

- Leia atentamente o enunciado completo antes de responder.
  - **Responda a cada grupo em folhas separadas.**
- 

**GRUPO I (6 valores)**

1 - Responda, numa frase, a cada uma das alíneas seguintes:

- a) Dê 2 exemplos de unidades de leitura (*Input*) e 2 exemplos de unidades de escrita (*Output*).
- b) Qual a diferença entre o *Bit* e o *Byte*.
- c) Para que serve a instrução `IMPLICIT NONE`.
- d) Represente o número  $21.75_{10}$  na base binária (apresente os cálculos efectuados).

2 - Num programa em *Fortran* foram declaradas as seguintes variáveis:

```
REAL :: a, b, c
INTEGER :: k, m, n
```

Qual o valor que irá ser armazenado em cada uma das variáveis:

```
a = 4 + 5 / 2 * 2.5
b = 4 + 5 / (2 * 2.5)
c = (4 + 5) / 2 * 2.5
k = 3 * SQRT (3.0 + 4 / 3) / 1.5 * 4
m = 8.5 / 2 + 0.75
n = 2**2**2 + 2**(4 + 3 / 2)
```

3 - Num programa em *Fortran* definiram-se as seguintes instruções:

```
REAL :: a = -125.7255, b = -2.17243
INTEGER :: k = 2452, m = -425134
CHARACTER (LEN = 6) :: palavra = 'tudo bem'

WRITE (*, "(2(2X,F6.2),X,I6)") a, b, m
WRITE (*, "(X,E10.3,3X,I5,2X,'esta',A4)") a, k, palavra
```

Apresente o resultado da execução dessas instruções.

4 - Escreva um subprograma que recebe uma sequência de N valores inteiros e determina o menor e o maior par.

## Grupo II (5 valores)

Na disciplina de "Programação e Computadores" os dados referentes à avaliação de cada aluno serão os seguintes:

Informação	Variável correspondente
nome do aluno (primeiro e último)	ALUNO ( i )
Classificação no 1º mini-teste (0 a 2 ou -1)	CLASS ( i , j ) i, ordem do aluno, 1 a N j, ordem do item de classificação, 1 a 4
Classificação no 2º mini-teste (0 a 2 ou -1)	
Classificação nos três grupos obrigatórios (0 a 16 ou -1)	
Classificação no 4º grupo opcional (0 a 4 ou -1)	

O nome de cada aluno será armazenada na variável ALUNO do tipo `character` com comprimento máximo de 30 e as classificações na variável CLASS. As classificações deverão ser introduzidas arredondadas às decimas. Os alunos que não foram avaliados em qualquer dos 4 itens terão a respectiva classificação codificada por -1.

- a) Escreva um subprograma em *Fortran* que recebe o número total de alunos, N, e a variável CLASS e calcula a nota final de cada aluno, arredondada às unidades, de acordo com o seguinte critério:

$$\text{Nota final} = \begin{cases} 1^\circ \text{ item} + 2^\circ \text{ item} + 3^\circ \text{ item} & ; \text{ se não fez o 4º grupo} \\ 3^\circ \text{ item} + 4^\circ \text{ item} & ; \text{ caso contrário} \end{cases}$$

(**Atenção:** não somar as classificações com valor igual a -1)

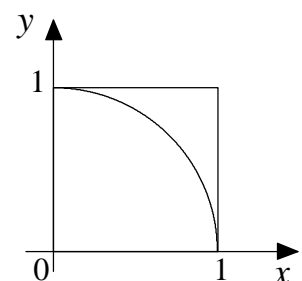
- b) Escreva um programa principal que lê toda a informação e determina, para cada aluno, a nota final usando o subprograma anterior. A pauta classificativa será escrita com o seguinte aspecto:

1 a N linhas {	Nome	1ºmini-teste	2ºmini-teste	Grupos obrigatórios	Grupo opcional	Nota final
	...	...	...	...	...	...

O programa deverá ainda calcular e escrever a nota final média.

## Grupo III (5 valores)

O valor de  $\pi$  pode ser estimado utilizando pontos aleatórios no interior de um quadrado de lado unitário, vértice inferior esquerdo em (0,0) e vértice superior direito em (1,1). É dado por:  $p \cong 4 \frac{NPC}{NPQ}$ , onde NPC é o número de pontos no interior da restrição da circunferência de raio unitário No primeiro quadrante e NPQ é o número total de pontos no interior do quadrado (ver figura ao lado).



- a) Escreva um subprograma em *Fortran* que gere um ponto de coordenadas aleatórias  $x$  e  $y$ , no interior do quadrado acima referido. Recorde que a subrotina `RANDOM_NUMBER(X)` atribui (pseudo-)aleatoriamente à variável  $X$  um número entre 0 e 1.
- b) Escreva um programa em *Fortran* que leia o número de pontos aleatórios a utilizar na estimativa e que calcule e apresente o valor aproximado de  $\pi$  para esse número de pontos.

# Teste de Programação e Computadores – FEUP – Dep<sup>to</sup> Engenharia Civil

Duração: 30m+15m

10/01/2000

- Leia atentamente o enunciado completo antes de responder.
- 

## Grupo IV (4 valores)

O método iterativo simples dá uma aproximação numérica à solução de equações não lineares. Consiste em escrever a equação na forma  $x = f(x)$ . Partindo de um valor inicial  $x_0$  e aplicando repetidamente a expressão  $x_{novo} = f(x_{anterior})$  a solução é sucessivamente refinada, desde que este processo seja convergente. O processo pára quando a diferença  $|x_{novo} - x_{anterior}|$  for inferior a um dado erro  $\varepsilon$  (se o processo é convergente), ou quando se atinge um número máximo de repetições previamente fixado.

- Escreva um subprograma que recebe o valor inicial  $x_0$ , o erro  $\varepsilon$  e o número máximo de repetições; e, determina a solução de uma equação da forma  $x = f(x)$  se durante o processo iterativo a diferença  $|x_{novo} - x_{anterior}|$  for alguma vez inferior a  $\varepsilon$ , ou devolve o valor 9999 em caso contrário. Note que a função  $f$  é implementada como uma FUNCTION.
- Usando o subprograma da alínea anterior, escreva um programa principal que resolva a equação  $x = \ln(x) + 2$ .