

Segundo Teste da UC de Teoria da Computação (LEI, FCT UNL)

Duração: 1h30m

20 Dezembro 2012

Nome:

Número:

Questão 1

Considere um pedómetro para ser instalado numa aplicação de telemóvel. O pedómetro pode executar as seguintes operações: COUNT, que prossegue com a contagem, ou suspende a contagem (funcionando alternadamente), e RESET que coloca a contagem a zero, parando o pedómetro. A operação GETSTEPS devolve o valor no mostrador do pedómetro. Estas operações são para o utilizador. Existe ainda outra operação STEP que será executada pelo hardware cada vez que o utilizador der um passo. O pedómetro só conta os passos quando a contagem não está suspensa. Modele o pedómetro usando uma estrutura.

1. Indique como modelar o conjunto de estados do sistema.
2. Indique o estado inicial do sistema (quando se liga o cronómetro, está a zero e com contagem suspensa).
3. Represente a operação STEP usando uma função ou relação.
4. Represente a operação COUNT usando uma função ou relação.
5. Represente a operação GETSTEPS usando uma função ou relação.
6. Represente a operação RESET usando uma função ou relação.
7. Indique, justificando, se o conjunto de estados do sistema é finito, infinito contável, ou infinito não contável.

Questão 2

Considere o alfabeto $\Sigma = \{0, 1\}$.

1. Defina uma expressão regular que represente todas as sequências de símbolos de Σ que contenham pelo menos uma ocorrência da palavra 101 (por exemplo, 1110010101100 pertence a essa linguagem).
2. Defina um NFA (autómato finito não determinista) que reconheça a linguagem referida na alínea 1.
3. Defina um DFA (autómato finito determinista) equivalente ao NFA definido na alínea 2. Em cada estado do DFA indique o conjunto de estados do NFA que lhe corresponde.

Questão 3

Considere o alfabeto (com 3 símbolos) $\Sigma = \{\text{open, close, end}\}$. Seja a gramática $G = \langle V, \Sigma, S, R \rangle$ onde $V = \{S, P, R\}$ e R contém as regras:

$$\begin{aligned} S &\rightarrow P \text{ end} \\ P &\rightarrow R P \text{ close} \\ P &\rightarrow \epsilon \\ R &\rightarrow \text{open} \end{aligned}$$

1. Indique uma derivação em G para a palavra

open open close close end

2. Calcule o conjunto $First(w)$ para cada regra $A \rightarrow w$ da gramática e o conjunto $Follow(H)$ para cada não terminal $H \in V$.
3. 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.

Questão 4

1. Dizemos que uma SBTM (Stack-Based Turing Machine) aceita a linguagem L sobre o alfabeto Σ se, sempre que inicia a sua computação com uma palavra $w \in \Sigma^*$ no topo da pilha, a sua execução termina com **true** no topo da pilha se $w \in L$ e com **false** se $w \notin L$. Uma palavra w é representada por uma sequência $(a_1, (a_2, (a_3, \dots)))$ de símbolos a_i . Defina uma SBTM que aceita a linguagem definida pela expressão regular $(\text{open close})^* \text{end}$ sobre o alfabeto $\Sigma = \{\text{open, close, end}\}$.