem recursivamente enumerável não-recursiva quanto uma linguagem também não-recursivamente enumerável.

5ª Questão - (1,0 ponto) Prove que o problema $P_1 = \{<M_1,M_2> \mid M_1$ e $M_2$ são Máquinas de Turing e o número de estados de $M_1$ é igual ao número de estados de $M_2\}$ é decidível.

Basta construir uma MT M que conta o número de estados de $M_1$, conta o número de estados de $M_2$ e depois compara para determinar se os números são idênticos ou não. Se forem, M pára e aceita. Se não forem, M pára e rejeita.

6ª Questão - (1,0 ponto) Prove que o problema $P_2 = \{<e_1,e_2> \mid e_1$ e $e_2$ são Expressões Reguares e $L(e_1) \neq L(e_2)\}$ é decidível.

Basta converter $e_1$ e $e_2$ para os correspondentes autômatos finitos mínimos e determinar se eles são idênticos, pois o algoritmo de minimização garante que eles são únicos. Em caso afirmativo, $L(e_1)=L(e_2)$ e $<e_1,e_2>$ não pertence à P. Em caso negativo, ou seja, se $L(e_1) \neq L(e_2)$, então $<e_1,e_2>$ pertence à P.

7ª Questão - (1,0 ponto) Prove que o problema $P_3 = \{<M_1,M_2> \mid M_1$ e $M_2$ são Máquinas de Turing e $L(M_1)=L(M_2)\}$ é indecidível (dica: use o Teorema de Rice).

Conforme o enunciado do Teorema de Rice, basta provar que a propriedade em questão é uma propriedade não-trivial da classe das linguagens recursivamente enumeráveis (aqui representada pelas máquinas $M_1$ e $M_2$). De fato, podemos considerar o caso particular das linguagens a*b e b*a. Elas são regulares e portanto recursivamente enumeráveis, e são diferentes uma da outra. Por outro lado, as linguagens a(aa)*b e (aa)*ab também são recursivamente enumeráveis (uma vez que são regulares) e são idênticas, pois representam cadeias que iniciam com um número ímpar de as e terminam com b.

8ª Questão - (1,0 ponto) Defina "redução". Justifique o fato de ela precisar ser total mas não precisar ser injetora nem sobrejetora.

Uma redução é uma função que mapeia instâncias de um problema P1 em instâncias de um problema P2, de tal forma que instâncias cuja resposta é afirmativa são sempre mapeadas em instâncias cuja resposta é também afirmativa, e instâncias cuja resposta é negativa são sempre mapeadas em instâncias cuja resposta também é negativa. A função função precisa ser total pois é necessário poder aplicar a função à qualquer instância do problema original, seja ela afirmativa ou negativa. Ela não precisa ser sobrejetora nem injetora pois ela é usada apenas para determinar se a instância considerada do problema original é afirmativa ou negativa.

9ª Questão - (1,0 ponto) Suponha que $L_U$ reduz para P. Prove que P é indecidível.

A prova é por contradição. Se P fosse decidível, então a combinação da redução de $L_U$ para P, com o algoritmo que decide P poderia ser usado para decidir $L_U$. Como é sabido que $L_U$ é indecidível, segue que a hipótese é falsa e P não pode ser decidível. Logo, P é indecidível.

10ª Questão - (1,0 ponto) Prove que o problema $P_4 = \{<M,w> \mid M$ é uma Máquina de Turing e w é uma entrada para M, tais que M entra em loop com w$\}$ é indecidível (dica: use redução a partir de $L_U$).

A redução a partir de $L_U$ pode ser feita da seguinte forma: construimos uma MT M' que simula M com a entrada w. Se M pára e aceita w, então M' pára. Se M pára e rejeita w, então M' entra em loop. E, naturalmente, se M entra em loop com w, M' entrará em loop também. Logo, temos uma redução de $L_U$ para o complemento de P4. Logo, o complemento de P4 é indecidível, e P4 também é indecidível (pois o complemento de uma linguagem não-recursiva não pode nunca ser uma linguagem recursiva).