

Curso	Engenharia de Computação		Tipo de Avaliação Trabalho	
Unidade Curricular	Algoritmo e Lógica de Programação II		Semestre 2	Bimestre 1
Docente	Patrícia Salles Maturana de Souza			Nota
Nome do Aluno	Luis Gustavo da Silva Porto	R.A. 115976	Período 2º	

QUESTÕES

- 1) Deverá ser entregue no máximo até o dia **da prova**
- 2) Vocês irão fazer os seguintes exercícios no codeblocks, copiar e colar em um único documento do word e salvar em PDF para me enviar na **plataforma EAD** neste mesmo link deste trabalho:

1 - Fazer um programa para calcular as médias de cada disciplina utilizando função, onde cada disciplina é uma função diferente e na função principal mostre todas as médias;

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

// Matemática
float mmat() {
    float n1, n2, n3;
    printf("---- Sistema para calcular a média de 3 matérias. ---\n");
    printf("Digite as 3 notas de Matemática:\n");
    scanf("%f %f %f", &n1, &n2, &n3);
    return (n1 + n2 + n3) / 3;
}

// Português
float mpt() {
    float n1, n2, n3;
    printf("Digite as 3 notas de Português:\n");
    scanf("%f %f %f", &n1, &n2, &n3);
    return (n1 + n2 + n3) / 3;
}

// Química
float mquim() {
    float n1, n2, n3;
    printf("Digite as 3 notas de Química:\n");
    scanf("%f %f %f", &n1, &n2, &n3);
    return (n1 + n2 + n3) / 3;
}

// Médias
int main() {
    setlocale(LC_ALL, "portuguese");

    float mediaMat, mediaPt, mediaQuim;

    // Calcular as médias
    mediaMat = mmat();
    mediaPt = mpt();
    mediaQuim = mquim();

    // Exibir médias
    printf("-----\n");
    printf("Suas médias são:\n");
    printf("Matemática: %.2f\n", mediaMat);
    printf("Português: %.2f\n", mediaPt);
    printf("Química: %.2f\n", mediaQuim);

    return 0;
}
```

2 - Faça um programa usando a sua criatividade utilizando função.

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

// Calcular a média
float cm(float n1, float n2, float n3) {
    return (n1 + n2 + n3) / 3;
}

int main() {
    setlocale(LC_ALL, "portuguese");

    float n1, n2, n3, m;

    printf("--- Programa para calcular média de 3 números ---\n");
    printf("1 - Digite o primeiro número: ");
    scanf("%f", &n1);

    printf("2 - Digite o segundo número: ");
    scanf("%f", &n2);

    printf("3 - Digite o terceiro número: ");
    scanf("%f", &n3);

    // Chamando a função para calcular a média
    m = cm(n1, n2, n3);

    // Resultado
    printf("-----\n");
    printf("A média dos três números é: %.2f\n", m);

    return 0;
}
```

3 - Criar uma função recursiva para calcular o fatorial de um número.

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

// Função recursiva para calcular o fatorial de um número
int fatorial(int n) {
    if (n == 0 || n == 1) {
        return 1; // Caso base: fatorial de 0 ou 1 é 1
    } else {
        return n * fatorial(n - 1); // Chamada recursiva
    }
}

// Função principal
int main() {
    setlocale(LC_ALL, "portuguese");

    int num;

    printf("Digite um número para calcular o fatorial: ");
    scanf("%d", &num);

    if (num < 0) {
        printf("Fatorial não é definido para números negativos.\n");
    } else {
        printf("O fatorial de %d é: %d\n", num, fatorial(num));
    }

