

Leia atentamente o enunciado de cada pergunta antes de iniciar a sua resolução.
Sempre que use tipos não-elementares deve ser apresentada a respectiva declaração de tipo que usar.

Exercício 1 (3,00)

- Apresente o *Modelo de Von Neumann* e descreva o funcionamento de cada uma das unidades do modelo.
- Quais as funções gerais de um Sistema Operativo (SO)? Explique porque é este considerado como o “*software* mais importante”.

Exercício 2 (2,25)

- Considere declaradas as seguintes variáveis num certo programa:

```
var k : integer;
    p, q, r: boolean;
    x : real;
```

Indique o tipo e calcule o valor de cada uma das expressões seguintes:

- | | |
|--|---|
| 1) $k := 5 + 6 * (k \bmod 2)$ | 6) $\text{ord}('h') - \text{ord}('a') + x$ |
| 2) $\text{not}(p \text{ or } q) = \text{not } p \text{ and } q$ | 7) $\text{chr}(\text{ord}('0') + 7)$ |
| 3) $k := (16 - \text{trunc}(\ln(\exp(1.0)))) / 4$ | 8) $\text{succ}(\text{round}(6.78))$ |
| 4) $(p \text{ and } (q \text{ and } \text{not } q)) \text{ or } \text{not } r$ | 9) $(\ln(x) \text{ and } x > 0) \text{ or } (\ln(-x) \text{ and } x < 0)$ |
| 5) $\text{not}(p \text{ or } \text{not } q) = \text{false}$ | |

- Escreva uma instrução de atribuição em PASCAL para cada uma das seguintes acções:
 - A variável lógica Imp4 é verdadeira se e só se o inteiro num NÃO FOR múltiplo de 4 ;
 - NumReal é verdadeira se e só se a equação $ax^2 + bx + c = 0$ NÃO TEM raízes complexas;
 - Sendo Car uma variável do tipo *char*, a variável Dig é verdadeira se e só se N representar um algarismo decimal;
 - A variável inteira N vale 1 se a variável inteira K for um algarismo par e 2 se K for ímpar;
 - Dado n, inteiro não-nulo, variável inteira Sinal toma o valor de -1 se n for negativo e 1 se n for positivo;
 - Sendo c um caracter, a variável LETRA toma o valor verdadeiro se e só se c for uma letra minúscula.

Exercício 3 (5,00)

- Um número primo é um natural que apenas é divisível por ele mesmo e pela unidade. Escreva e implemente em Pascal, uma função que recebe um inteiro e determine se é um número primo.
- Elabore e implemente um procedimento que recebe um número inteiro e , usando a função anterior, calcula e escreve todos os números primos *menores que seu dobro*.
- Explique que diferenças existem entre um *procedimento* e uma *função* em Pascal.

Exercício 4 (2,25)

Escreva um subprograma para pedir 3 valores reais ao utilizador e escrever qual é o valor máximo e qual é o valor mínimo de entre eles, fazendo o menor número de comparações possível.

Exercício 5 (7,50)

No que se segue, ao **tipo** que usar para vectores de inteiros deve dar o nome de *Tabela*.

- Escreva, e implemente em Pascal, um procedimento para ler e armazenar num vector do tipo *Tabela* uma sequência de números inteiros **não-nulos**, devolvendo o vector assim construído, de nome *vec*, bem como a sua dimensão útil, *dim*, para serem usados fora deste módulo.
Note que a quantidade de números a ler é, à partida, desconhecida e, ainda, que não deve ser ultrapassada a capacidade máxima declarada para um vector do tipo *Tabela*.

Sugestão: use um valor nulo como sentinela para assinalar o fim da introdução voluntária de números.

- Considere que são dados 2 vectores, v_1 e v_2 , de dimensões n_1 e n_2 . Escreva um procedimento ~~que receba estes dados e, sabendo que os vectores estão ordenados por ordem estritamente decrescente~~, devolva um novo vector que resulte de juntar os 2 vectores num só, mantendo a ordem.

- Suponha que está definido um procedimento de cabeçalho

```
procedure ord_vec(var v : Tabela; var n : 1..Maxint);
```

que, dado um vector v e o índice da última posição ocupada, n , ordena os elementos de v por ordem estritamente decrescente.

Escreva um **programa** em Pascal para recolher 2 vectores de inteiros, ordená-los e efectuar a sua fusão ordenada para um novo vector. Após esta fusão, o seu programa deve, ainda, escrever o resultado do vector resultante.

- Escreva, agora, o procedimento `ord_vec` em Pascal, escolhendo a estratégia de ordenamento que mais lhe agradar.
- Elabore e implemente um procedimento para inserir, num dado vector, *já ordenado por ordem decrescente*, um dado elemento, *num*. Atenção: a inserção deve ser efectuada de modo a não alterar a ordem estrita entre todos os elementos do vector.