

Departamento de Matemática — Universidade de Coimbra

Ano Lectivo de 200x/200(x + 1) Introdução aos Computadores e Programação Exame Tipo

Duração da prova: 2h30

1. (Pergunta (teórica) simples)

Diga o que entende por *metodologia de programação estruturada e descendente*.

2. (Pergunta muito simples sobre os condicionais, pretende-se como resposta um programa em FORTRAN)

Escreva um programa, em FORTRAN, que calcule as raízes reais de uma equação do segundo grau $ax^2 + bx + c = 0$. O programa deve ler os valores de a , b , e c , e deve escrever o tipo de raízes assim como, conforme os casos, os seus valores.

3. (Pergunta muito simples sobre os ciclos, pretende-se como resposta um programa em FORTRAN)

Dado uma lista de n valores e respectivos pesos $(v_i, p_i), 1 \leq i \leq n$, calcule a sua média ponderada $m = \frac{\sum_{i=1}^n v_i p_i}{n}$.

4. (Pergunta simples envolvendo todas os tipos de instruções simples aprendidas, pretende-se como resposta um programa em FORTRAN)

Num posto de controle de tráfego pretende-se contabilizar o número de veículos que aí passam fazendo a separação em três classes: *motos*, *carros*, e *camiões*. Escreva um programa em FORTRAN que vá lendo valores inteiros, respeitantes aos códigos das classes, até ao momento em que o operador decida terminar a contagem, após isso o programa deve imprimir o valor do tráfego, em percentagem, para cada uma das classes.

5. (Pergunta simples envolvendo os tipos de dados estruturados, pretende-se como resposta um programa em FORTRAN)

Escreva um programa, em FORTRAN, que dado um vector de elementos naturais a ler do canal de entrada (teclado), inverta a ordem pela qual se encontram, sem que para tal use um vector auxiliar.

6. (Pergunta envolvendo sub-programas, pretende-se como resposta um sub-programa em FORTRAN)

Dado um vector de n componentes reais $\{x_i\}_{i=1, \dots, n}$ pretende-se normalizá-lo, isto é, dividir cada componente pela norma do vector ($x_i = x_i / \|x\|, i = 1, \dots, n$. $\|x\| = \sqrt{\sum_{i=1}^n x_i^2}$).

- Construa um sub-programa, em FORTRAN, que dado um vector calcule a sua norma;
 - Construa um sub-programa, em FORTRAN, que dado um vector o normalize;
 - Escreva o respectivo programa principal.
-

Respostas

1. Por metodologia de programação estruturada e descendente entende-se o método de construção de programas em que: divide-se o problema em sub-problemas de menor complexidade, repetindo o processo até se atingir problemas de resolução simples; resolve-se cada um desses problemas isoladamente construindo o sub-programa respectivo; combinam-se os sub-programas de forma a construir o programa global.

2. PROGRAM equacao2grau

```
IMPLICIT NONE

REAL :: a,b,c
REAL :: x1,x2,binomio

WRITE (*,*) "Introduza a,b,c: "
READ (*,*) a,b,c

binomio = b**2-4*a*c
IF (binomio < 0) THEN
  WRITE (*,*) "Nao tem raizes reais"
ELSE
  IF (binomio == 0) THEN
    x1 = -b/(2*a)
    WRITE (*,*) "Uma raiz real dupla, ",x1
  ELSE
    x1 = (-b+SQRT(binomio))/(2*a)
    x2 = (-b-SQRT(binomio))/(2*a)
    WRITE (*,*) "Duas raizes reais, ",x1,x2
  END IF
END IF
END PROGRAM
```

3. PROGRAM media_ponderada

```
IMPLICIT NONE

REAL :: media, vi, pi
INTEGER :: n,i

WRITE (*,*) 'Quantos valores/pesos quer introduzir? '
READ (*,*) n
WRITE (*,*) 'Introduza agora os ',n,' pares, valor/peso'
media = 0
DO i=1,n
  READ (*,*) vi,pi
  media = vi*pi
END DO
media=media/n
WRITE (*,*) 'A media ponderada e': ',media
END PROGRAM media_ponderada
```

