

PROPOSTAS DE TRABALHO E AUTO-AVALIAÇÃO

1. Considera o método de abordagem de algoritmos e programação conhecido por abordagem *top-down*.

- 1.1. Esclarece o significado da expressão *top-down* neste contexto.
- 1.2. Em que situações é que este método de abordagem é especialmente útil?
- 1.3. Explicita a finalidade ou utilidade principal dos subprogramas ou subalgoritmos.

2. Considera os dois tipos de subprogramas existentes na linguagem Pascal.

- 2.1. Explicita as principais diferenças entre um procedimento (*procedure*) e uma função (*function*).
- 2.2. Explicita a diferença entre variáveis globais e variáveis locais.

3. Considera a existência de subprogramas com parâmetros.

- 3.1. Explica o que são parâmetros no contexto de subprogramas e para que servem esses parâmetros.
- 3.2. Explicita a diferença entre parâmetros e argumentos (quando é que se usam uns e quando se usam os outros).
- 3.3. Explicita a diferença entre passagem de argumentos a parâmetros:
 - a) por valor;
 - b) por referência.

4. Os dois programas a seguir apresentados visam calcular a raiz quadrada de um número introduzido pelo utilizador. O primeiro utiliza um procedimento e o segundo uma função.

```

Program RaizQuadrada1 ;
Var N, R : Real ;

Procedure RaizQ ;
begin
  R := SQRT (N) ;
end ;

Begin {Programa principal}
  Write ('Introduza um número');
  Read (N);
  RaizQ;
  Write ('A raiz quadrada de ',
        N , 'é ', R)
End.
  
```

```

Program RaizQuadrada2 ;
Var N : Real ;

Function RaizQ : Real ;
  {...}

Begin {Programa principal}
  Write ('Introduza um número');
  Read (N);
  {...}
End.
  
```

- 4.1. Completa o programa RaizQuadrada2 nos espaços sinalizados com {...}, de modo que ele funcione do mesmo modo que RaizQuadrada1.
- 4.2. A partir deste exemplo concreto, esclarece a diferença entre um procedimento (*procedure*) e uma função (*function*) ao nível da estrutura de cada um e da forma como são chamados a executar no programa.

5. O algoritmo abaixo apresentado pede ao utilizador dois números inteiros e visa calcular a soma dos números ímpares compreendidos entre o primeiro e o segundo.

```

Algoritmo Somar_números_ímpares ;
Váriáveis N1, N2, Conta, Soma : Inteiros ;

Início
  Escrever ('Introduza um 1º número');
  Ler (N1);
  Escrever ('Introduza um 2º número');
  Ler (N2);
  
```

```

Se N2 < N1 Então
  Escrever ('O 2º número tem de ser maior');
Senão
  Início
  Soma ← 0;
  Para Conta ← N1 Até N2 Fazer
    Se (Conta Mod 2) = 1 Então
      Soma ← Soma + Conta
    FimSe
  FimFazer
  Fim
FimSe ;
  Escrever ('A soma dos ímpares é', Soma);
Fim.
  
```

- 5.1. Traduz o algoritmo para Pascal.
- 5.2. Escreve uma nova versão do programa em que a parte que está delimitada pelo retângulo vermelho seja transformada num procedimento com o nome Calcula_Soma.
- 5.3. Verifica se, das quatro variáveis declaradas no algoritmo, alguma delas pode passar a variável local do procedimento Calcula_Soma. Justifica a tua resposta.
- 5.4. Escreve uma nova versão do programa em que o procedimento Calcula_Soma é transformado na função Valor_Soma, sem alterar o funcionamento geral do programa.

6. Considera o programa abaixo apresentado.

```

Program EscreveCaracteres ;
Var Q : Integer ;

Procedure EscreveX ;
Var i : Integer ;
begin
  For i := 1 To Q Do
    Write ('X') ;
  end ;
end ;

Begin {Programa principal}
  Write ('Quantos caracteres?');
  Read (Q);
  EscreveX;
End.
  
```

- 6.1. Supõe que é introduzido o valor 10 quando, na parte principal do programa, é pedido 'Quantos caracteres?'. Descreve detalhadamente o que irá acontecer em seguida no programa.
- 6.2. No programa apresentado, existem duas variáveis (Q e i). Qual a diferença entre elas? A variável i, tal como está declarada, pode ser utilizada fora do procedimento EscreveX? Justifica a tua resposta.
- 6.3. Escreve uma nova versão do procedimento EscreveX em que este passa a ter um parâmetro P que irá indicar quantas vezes o carácter X deverá ser escrito dentro do ciclo For. Proceda às rectificações que achares necessárias para que o programa continue a funcionar correctamente.
- 6.4. Escreve uma nova versão do programa EscreveCaracteres, utilizando uma variável K do tipo *Char*, que irá servir para o utilizador indicar qual o carácter que pretende que seja escrito Q vezes. O procedimento EscreveX deve conter um parâmetro para fazer entrar o carácter indicado pelo utilizador em K.