    return 0;
}
```

4 - Uma função é um conjunto de instruções criadas para cumprir uma tarefa particular e agrupadas numa unidade com um nome para referenciá-la. A existência de funções evita que o programador tenha de escrever o mesmo código repetidas vezes. Tendo essas informações, e lembrando o que aprendemos com vetores. Faça um programa que leia um vetor e mostre o maior e o menor valor utilizando funções.

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

int encontrar_maior(int vetor[], int tamanho) {
    int maior = vetor[0];
    for (int i = 1; i < tamanho; i++) {
        if (vetor[i] > maior) {
            maior = vetor[i];
        }
    }
    return maior;
}

int encontrar_menor(int vetor[], int tamanho) {
    int menor = vetor[0];
    for (int i = 1; i < tamanho; i++) {
        if (vetor[i] < menor) {
            menor = vetor[i];
        }
    }
    return menor;
}

int main() {
    setlocale(LC_ALL, "portuguese");

    int tamanho;

    printf("Digite o tamanho do vetor: ");
    scanf("%d", &tamanho);

    int vetor[tamanho];

    printf("Digite os elementos do vetor:\n");
    for (int i = 0; i < tamanho; i++) {
        printf("Elemento %d: ", i + 1);
        scanf("%d", &vetor[i]);
    }

    int maior = encontrar_maior(vetor, tamanho);
    int menor = encontrar_menor(vetor, tamanho);

    printf("Maior valor no vetor: %d\n", maior);
    printf("Menor valor no vetor: %d\n", menor);

    return 0;
}
```

5 - Funções facilitam ao programador quando há necessidade de se repetir o mesmo código. Sabendo disso, e lembrando o que aprendemos sobre matrizes. Faça um programa utilizando funções, uma função para a criação de matriz e outra função para mostrar a matriz e outra função para mostrar a transposta dela, ou seja, quando invertemos a linha pela coluna na impressão.

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

void criar_matriz(int matriz[10][10], int linhas, int colunas) {
    printf("Digite os elementos da matriz:\n");
    for (int i = 0; i < linhas; i++) {
        for (int j = 0; j < colunas; j++) {
            printf("Elemento [%d][%d]: ", i + 1, j + 1);
            scanf("%d", &matriz[i][j]);
        }
    }
}

void mostrar_matriz(int matriz[10][10], int linhas, int colunas) {
    printf("\nMatriz:\n");
    for (int i = 0; i < linhas; i++) {
        for (int j = 0; j < colunas; j++) {
            printf("%d ", matriz[i][j]);
        }
        printf("\n");
    }
}

void mostrar_transposta(int matriz[10][10], int linhas, int colunas) {
    printf("\nMatriz Transposta:\n");
    for (int i = 0; i < colunas; i++) {
        for (int j = 0; j < linhas; j++) {
            printf("%d ", matriz[j][i]);
        }
        printf("\n");
    }
}

int main() {
    setlocale(LC_ALL, "portuguese");

    int linhas, colunas;
    int matriz[10][10];

    printf("Digite o número de linhas da matriz: ");
    scanf("%d", &linhas);
    printf("Digite o número de colunas da matriz: ");
    scanf("%d", &colunas);

    criar_matriz(matriz, linhas, colunas);

    mostrar_matriz(matriz, linhas, colunas);

    mostrar_transposta(matriz, linhas, colunas);

    return 0;
}
```

6 - Faça um programa utilizando função para somar dois números;

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

int somar(int a, int b) {
    return a + b;
}

int main() {
    setlocale(LC_ALL, "portuguese");
    int n1, n2, resultado;

    printf("Digite o primeiro número: ");
    scanf("%d", &n1);

    printf("Digite o segundo número: ");
    scanf("%d", &n2);

    resultado = somar(n1, n2);

    // Exibindo o resultado
    printf("A soma de %d e %d é: %d\n", n1, n2, resultado);

    return 0;
}
```

7 - Faça um programa utilizando função para subtrair dois números;

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

// Função para subtrair dois números
int sub(int a, int b) {
    return a - b;
}

// Função principal
int main() {
    setlocale(LC_ALL, "portuguese");
    int n1, n2, resultado;

    // Entrada dos dois números
    printf("Digite o primeiro número: ");
    scanf("%d", &n1);

    printf("Digite o segundo número: ");
    scanf("%d", &n2);

    // Chamando a função de subtração
    resultado = sub(n1, n2);