4. PROGRAM trafego

```
IMPLICIT NONE

INTEGER :: moto, carro, camiao, terminar
INTEGER :: veiculo, total

! inicializacao
moto=0
carro=0
camiao=0
total=0

WRITE (*,*) 'Introduza o codigo da classe que pretende contabilizar'
WRITE (*,*) '0=moto, 1=carro, 2=camiao, 9=terminar'

! fazer a contagem
READ (*,*) veiculo
DO WHILE (veiculo /= 9)
  SELECT CASE (veiculo)
  CASE (0)
    moto=moto+1
  CASE (1)
    carro=carro+1
  CASE (2)
    camiao=camiao+1
  CASE default
    WRITE (*,*) 'Codigo errado'
    total=total-1
  END SELECT
  total=total+1
  READ (*,*) veiculo
END DO

! calculo das percentagens
moto=100*moto/total
carro=100*carro/total
camiao=100*camiao/total

! escrever o resultado
WRITE (*,*) 'Percentagem de motos: ', moto,'%
WRITE (*,*) 'Percentagem de carros: ', carro,'%
WRITE (*,*) 'Percentagem de camioes: ', camiao,'%
END PROGRAM trafego
```

5. PROGRAM inverte_vector

```
IMPLICIT NONE

INTEGER :: dim
INTEGER, DIMENSION(1000) :: v

INTEGER :: i,aux

WRITE (*,*) 'Introduza os elementos do vector '

WRITE (*,*) '(termine com um numero negativo)'

! leitura dos elementos
i=1
READ (*,*) aux
DO WHILE (aux > 0)
    v(i)=aux
    i=i+1
    READ (*,*) aux
END DO
dim=i-1
! inversao da ordem, por troca dos elementos em
! posicoes diametralmente opostos
DO i=1,dim/2
    aux=v(i)
    v(i)=v(dim+1-i)
    v(dim+1-i)=aux
END DO
! escrita do vector, recorrendo a um "do" implicito
WRITE (*,*) (v(i),i=1,dim)
END PROGRAM inverte_vector
```

6. (a) SUBROUTINE norma(dim,v,nv)

```
IMPLICIT NONE

INTEGER, INTENT(IN) :: dim
REAL, INTENT(IN), DIMENSION(100) :: v
REAL, INTENT(OUT) :: nv
INTEGER :: i

nv=0
DO i=1,dim
    nv=nv+v(i)**2
END DO
nv = SQRT(nv)
END SUBROUTINE norma
```

(b) SUBROUTINE normaliza(dim,nv,v)

```
IMPLICIT NONE

INTEGER, INTENT(IN) :: dim
REAL, INTENT(IN) :: nv
REAL, INTENT(INOUT), DIMENSION(100) :: v
INTEGER :: i

DO i=1,dim
    v(i)=v(i)/nv
END DO
END SUBROUTINE normaliza
```

(c) PROGRAM testa_normaliza

```
IMPLICIT NONE

INTEGER :: dim
REAL, DIMENSION(100) :: v
INTEGER :: i
REAL :: nv

! ler os dados de entrada
WRITE (*,*) 'Qual e'' a dimensao do vector? '
READ (*,*) dim
WRITE (*,100) dim
100 FORMAT (1x,'Introduza os',I3,' elementos do vector')
READ (*,*) (v(i),i=1,dim)
! calcular a norma
CALL norma(dim,v,nv)
! normalizar o vector
CALL normaliza(dim,nv,v)
! escrever o vector resultante
WRITE (*,200) (v(i),i=1,dim)
200 FORMAT(1x,20F6.2)
END PROGRAM testa_normaliza
```