

Departamento de Matemática — Universidade de Coimbra

Ano Lectivo de 2003/2004

Métodos de Programação II

Exame 26/1/2004

Leia atentamente o enunciado de cada pergunta antes de iniciar a sua resolução.

Sempre que necessário declare os tipos de dados que usar.

Duração da prova: 2h30

Os programas devem ser escritos em *Pascal*

1. Pretende-se definir o Tipo Abstracto de Dados Racionais como sendo pares de inteiros ($\mathbb{Q} = \mathbb{Z} \times \mathbb{Z}$), com as operações usuais:
- X(a) diga o que se entende por um Tipo Abstracto de Dados (5 linhas no máximo!);
 - X(b) defina um tipo que implemente os objectos do T.A.D. Racionais;
 - X(c) diga quais as operações que o T.A.D. Racionais deve possuir, e escreva os cabeçalhos dos sub-programas correspondentes.
2. O operador de telecomunicações *TeleLigaçõesPT* taxa as ligações GPRS (internet/tele-móveis) da seguinte forma:
- T1 — Fins-de-semana e Feriados — 0.20€/MB
 - T2 — Dias da semana — 0.70€/MB

Pretende-se calcular o montante a pagar por um dado cliente num dado período de tempo (e.g. um mês). Escreva um sub-programa que dado um vector contendo n pares (t, m) , referentes ao tipo da ligação e aos "megabytes" (MB) transferidos, calcule e devolva o montante a pagar pelo cliente. Assuma que o vector contém os dados referentes ao período em questão.

3. Uma matriz $A_{n \times n}$ diz-se estritamente diagonal dominante por linhas, se e só se:

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \quad \text{para } i = 1, \dots, n.$$

Escreva um sub-programa que verifique se uma dada matriz quadrada é estritamente diagonal dominante por linhas.

4. Escreva, usando um algoritmo recorrente, um sub-programa que, dados um elemento k e um vector v de n ($0 < n \leq 100$) elementos inteiros, determine o número de ocorrências de k no vector.
5. Dados $(a, b), (c, d) \in \mathbb{Z} \times \mathbb{Z}$, diz-se que (a, b) é lexicograficamente menor que (c, d) , se $a < c$, ou se $a = c$ e $b < d$.

Considerando os tipos de dados

```
type pares = record x,y: integer
end;
vector = array [1..1000] of pares;
```

escreva um sub-programa para ordenar lexicograficamente um dado vector.

v.s.f.f.

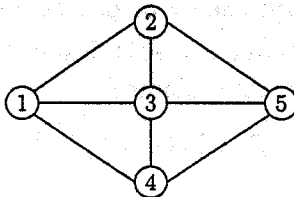
6. Um grafo é uma estrutura, (N, A) , com um conjunto finito de nós $N = \{1, \dots, n\}$ e um conjunto de arcos A , de elementos $\{i, j\}$, tais que $i, j \in N$, cada um representando uma ligação entre um par de nós.

Uma tal estrutura pode ser identificada, por exemplo, pelos nós que a constituem e pelos nós a que cada um deles se liga. Deste modo, considere que um grafo é representado, em Pascal, pelas seguintes declarações:

```
type nos      = 1..100;
     vizinhos  = set of nos;
     grafo    = array [nos] of vizinhos;
```

em que $i \in N$ se diz um vizinho de $j \in N$, se e só se $\{i, j\} \in A$.

Por exemplo, a figura seguinte mostra um grafo cujos nós são os círculos numerados e os arcos são as linhas que unem cada par de nós.



De acordo com as declarações acima o grafo na figura seria representado pelo vector:

{2, 3, 4}	{1, 3, 5}	{1, 2, 4, 5}	{1, 3, 5}	{2, 3, 4}
-----------	-----------	--------------	-----------	-----------

Escreva sub-programas para:

- Calcular o grau (nº de vizinhos) de um nó i num grafo.
- Remover um nó i de um grafo (e, claro, todos os arcos que incidem em i).
- Verificar se um nó j é atingível a partir de um nó i num grafo, isto é, se existe pelo menos uma sequência de arcos que liga i a j .

SUGESTÃO: Utilize recorrência.