    // Exibindo o resultado
    printf("A subtração de %d e %d é: %d\n", n1, n2, resultado);

    return 0;
}
```

8 - Faça um programa utilizando função para elevar ao cubo um número;

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

float cubo(float n) {
    return n * n * n;
}

int main() {
    setlocale(LC_ALL, "portuguese");
    float n, resultado;

    printf("Digite um número: ");
    scanf("%f", &n);

    resultado = cubo(n);

    printf("O número %.2f ao cubo é: %.2f\n", n, resultado);

    return 0;
}
```

9 - Faça um programa que peça duas matrizes e calcule a adição destas, isso utilizando 2 funções;

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>
#define LINHAS 3
#define COLUNAS 3

void ler_matriz(int m[LINHAS][COLUNAS]) {
    for (int i = 0; i < LINHAS; i++) {
        for (int j = 0; j < COLUNAS; j++) {
            printf("Digite o elemento [%d][%d]: ", i + 1, j + 1);
            scanf("%d", &m[i][j]);
        }
    }
}

void somar_matrizes(int m1[LINHAS][COLUNAS], int m2[LINHAS][COLUNAS], int resultado[LINHAS][COLUNAS]) {
    for (int i = 0; i < LINHAS; i++) {
        for (int j = 0; j < COLUNAS; j++) {
            resultado[i][j] = m1[i][j] + m2[i][j];
        }
    }
}

int main() {
    setlocale(LC_ALL, "portuguese");
    int m1[LINHAS][COLUNAS], m2[LINHAS][COLUNAS], resultado[LINHAS][COLUNAS];

    printf("Digite os elementos da primeira matriz:\n");
    ler_matriz(m1);

    printf("Digite os elementos da segunda matriz:\n");
    ler_matriz(m2);

    somar_matrizes(m1, m2, resultado);

    printf("Resultado da soma das matrizes:\n");
    for (int i = 0; i < LINHAS; i++) {
        for (int j = 0; j < COLUNAS; j++) {
            printf("%d ", resultado[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

10 - Faça um programa usando a sua criatividade que uma função chame a outra;

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>

void f1() {
    printf("Esta mensagem foi impressa pela função 'F1'.\n");
}

void f2() {
    printf("A função 'F2' está chamando a função 'F1'.\n");
    f1();
}

int main() {
    setlocale(LC_ALL, "portuguese");
    f2();

    return 0;
}
```

11 - Faça um programa usando a sua criatividade utilizando matriz e função;

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>
#define LINHAS 3
#define COLUNAS 3

void ler_matriz(int m[LINHAS][COLUNAS]) {
    for (int i = 0; i < LINHAS; i++) {
        for (int j = 0; j < COLUNAS; j++) {
            printf("Digite o elemento [%d][%d]: ", i + 1, j + 1);
            scanf("%d", &m[i][j]);
        }
    }
}

int somar(int m[LINHAS][COLUNAS]) {
    int soma = 0;
    for (int i = 0; i < LINHAS; i++) {
        for (int j = 0; j < COLUNAS; j++) {
            soma += m[i][j];
        }
    }
    return soma;
}

int main() {
    setlocale(LC_ALL, "portuguese");
    int m[LINHAS][COLUNAS];

    printf("Digite os elementos da matriz 3x3:\n");
    ler_matriz(m);

    int soma = somar(m);

    printf("A soma dos elementos da matriz é: %d\n", soma);

    return 0;
}
```

12 - Faça um programa usando a sua criatividade utilizando vetores e função.

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>
#define t 5

void ler_vetor(int v[t]) {
    for (int i = 0; i < t; i++) {
        printf("Digite o elemento %d: ", i + 1);
        scanf("%d", &v[i]);
    }
}

int somar(int v[t]) {
    int soma = 0;
    for (int i = 0; i < t; i++) {
        soma += v[i];
    }
    return soma;
}

int main() {
    int v[t];

    printf("Digite %d números inteiros:\n", t);
    ler_vetor(v);

    int soma = somar(v);
    printf("A soma dos elementos do vetor é: %d\n", soma);

    return 0;
}
```