

Lista 8**Alocação dinâmica****Questão 1**

Qual é a diferença entre alocação estática e dinâmica?

Questão 2

Dados dois vetores x e y , ambos com n elementos, $1 \leq n \leq 100$, determinar o produto escalar desses vetores. Use alocação dinâmica de memória.

$$a = (a_1, a_2, \dots, a_n) \text{ e } b = (b_1, b_2, \dots, b_n)$$

$$a \cdot b = \sum_{i=1}^n a_i \times b_i$$

Questão 3

Qual o resultado da execução do programa abaixo? Ocorre algum erro?

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int *misterio(int n){
5     int i, *vet = malloc(n*sizeof(int));
6     vet[0] = 1;
7     for(i=1; i<n; i++) vet[i] = i*vet[i-1];
8     return vet;
9 }
10
11 int main(){
12     int i, n, *v;
13     printf("Digite n:");
14     scanf("%d", &n);
15     v = misterio(n);
16     for(i=0; i<n; i++) printf("%d\n", v[i]);
17     free(v);
18     return 0;
19 }
```

Questão 4

Faça um programa que leia a dimensão n de um vetor, em seguida aloca dinamicamente dois vetores do tipo `double` de dimensão n , faz a leitura de cada vetor e finalmente e imprime o resultado da soma dos dois vetores.

Questão 5

Faça uma função que recebe como parâmetro dois vetores de inteiros representando conjuntos de números inteiros, e devolve um outro vetor com o resultado da união dos dois conjuntos. O vetor resultante deve ser alocado dinamicamente.

```
1 int *uniao(int *v1, int n1, int *v2, int n2);
```

onde `n1` e `n2` indicam o número de elementos em `v1` e `v2` respectivamente.

Questão 6

Escreva uma função que recebe uma matriz M e depois verifica se esta é uma matriz triangular inferior.

```
1 int triangular(int *M, int l, int c);
```

onde `l` e `c` indicam o número de linhas e colunas da matriz M .

Questão 7

Faça uma função que leia todas as posições de uma matriz $l \times c$. Em seguida, mostra o índice da linha e o índice da coluna nas posições com números pares. No final, exiba o número de posições pares e ímpares.

```
1 int pares(int *M, int l, int c);
```

onde `l` e `c` indicam o número de linhas e colunas da matriz M .

Questão 8

Escreva uma função que receba 2 matrizes $n \times n$, mostre-as na tela e retorne outra matriz com a soma das duas matrizes. Use alocação dinâmica de memória.

```
1 int **somaMatriz(int *A, int l1, int c1, int *B, int l2, int c2);
```

onde `l1` e `c1` indicam o número de linhas e colunas da matriz A e `l2` e `c2` indicam o número de linhas e colunas da matriz B .

Relembrando...

$$\begin{array}{c} \text{A} \\ \begin{bmatrix} 0 & 1 & 0 & 2 & 3 \\ 0 & 1 & 0 & 2 & 3 \\ 0 & 1 & 0 & 2 & 3 \\ 0 & 1 & 0 & 2 & 3 \\ 0 & 1 & 0 & 2 & 3 \end{bmatrix} \end{array} + \begin{array}{c} \text{B} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 & 3 \end{bmatrix} \end{array} = \begin{array}{c} \text{C} \\ \begin{bmatrix} 0 & 1 & 0 & 4 & 3 \\ 1 & 2 & 1 & 3 & 4 \\ 0 & 1 & 0 & 2 & 3 \\ 2 & 3 & 2 & 4 & 5 \\ 3 & 4 & 3 & 5 & 6 \end{bmatrix} \end{array}$$

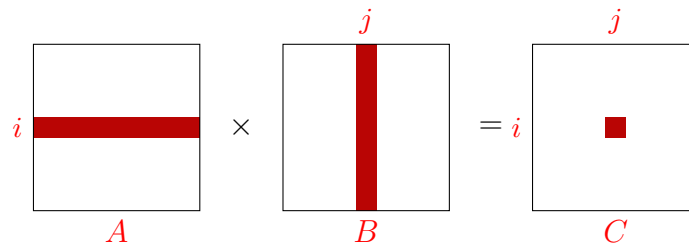
Questão 9

Escreva uma função que receba um valor inteiro n , e em seguida, aloca dinamicamente duas matrizes A e B em $\mathbb{R}^{n \times n}$ e retorna o resultado de $C = A \times B$

```
1 int** multiplicaM(int *A, int l1, int c1, int *B, int l2, int c2);
```

onde $l1$ e $c1$ indicam o número de linhas e colunas da matriz A e $l2$ e $c2$ indicam o número de linhas e colunas da matriz B .

Relembrando...



C_{ij} é o produto interno da linha i de A com a coluna j de B

$$C_{ij} = \sum_{k=1}^n A_{ik} B_{kj}$$

Questão 10

O que acontece quando o valor de $n = 4000000$?

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(){
5     int n, i;
6     printf("Digite o tamanho do vetor:");
7     scanf("%d", &n);
8     int v[n]; //Vetor alocado com tamanho n não pré-estabelecido
9
10    for(i=0; i<n; i++) v[i] = i;
11    for(i=0; i<n; i++) printf("%d\n", v[i]);
12
13    return 0;
14 }
```