

# Instituto Superior de Engenharia do Porto

## Departamento de Engenharia do Porto

### Algoritmia e Programação

#### 1º Exercício Prático

Nome: \_\_\_\_\_

Número: \_\_\_\_\_

Turma: \_\_\_\_\_

**Duração: 20 minutos**

1. Escreva um algoritmo que dado três números indique qual o maior e o menor.

ED: num1, num2,num3,maior,menor inteiro

Algoritmo

Início

```
ler(num1,num2,num3)
maior←num1
menor←num1
se (num2 > maior)
    então maior←num2
senão
    se (num2 <menor)
        então menor←num2
    fse
fse
se (num3 > maior)
    então maior←num3
senão
    se (num3 <menor)
        então menor←num3
    fse
fse
escrever("O maior : "maior" O menor: "menor)
```

Fim

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    int num1,num2,num3,maior,menor;
    printf("Introduza os numeros: ");
    scanf("%d%d%d",&num1,&num2,&num3);
    maior=menor=num1;
    if(num2>maior)
        maior=num2;
    else
        if(num2<menor)
            menor=num2;
    if(num3>maior)
        maior=num3;
    else
        if(num3<menor)
            menor=num3;
    printf("Maior:%d\nMenor: %d\n",maior,menor);
```

```
}
```

2. Descreva um algoritmo que dado um numero, determine se é um numero perfeito. Um nº perfeito é aquele cuja soma dos seus divisores é igual ao próprio nº.

Exemplo:

- 6 -> 1+2+3 = 6 => "é perfeito"
- 8 -> 1+2+4 = 7 => "não é perfeito"

<pre>ED: num, i, soma inteiro Algoritmo Inicio     ler(num)     soma←0     Repetir para i←1 ate (num/2) fazer, passo 1         se (num mod i = 0)             então soma←soma+i         fse     Fpara     se (soma =num)         então escrever(num "é perfeito")     senão         escrever(num "não é perfeito")     fse Fim</pre>	<pre>#include&lt;stdio.h&gt; void main() {     int num,i,soma;     printf("Introduza o numero: ");     scanf("%d",&amp;num);     soma=0;     for(i=1;i&lt;=num/2;i++){         if((num%i)==0)             soma=soma+i;     }     if(soma==num)         printf("%d e perfeito\n",num);     else         printf("%d nao e perfeito\n",num); }</pre>
--	---

3. Indique e justifica o que este algoritmo faz. Faça a traçagem para complementar a resposta.

```

DEF: n, s, i INTEIRO
ALGORITMO
  INICIO
    1-      s ← 0
    2-      LER(n)
    3-      REPETIR PARA i ← 1 ATE n FAZER, PASSO 1
    4-          s ← s + 1 / (2i - 1)
    5-      FPARA
    6-      ESCREVER(s)
  FIM
  
```

Este algoritmo calcula a seguinte série  $S = \sum_{i=1}^n \frac{1}{2^i - 1}$  que tem como particularidade o facto dos elementos da série serem  $\frac{1}{\text{numeros\_impares}}$ , isto é,  $S = \frac{1}{1} + \frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \frac{1}{9} + \dots$

	n	s	i	i <= n	Saída
1		0			
2	5				
3			1	v	
4		0+1/1			
3			2	v	
4		1+1/3			
3			3	v	
4		(1+1/3) +1/5			
3			4	v	
4		(1+1/3+1/5) +1/7			
3			5	v	
4		(1+1/3+1/5 +1/7) +1/9			
3			6	f	
6					1+1/3+1/5 +1/7+1/9