

PROVA (PARTE 1)

Universidade Federal de Goiás (UFG) - Regional Jataí Jataí
Bacharelado em Ciência da Computação
Teoria da Computação
Esdras Lins Bispo Jr.

15 de agosto de 2016

ORIENTAÇÕES PARA A RESOLUÇÃO

- A avaliação é individual, sem consulta;
- A pontuação máxima desta avaliação é 10,0 (dez) pontos, sendo uma das 06 (seis) componentes que formarão a média final da disciplina: quatro testes, uma prova e exercícios;
- A média final (MF) será calculada assim como se segue

$$MF = MIN(10, S)$$
$$S = \left(\sum_{i=1}^4 0,2.T_i \right) + 0,2.P + EB$$

em que

- S é o somatório da pontuação de todas as avaliações,
 - T_i é a pontuação obtida no teste i ,
 - P é a pontuação obtida na prova, e
 - EB é a pontuação total dos exercícios-bônus.
- O conteúdo exigido desta avaliação compreende o seguinte ponto apresentado no Plano de Ensino da disciplina: (1) Teoria da Computação, (2) Modelos de Computação e (3) Problemas Decidíveis.

Nome:

Assinatura:

Primeiro Teste

1. (5,0 pt) Explique porque a descrição abaixo não é uma descrição de uma máquina de Turing legítima.

M_{ruim} = “A entrada é um polinômio p sobre as variáveis x_1, \dots, x_k .

- (a) Tente todas as possíveis valorações de x_1, \dots, x_k para valores inteiros.
- (b) Calcule o valor de p sobre todas essas valorações.
- (c) Se alguma dessas valorações torna o valor de p igual a 0, aceite; caso contrário, rejeite.”

R - O problema está nos passos (a) e (b). Testar todas as possíveis valorações para k variáveis inteiras pode ser realizada de forma que o passo (c) nunca seja executado. Se admitirmos $k = 1$ e escolhêssemos a seguinte estratégia:

- (i) Faça o teste para $x_1 = 0$;
- (ii) Faça o teste com x_1 para todos os inteiros negativos;
- (iii) Faça o teste com x_1 para todos os inteiros positivos.

o passo (iii) nunca seria executado, pois o passo (ii) entraria em um laço infinito (pois existem infinitos valores inteiros negativos). Logo, a descrição dos passos (a) e (b) estão inadequadas, não fornecendo informações suficientes para, se necessário, construir a máquina de Turing em um nível formal.

2. (5,0 pt) A operação binária ou-exclusivo, representada pelo símbolo \otimes , é definida da seguinte forma:

$$X \otimes Y = (\overline{X} \cap Y) \cup (X \cap \overline{Y})$$

em que X e Y são dois conjuntos quaisquer.

Mostre que a classe de linguagens decidíveis é fechada sob a operação de ou-exclusivo.

Prova: Sejam A e B duas linguagens decidíveis. É possível construir duas máquinas de Turing M_A e M_B que decidem as linguagens A e B , respectivamente (Definição 3.6). Iremos construir a máquina de Turing M_{aux} , a partir de M_A e M_B , que decide $A \otimes B$. A descrição de M_{aux} é dada a seguir:

M_{aux} = “Sobre a entrada ω , faça:

- (a) Rode M_A sobre ω ;
- (b) Rode M_B sobre ω ;
- (c) Se M_A rejeita e M_B aceita, *aceite*;
- (d) Se M_A aceita e M_B rejeita, *aceite*;
- (e) *Rejeite*.”.

Como foi possível construir M_{aux} , então $A \otimes B$ é decidível. Ora, se $A \otimes B$ é decidível, então a classe de linguagens decidíveis é fechada sob a operação de ou-exclusivo ■

Segundo Teste

3. (5,0 pt) Mostre que a classe de linguagens Turing-reconhecíveis é fechada sob a operação de interseção.

