

Métodos de Programação III

LESI + LMCC (3º ano)

Ficha Teórico-Prática nº 5
(Autómatos Deterministas (2))

Ano lectivo 2003/2004

Este texto está escrito em \LaTeX .

O documento é obtido "compilando" este ficheiro em \LaTeX :

latex mp303f05.tex

1 Simulação com Autómatos Reactivos

Exercício 1.1 Desenhe um **Autómato Determinista Reactivo** para simular o comportamento de cada um dos seguintes sistemas:

1. Considere o ambiente representado na figura seguinte:

C_0 : Robot	C_1 : Ouro
---------------	--------------

Um *robot anda à procura de uma pepita de ouro* numa sala com duas células. Quando começa encontra-se na célula C_0 , estando a pepita na célula C_1 . Este robot tem uma capacidade de movimentação bastante limitada, apenas conseguindo andar para a direita, andar para a esquerda, pegar na pepita ou largar a pepita (ordens d , e , p e l , respectivamente). Quando o robot executa uma ordem inconsequente, como por exemplo, andar para a direita em C_1 , ou largar a pepita sem a ter, nada acontece. No momento que o robot larga a pepita na célula C_0 , o ambiente volta ao seu estado inicial, isto é, o robot mantém-se na célula C_0 e a pepita volta a C_1 .

Diga que alterações esse autómato sofreria se quisesse contemplar a situação em que o robot tivesse um defeito no funcionamento das suas garras, originando que, ocasionalmente, quando o robot tenta pegar a pepita, esta continue no chão.

2. Para *classificar pequenos textos* (artigos de jornais ou revistas, memorandos, mensagens electrónicas, etc.) em categorias pré-estabelecidas, usa-se uma tabela de palavras-chave (TPc) e conta-se o número palavras do texto que concordam com essas palavras-chave, calculando-se somatórios pesados por categorias. Na TPc cada entrada (palavra-chave) é associada a uma categoria (que se julga que essa palavra caracteriza) e a um peso (que de algum modo é proporcional à credibilidade dessa associação); assim, sempre que uma palavra do texto existe na TPc, em vez de simplesmente se contar mais uma ocorrência, adiciona-se o respectivo peso a um acumulador associado à categoria correspondente. No fim o texto é considerado como pertencente à categoria que tiver um maior somatório.

Pretende-se, então, desenvolver um programa eficiente que realize a tarefa de classificação, nos moldes acima explicados, à medida que vai lendo, um a um, os caracteres do texto de entrada, considerando que o sistema a desenvolver pode ser encarado como uma Máquina de Transição de Estados.

Para concretizar a ideia, assumo a seguinte TPc

escudos	economia	3
euros	economia	5
economia	economia	20
finanças	economia	20
castelo	cultura	10
livro	cultura	15
literatura	cultura	18
pintura	cultura	18
sky	desporto/lazer	20
campismo	desporto/lazer	18

relativamente às 3 categorias: **economia**, **cultura**, **desporto/lazer**.

Nota: pode considerar que, quando o texto chegar ao fim, será lido o caracter \$.

3. Uma das variantes do conhecido *jogo das setas* (em que se lançam pequenas flechas contra um alvo circular dividido em sectores iguais marcados com pontos de 1 a 20) é designada por *jogo dos 36*: um jogador ganha quando, ao fim de lançar várias setas, conseguir acumular exactamente 36 pontos; perderá logo que a soma ultrapasse esse valor. Contudo só contam os pontos quando uma seta atingir um dos seguintes sectores: 20, 17, 16, 12, 10, 8, 6, 1; cada vez que uma seta caia noutra sector, com pontos diferentes destes, a jogada não conta, é ignorada.

Considerando que o sistema a desenvolver pode ser encarado como uma Máquina de Transição de Estados, pretende-se, desenvolver um programa eficiente que vá recebendo os pontos obtidos por um jogador e determine se este deve continuar a jogar, ou se deve parar por já ter ganho, ou por ter perdido.

4. Modele o comportamento do *editor de texto vi*, numa sua versão muito simplificada.

Como é sabido esse editor está normalmente em modo de aceitação de comandos, comandos esses que são descritos por 1 carácter (letra ou o sinal ':') eventualmente seguido de argumentos (outra letra ou número (um ou mais dígitos)).

Neste caso considere apenas os comandos: sair (':q'); gravar (':w'); apagar um palavra, toda a linha, 'n' linhas ('dw' — 'dd' — 'dn'); inserir texto ('i' seguido de quaisquer caracteres até aparecer o carácter 'ESC').

5. Modele o funcionamento de uma *máquina pública de fotografias tipo passe*.

Suponha que a máquina parte dum estado inicial em que está à espera da selecção do número de fotos pretendida (2, 4 ou 8); uma vez escolhido o número de fotos, a máquina fica a aguardar a introdução de dinheiro (moedas de 50 cêntimos, 1 ou 2 euros)—2 fotos custam 1 euro, 4 custam 1,50 e 8 custam 2 euro. Introduzido o dinheiro no montante igual ou superior à quantia correcta (não há devolução de troco), a máquina aguarda que o cliente se posicione devidamente e accione o botão de arranque; então, acende uma luz vermelha que passará a verde após os disparos necessários. Finalmente aguarda que um sensor indique que o cliente pegou nas fotografias produzidas e volta ao estado inicial.

6. Pretende-se simular o funcionamento de uma *máquina de vender cho-*

colates. Para resolver o problema, defina o stock que teria no início do dia (nome, preço e quantidade de cada produto disponível).

Considere que vai processar os registos das vendas diárias (nome do chocolate escolhido e a quantia introduzida)—comece por tratar o caso de 1 registo e depois expanda a solução para vários registos. O objectivo é calcular: a evolução do stock ao longo do dia; e o dinheiro acumulado.

A máquina não fornecerá nenhum chocolate se: o produto seleccionado não existir em stock; ou o nome do produto não constar da lista de produtos disponíveis na máquina; ou a quantia introduzida for inferior ao preço. Em qualquer um destes casos o movimento de venda é ignorado.

Caso a quantia introduzida não seja exacta, a máquina deverá restituir o excesso. Note que a quantia acumulada deverá corresponder ao número de chocolates vendidos vezes o preço respectivo.

7. Pretende-se simular o comportamento de um *aspirador industrial automático* que faz a limpeza de uma unidade industrial rectangular (de dimensões conhecidas à partida), de modo a visualizar-se no monitor as zonas que foram limpas e as que ficaram por aspirar. Os movimentos do aspirador serão sempre em linha recta para a esquerda ou direita, para cima ou para baixo. Em cada deslocação pode ter o sistema de aspiração ligado ou desligado e pode mover-se um número de passos variável (por omissão será 1 passo).

O canto inferior esquerdo do pavilhão é o ponto de repouso, tendo, por isso, coordenadas (0,0); considere que esta é sempre a posição inicial. O robot é ligado automaticamente pelo primeiro comando de deslocação; será desligado pelo comando de repouso que o coloca na posição inicial.

Considere que vai processar uma sequência de comandos de deslocação e comando da aspiração do robot. O processador deve validar os movimentos do robot de modo a não deixar ultrapassar os limites.