

UNIVERSIDADE DA BEIRA INTERIOR

Programação – LEI

1º Semestre

Frequência 2 (8 valores)

1:15 h

2023/2024

1. [1.25 val]

Considere as seguintes declarações de variáveis:

int **V, *W, *X;

e que **sizeof(int) = 4** e **sizeof(int *) = 8**.

(esquema de bloco de memória)

	...	
100500	110172	W
	...	
100800	110160	X
	...	
110160	50	
110164	10	
110168	15	
110172	60	
110176	70	
110180	80	
	...	
110500	100800	V
	...	

Considerando os valores que constam no esquema de um bloco de memória que se encontra ao lado, determine o valor de cada uma das seguintes expressões:

- a) $X + 2$
- b) $V[0]$
- c) $W + 4$
- d) $(*V)[2]$
- e) $V + 2$
- f) $*V + 3$
- g) $**V + 4$
- h) $*W + 4$
- i) $*(W - 2)$
- j) $*(X + 3)$

NOTA: apresente todos os cálculos efetuados

2. [1.75 val]

Implemente um **subprograma** em C que dado um número inteiro positivo **N** ($N > 0$ é parâmetro do subprograma), **devolva** a quantidade de divisores de N. Exemplo: divisores de 10 = 4 ($\{1, 2, 5, 10\}$).

3. [2.50 val]

Considere o ficheiro de texto "Inteiros.txt", apenas com números inteiros. Implemente um **programa em C** que realize as seguintes ações (pela ordem indicada):

- 1) construir um array de 1 dimensão apenas com os números positivos não nulos (> 0) do ficheiro "Inteiros.txt", usando gestão de memória dinâmica, e
- 2) guarde os números pares do array no ficheiro de texto "Pares.txt" e os números ímpares do array no ficheiro "Impares.txt".

NOTA: Não é preciso verificar se há **ERRO** na **abertura dos ficheiros**, nem na **alocação de memória**

4. [2.50 val]

Implemente um **subprograma** em C que dado um array de 1 dimensão **X** com **N** ($N > 0$) números inteiros positivos (X e N são parâmetros do subprograma), **determine e devolva** como resultados o **menor** número e o **segundo menor** número do array **X** (se todos os números de X são iguais, então não há segundo menor número e deve devolver -1 para indicar este facto => segundo menor = -1).

NOTA: o array só pode ser percorrido uma única vez.