Prova: Sejam A e B duas linguagens Turing-reconhecíveis. É possível construir duas máquinas de Turing M_A e M_B que reconhecem as linguagens A e B , respectivamente (Definição 3.5). Iremos construir a máquina de Turing M_{aux} , a partir de M_A e M_B , que reconhece $A \cap B$. A descrição de M_{aux} é dada a seguir:

M_{aux} = “Sobre a entrada ω , faça:

- (a) Rode M_A sobre ω ;
- (b) Rode M_B sobre ω ;
- (c) Se M_A e M_B aceitam, *aceite*;
- (d) Caso contrário, *rejeite*”.

Como foi possível construir M_{aux} , então $A \cap B$ é Turing-reconhecível. Ora, se $A \cap B$ é Turing-reconhecível, então a classe de linguagens Turing-reconhecíveis é fechada sob a operação de interseção ■

4. (5,0 pt) Considere o problema de se determinar se um AFD e uma expressão regular são equivalentes. Expresse esse problema como uma linguagem e mostre que ele é decidível.

R - Este problema pode ser expresso pela linguagem a seguir:

$$EQ_{AFD-ER} = \{\langle A, B \rangle \mid A \text{ é um AFD, } B \text{ é uma expressão regular e } L(A) = L(B)\}$$

em que qualquer par $\langle A, B \rangle \in EQ_{AFD-ER}$, se o AFD A e a expressão regular B são equivalentes.

Pode-se mostrar que EQ_{AFD-ER} é decidível construindo uma máquina de Turing M que a decida (Definição 3.6). A descrição de M é dada a seguir:

M = “Sobre a entrada $\langle A, B \rangle$, em que A é um AFD e B é uma expressão regular, faça:

- (a) Converta a expressão regular B no AFD equivalente C (Teorema 1.54 e Definição 1.16);
- (b) Construa a MT T que decide EQ_{AFD} (Teorema 4.5);
- (c) Rode T sobre $\langle A, C \rangle$:
 - i. Se T aceita, *aceite*;
 - ii. Caso contrário, *rejeite*.”

Como foi possível construir M , então EQ_{AFD-ER} é decidível ■

Teoremas Auxiliares

Definição 1.16: Uma linguagem é chamada de uma linguagem regular se algum autômato finito a reconhece.

Teorema 1.25: A classe de linguagens regulares é fechada sob a operação de união.

Teorema 1.26: A classe de linguagens regulares é fechada sob a operação de concatenação.

Teorema 1.26.1: A classe de linguagens regulares é fechada sob a operação de complemento.

Teorema 1.39: Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

Teorema 1.49: A classe de linguagens regulares é fechada sob a operação estrela.

Teorema 1.49.1: A classe de linguagens regulares é fechada sob a operação de intersecção.

Teorema 1.54: Uma linguagem é regular se e somente se alguma expressão regular a descreve.

Definição 3.5: Chame uma linguagem de Turing-reconhecível se alguma máquina de Turing a reconhece.

Definição 3.6: Chame uma linguagem de Turing-decidível ou simplesmente decidível se alguma máquina de Turing a decide.

Teorema 3.13: Toda máquina de Turing multifita tem uma máquina de Turing que lhe é equivalente.

Teorema 3.16: Toda máquina de Turing não-determinística tem uma máquina de Turing determinística que lhe é equivalente.

Teorema 3.21: Uma linguagem é Turing-reconhecível se e somente se algum enumerador a enumera.

Teorema 4.1: A_{AFD} é uma linguagem decidível.

Teorema 4.2: A_{AFN} é uma linguagem decidível.

Teorema 4.3: A_{EXR} é uma linguagem decidível.

Teorema 4.4: V_{AFD} é uma linguagem decidível.

Teorema 4.5: EQ_{AFD} é uma linguagem decidível.

Teorema 4.9: Toda linguagem livre-de-contexto é decidível.

Teorema 4.11: A_{MT} é uma linguagem indecidível.

Definição 4.14: Um conjunto A é contável se é finito ou tem o mesmo tamanho que N .