



Exame de Teoria da Computação

DI FCT UNL

23 Jan 2012

1. Defina os seguintes conjuntos:

- (a) O conjunto dos pares ordenados de números naturais em que o valor da primeira componente do par não é maior que o valor da segunda componente do par.
- (b) Uma factura regista um número, uma data, o nome do comprador, a lista com nomes dos produtos comprados e respectivo preço (em cêntimos), e o valor total da factura. Indique qual o conjunto *FACT* com que pode modelar o conjunto de todas as facturas.
- (c) Defina a extensão da função $CUST : \wp(FACT) \times STRING \rightarrow NAT$ que dado um conjunto de facturas e o nome de um comprador devolve quantas facturas existem desse comprador.

2. Modele o seguinte sistema com uma estrutura. Uma máquina de venda de chocolates contém um depósito para os chocolates e um depósito com o dinheiro recebido. Cada chocolate custa 2\$.

- (a) Modele o conjunto de todos os estados possíveis da máquina de venda.
- (b) Modele (com uma função) a operação *fill* que adiciona k chocolates ao depósito respectivo.
- (c) Modele (com uma função parcial) a operação *buy* que compra um chocolate.
- (d) Modele (com uma função) a operação *value* que indica a importância (em dinheiro) em caixa resultante das vendas efectuadas, e retira todo o dinheiro da máquina.

3. Considere o seguinte alfabeto (com 7 símbolos)

$$\Sigma = \{ \text{open_door, close_door,} \\ \text{drive, enter_car, exit_car, turn_motor_on, turn_motor_off} \}$$

Baseado neste alfabeto, defina uma expressão regular cuja linguagem é constituída pelas palavras que descreve uma ou mais interações válidas entre um condutor e um automóvel, sendo que no estado inicial e final o carro deve estar fechado e o condutor fora dele.

4. Considere o alfabeto anterior (abreviando símbolos) e a expressão regular

$$(\text{tm_on d tm_off})^* + (\text{tm_on tm_off})$$

Defina um NFA e depois um DFA que reconheça a linguagem denotada por esta expressão regular. Justifique aplicando o método de construção de um NFA a partir de uma ER e de um DFA a partir de um NFA.

5. Considere o alfabeto (com 3 símbolos) $\Sigma = \{\text{cd_filename, cd_... , not_found}\}$. Seja a gramática $G = \langle V, \Sigma, S, R \rangle$ onde $V = \{S, T\}$ e R contém as regras:

$$\begin{aligned} S &\rightarrow \text{cd_filename } T \\ T &\rightarrow \text{not_found } S \\ T &\rightarrow S \text{ cd_...} \\ S &\rightarrow \epsilon \end{aligned}$$

- (a) Indique uma derivação em G para a palavra

cd_filename notfound cd_filename cd_...

- (b) Verifique se a gramática apresentada é LL(1). Em caso positivo apresente a tabela de transições do analisador sintático determinista. Em caso negativo indique onde existe um conflito de determinismo.

6. A SBTM (Stack-Based Turing Machine) estudada nas aulas possui uma pilha de dimensão arbitrária, ou seja, um programa SBTM pode empilhar na pilha quantos elementos forem necessários. Discuta, justificando, se as seguintes afirmações são verdadeiras ou falsas.

- (a) Existem programas para a SBTM que utilizam um número infinito de posições na pilha. Em caso positivo, dê um exemplo.
- (b) Existem programas para a SBTM que utilizam um número infinito de posições de memória. Em caso positivo, dê um exemplo.
- (c) Um programa SBTM que termine sempre a sua execução só pode utilizar um número finito de células de memória e de posições da pilha.
- (d) Existem funções computáveis pela SBTM que não são computáveis numa SBTM cuja pilha está limitada a um máximo de 8 posições.