

UNIVERSIDADE FERNANDO PESSOA**Disciplina:** Programação**Ano:** 1º**Ano Lectivo:** 2002/2003**Exame:** sem consulta;**Duração:** 2h 00m**Data:** 21 de Julho de 2003**Licenciaturas:** Engenharias (Ambiente, Civil, Comunicação, Informática)

1)

1.1) Escreva um programa em pseudo-código que leia, acerca de um jogo de futebol, o nome do clube, o nome do clube oponente o número de golos marcados e sofridos e imprima no ecrã (formatado tal como indicado no exemplo) se foi vitória (3 pontos), empate (1 ponto) ou derrota (0 pontos), a pontuação obtida no jogo, o nome dos clubes assim como os golos marcados por ambos.

Exemplo:

Nome? FC Portugal

Oponente? S Benin

Golos Marcados? 3

Golos Sofridos? 1

(Vitória – 3 pontos) FC Portugal 3, S Benin 1

1.2) Escreva o respectivo algoritmo usando um fluxograma.

1.3) Faça o seguimento e teste (rastreamento) do algoritmo para o exemplo apresentado em cima.

1.4) Altere o programa anterior de forma a ler para o mesmo clube os nomes de 18 jogos de futebol da primeira divisão e apresentar a pontuação do clube no ecrã. Sugestão: utilize os vectores `nome_oponente: array[1..18] of string[15]`; `marcados, sofridos, pontos: array[1..18] of integer`;

2) Dada uma matriz de números reais com tamanho 128X128 representando a intensidade dos pontos de uma imagem, considere que cada posição da matriz está preenchida com valores compreendidos entre 0 e 1. Escreva um algoritmo que para cada ponto interior da matriz (que são todos os pontos da matriz excepto os pontos das linhas ou colunas da periferia da matriz) o substitua pela média de si próprio com todos os seus 8 pontos vizinhos. Vá escrevendo o resultado numa nova matriz com as mesmas dimensões da matriz original. Como exemplo pode ver na matriz de baixo que os pontos vizinhos do ponto p são $v1, v2 \dots v8$.

| | | |
|----|----|----|
| v1 | v2 | v3 |
| v8 | p | v4 |
| v7 | v6 | v5 |

3) Considere três vectores de strings: `tit_lisboa, tit_ny, tit_tokyo[]: array[1..100] of string[30]`; com os nomes dos títulos cotados nas bolsas de Lisboa, Nova Iorque e Toquio. Considere ainda três vectores: `cot_lisboa, cot_ny cot_tokyo: array[1..100] of integer`; com as respectivas cotações. A cotação das acções de Lisboa é dada em euros, a de Nova Iorque em dólares e a de Tóquio em ienes. Considere ainda que a conversão de ienes para euros e de dólares para euros é feita recorrendo as variáveis `iene_euro` e `dolar_euro` usando a fórmula:

$$\langle \text{valor em euros} \rangle \leftarrow \langle \text{valor em ienes} \rangle * \text{iene_euro}$$
$$\langle \text{valor em euros} \rangle \leftarrow \langle \text{valor em dolares} \rangle * \text{dolar_euro}$$

Sabendo que a constituição da carteira de títulos de um cliente é dada por uma matriz `carteira: array[1..n_tit, 1..3] of integer`; em que `n_tit` é o número de títulos do cliente e para cada título os três valores contêm a seguinte informação: número do título, número de acções que o cliente possui desse título e bolsa a que pertence o título (1 - Lisboa, 2- Nova Iorque e 3 - Tóquio). Escreva um algoritmo que calcule o valor da carteira desse cliente em euros usando as conversões respectivas.

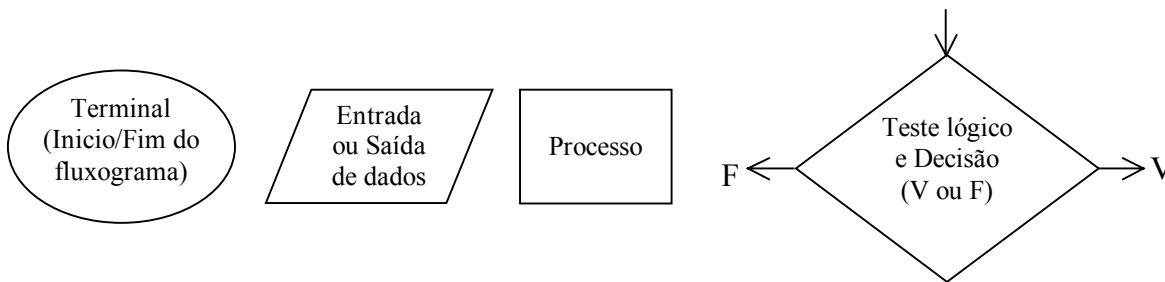
[Cotação em valores: 1.1) 3; 1.2) 2; 1.3) 2; 1.4) 2; 2) 5,5; 3) 5,5]

Sugestões de notação para descrição dos seus algoritmos ao longo do teste:

Pseudo-código:

- Leia(...)
- Escreva(...)
- Para <expressão_inicial> Até <expressão_final> Faça <bloco>
- Se <condição> Então <bloco> Senão <bloco>
- Repita <bloco> AtéQue <condição>
- Enquanto <condição> Faça <bloco>
- Caso <expressão> Seja <caso_1> : <bloco> <caso_n> : <bloco> Senão <bloco> FimCaso
- <operador de atribuição>: ←
- <bloco>: Inicio Fim

Fluxogramas:



(Nota1: poderá, principalmente na descrição textual do algoritmo, usar outra notação com a qual esteja mais à vontade como por exemplo o Pascal ou outra. Não se preocupe demasiado com a sintaxe mas não deixe de ser claro na descrição do seu algoritmo)

Nota2: poderá usar os operadores relacionais, lógicos e aritméticos normais assim como algumas funções matemáticas mais vulgares)

Vectores e Cadeias de Caracteres:

Considere os vectores e matrizes como sendo de tamanho fixo

Relativamente às variáveis do tipo *string* considere o seguinte:

- Pode consultar/alterar o valor de um caracter dentro da string através da operação normal de acesso a elementos de um vector
- Operador de concatenação de strings: +
- Função que devolve o comprimento da string: length(s)
- Função para copiar/extrair substring de uma string: copy (s, index_ini, count)
- Função para determinar posição de uma substring dentro de uma string: pos(substr, s)
- Conversão de string para um valor numérico: val(s, v, error_code)
- Conversão de um valor numérico para string: str(x, s)