



Universidade Federal do Maranhão - UFMA
Disciplina: Linguagem de Programação I – 2025.2
Professor: Geraldo Braz Junior, Tiago Bonini Borchartt
Nome: _____

Data: 05/11/2025

Primeira Avaliação

Observações: Todas as questões devem ser implementadas com funções.

1) [2pt] A função a seguir recebe dois vetores ordenados em ordem crescente ($v1$ e $v2$), um parâmetro n que indica o tamanho dos vetores $v1$ e $v2$, e deveria mesclar os elementos de $v1$ e $v2$ em um terceiro vetor ($v3$), com tamanho $n*2$, de forma que $v3$ também se mantivesse ordenado. Porém, ao receber os seguintes vetores $v1[] = \{1, 3, 5, 6, 8\}$ e $v2[] = \{2, 5, 7, 8, 9\}$, o vetor resultante $v3$ está ficando com os seguintes valores $\{1, 2, 3, 0, 0, 0, 0, 0, 0, 0\}$.

Corrija a função a seguir para que ela realize corretamente a mescla dos vetores, mantendo a ordenação.

```
void mesclarOrdenado(int n, int v1[], int v2[], int v3[]){
    for(int i=0, j=0, k=0; i<n*2; i++){
        v3[i] = 0;
        if(v1[j] < v2[k]){
            v3[i] = v1[j];
            j++;
        }
        else if(v1[j] > v2[k]){
            v3[i] = v2[k];
            k++;
        }
    }
}
```

2) [2pt] Uma função em C para multiplicar matrizes foi entregue para avaliação, mas a mesma está entregando valores errados. Segue abaixo a implementação da função. Sua missão é corrigir a mesma.

```
#define MAX 10

void multiplicaMatrizes(int A[MAX][MAX], int B[MAX][MAX], int C[MAX][MAX], int n, int m,
int p) {
    int i, j, k;

    for (i = 0; i < n; i++) {
        for (j = 0; j < p; j++) {
            for (k = 0; k < p; k++) {
                C[i][j] += A[k][i] * B[j][k];
            }
        }
    }
}
```

3) [3pt] Escreva um programa em C que leia uma frase (string) e separe as palavras contidas nela. A frase pode conter até 2048 caracteres, e cada palavra possui tamanho máximo de 100 caracteres. Na frase, as palavras estão separadas por espaço em branco, ponto e vírgula e ponto final.

4) [4pt] Continuando a questão 3, adicione uma função que conte o número de palavras repetidas na frase e em seguida imprima a palavra e a quantidade de repetições.

5) [4pt] Escreva um programa em C que gere uma matriz aleatória de números 0 e 1. A matriz possui dimensões 1024x1024. Depois o programa deve procurar nas linhas e colunas a maior sequência consecutiva de valores 1. Ao fim, o programa deve imprimir em tela qual o tamanho da maior sequência e em qual linha/coluna a sequência aconteceu.

3^a) ~~int conta_palavra(char vetor[2048])~~
~~{ int cont = 0;~~
~~for (int i = 0; i < 2048; i++)~~
~~{~~

3^a) int main()
 {

char frase[2048]

int i, cont = 0, ~~conta~~ palavra = 0;

for (i = 0; frase[i] != '\0'; i++)
 {

if (frase[i] == ' ' || frase[i] == ';' || frase[i] == '.')
 {

palavra++;

cont = 0;

else

{

cont++;

}

if (cont > 100)

3^a) int main()
 {

char frase[2048];

int i, cont = 0, palavra = 0, booleano = 1;

for (i = 0; frase[i] != '\0'; i++)
 {

if ((frase[i] == ' ' || frase[i] == ';' || frase[i] == '.') && booleano == 1)
 {

palavra++;

cont = 0;

}

→ não era para contar palavras, era para separar!


```

else
{
    cont++;
}
if (cont > 100)
{
    break;
}
return palabra;
}

```

12)

```

void mezclaOrdenada (int n, int V1[], int V2[], int V3[]) {
    for (int i=0, j=0, k=0; i < n*2; i++) {
        V3[i] = 0;
        if (V1[j] < V2[k]) {
            V3[i] = V1[j];
            j++;
        }
        else if (V1[j] >= V2[k]) {
            V3[i] = V2[k];
            k++;
        }
    }
}

```

```
printf("%d", maior);
```

```
if (maior < 0)
```

```
if (indice - ultimo_valor - maior < 0)
{
```

```
    indice - ultimo_valor -= 1;
```

```
    indice - ultimo_valor = 10 * 1 + (indice - ultimo_valor - maior);
}
```

```
printf("a maior sequencia esta entre [%d][%d] e [%d][%d]", indice - ultimo_valor,
       indice - ultimo_valor - maior, indice - ultimo_valor, indice - ultimo_valor + 1);
```

```
return 0;
}
```

```
#define MAX 10
```

```
2ª) void multiplicaMatrizes (int A[MAX][MAX], int B[MAX][MAX], int C[MAX][MAX],
                             int n, int m, int p) {
```

```
    int i, j, k;
```

```
    for (i = 0; i < n; i++) {
```

```
        for (j = 0; j < p; j++) {
```

```
            for (k = 0; k < p; k++) {
```

```
                C[i][j] += A[i][k] * B[k][j];
```

```
            }
```

```
        }
```

```
    }
```

```
}
```



```
5ª) #include <time.h>
#include <stdlib.h>
```

```
int main()
```

```
{
    srand(time(NULL));
```

```
int matriz[1024][1024], linha = 1024, coluna = 1024, i, j, maior = 1, numero;
```

```
int indice_ultimo_valor, indice_ultimo_valor2;
```

```
for (i = 0; i < linha; i++)
```

```
{
    for (j = 0; j < coluna; j++)
```

```
{
    matriz[i][j] = rand() % 2;
```

```
}
for (i = 0; i < linha; i++)
```

```
{
    for (j = 0; j < coluna; j++)
```

```
{
    if (matriz[i][j] == 1)
```

```
        numero++;
```

```
    }
    else
```

```
    {
        if (numero > maior)
```

```
        {
            maior = numero;
```

```
            numero = 0;
            indice_ultimo_valor = i;
```

```
            indice_ultimo_valor2 = j;
```

```
            numero = 0;
```

```
        }
```

```
    }
```