

<b>Trabalho</b>	<b>Campus:</b> Campo Grande	<b>Data:</b> 26 / 05 / 24	<b>Turma:</b> 3006
<b>Disciplina:</b> Introdução à Programação Estruturada em C			<b>Professor:</b> Alessandro Calin
<b>Matrícula:</b> 202402367625		<b>Nome:</b> Júlio César Machado Fernandes	

Para cada programa você deve colocar o código e o print do programa rodando.  
ESCOLHA APENAS 6 DOS PROGRAMAS ABAIXO, OU SEJA, VOCÊ PODE ELIMINAR UM.

Forma de entrega:

Armazene o projeto (documento com os códigos e prints) em um repositório no GIT.

Anexar a documentação do projeto (PDF) no GIT.

Compartilhe o link do repositório do GIT com a apresentação no SAVA

1) Faça um programa que solicite o número de elementos de vetor, solicite os elementos e armazeno-os no vetor, e imprima a quantidade de elementos pares e ímpares

### Eliminado

2) Desenvolver um algoritmo que leia dez números inteiro e verifique e imprima quantos são divisíveis por 5 e por 3 ao mesmo tempo.

```
#include <stdio.h>
int main()
{
    int arr[10],divisiveis15;
    divisiveis15 = 0;
    for (int i = 0; i < 10; i++) {
        printf("\nDigite o valor %d: ",i+1);
        scanf("%d",&arr[i]);
        if (arr[i]%15 == 0) {
            divisiveis15 ++;
        }
    }
    printf("\nDos 10 valores do vetor %d são divisíveis por 5 e 3 ao mesmo tempo.",divisiveis15);
    return 0;
}
```

```
Digite o valor 1: 15
Digite o valor 2: 15
Digite o valor 3: 30
Digite o valor 4: 45
Digite o valor 5: 60
Digite o valor 6: 10
Digite o valor 7: 12
Digite o valor 8: 13
Digite o valor 9: 40
Digite o valor 10: 19
Dos 10 valores do vetor 5 são divisíveis por 5 e 3 ao mesmo tempo.
...Program finished with exit code 0
Press ENTER to exit console.
```

3) Fazer um programa que faz uma pesquisa com pessoas entre 18 e 80 anos. O programa deve solicitar a quantidade de pessoas a ser entrevistadas. Armazenar a idade dessas pessoas em um vetor e imprimir quantas pessoas de cada faixa etária foram entrevistadas de acordo com a tabela abaixo:

$\geq 18$  e  $< 35$        $\rightarrow$     jovem  
 $\geq 35$  e  $< 65$     $\rightarrow$     adulto  
 $\geq 65$              $\rightarrow$     idoso

O programa deve imprimir o quantitativo de jovens, adultos e idosos. Desta forma essas variáveis que irão contar deverão ser inicializadas com zero.

```
#include <stdio.h>

int main()
{
    int qtd,jovem,adulto,idoso;
    jovem = adulto = idoso = 0;
    printf("Digite a quantidade de pessoas entrevistadas: ");
    scanf("%d",&qtd);
    int idades[qtd];
    for (int i = 0; i < qtd; i++) {
        printf("Digite a idade da pessoa %d: ",i+1);
        scanf("%d",&idades[i]);
        int idade = idades[i];
        if (idade >= 18 && idade < 35) {
            jovem++;
        } else if (idade >= 35 && idade < 65) {
            adulto++;
        } else if (idade >= 65) {
            idoso++;
        }
    }
    printf("\nJovens: %d \nAdultos: %d \nIdosos: %d",jovem,adulto,idoso);
    return 0;
}
```

```
Digite a quantidade de pessoas entrevistadas: 3
Digite a idade da pessoa 1: 18
Digite a idade da pessoa 2: 34
Digite a idade da pessoa 3: 67

Jovens: 2
Adultos: 0
Idosos: 1

...Program finished with exit code 0
Press ENTER to exit console.
```

4) Faça um programa que leia 10 números inteiros, armazene-os em um vetor, solicite um valor de referência inteiro e:

- imprima os números** do vetor que são maiores que o valor referência
- retorne **quantas vezes** o valor de referência aparece no vetor

```
#include <stdio.h>

int main() {

    int
    arr[10],referencia,qtdReferencia,maiorReferencia;

    qtdReferencia = maiorReferencia = 0;

    for (int i = 0; i < 10; i++) {

        printf("\nDigite o %dº valor: ",i+1);
```

```

        scanf("%d",&arr[i]);

    }

    printf("\nDigite um valor de referência: ");
    scanf("%d",&referencia);

    for (int i = 0; i < 10; i++) {

        if (arr[i] == referencia) {
            qtdReferencia++;
        }

        if (arr[i] > referencia ) {
            maiorReferencia++;
        }

    }

    int
    arrMaiorRef[maiorReferencia],indiceInicial;
    indiceInicial = 0;

    for (int i = 0; i < 10; i++) {

        if (arr[i] > referencia) {

            for (int j = indiceInicial; j <
maiorReferencia ; j++) {
                arrMaiorRef[j] = arr[i];
            }

            indiceInicial++;

        }

    }

    printf("\nO número: %d aparece %d
vezes",referencia,qtdReferencia);

    printf("\nnúmeros maiores que %d:
",referencia);

    for (int i = 0; i < maiorReferencia; i++) {
        printf("%d,",arrMaiorRef[i]);
    }

```

```

Digite o 1° valor: 1
Digite o 2° valor: 2
Digite o 3° valor: 3
Digite o 4° valor: 4
Digite o 5° valor: 5
Digite o 6° valor: 6
Digite o 7° valor: 6
Digite o 8° valor: 7
Digite o 9° valor: 8
Digite o 10° valor: 9
Digite um valor de referência: 6
O número: 6 aparece 2 vezes
números maiores que 6: 7,8,9,
...Program finished with exit code 0
Press ENTER to exit console.

```

5) Fazer um programa que crie um Struct com a matricula, nome, nota da AV, simulado1 e simulado 2 de um aluno. O programa para cada aluno deve apresentar os dados lidos destes alunos na Struct. (Fonte de estudo o programa de Struct da Folha).

```
#include <stdio.h>
#include <string.h>

typedef struct {

    int matricula;
    char nome[40];
    float notaAv;
    float simulado1;
    float simulado2;

}aluno;

int main() {

    int qtd;

    printf("\nDigite a quantidade e alunos para cadastrar: ");
    scanf("%d",&qtd);

    aluno arr[qtd];

    for (int i = 0; i < qtd; i++){

        printf("\nDigite a matrícula do %dº aluno: ",i+1);
        scanf("%d",&arr[i].matricula);

        printf("\nDigite o nome do %dº aluno: ",i+1);
        scanf("%s",arr[i].nome);

        printf("\nDigite a nota da AV de %s: ",arr[i].nome);
        scanf("%f",&arr[i].notaAv);

        printf("\nDigite a nota do Simulado 1 de %s: ",arr[i].nome);
```

```

scanf("%f",&arr[i].simulado1);

printf("\nDigite a nota do Simulado 2 de %s: ",arr[i].nome);
scanf("%f",&arr[i].simulado2);


printf("\n_____");

}

printf("\nAlunos cadastrados: ");

for (int i = 0; i < qtd; i++) {

    printf("\nNome: %s",arr[i].nome);
    printf("\nMatrícula: %d",arr[i].matricula);
    printf("\nNota AV: %f",arr[i].notaAv);
    printf("\nNota Simulado 1: %f",arr[i].simulado1);
    printf("\nNota Simulado 2: %f",arr[i].simulado2);

    printf("\n_____");

}

}

```

```

Digite a quantidade e alunos para cadastrar: 2
Digite a matrícula do 1º aluno: 14
Digite o nome do 1º aluno: Julio
Digite a nota da AV de Julio: 10
Digite a nota do Simulado 1 de Julio: 1
Digite a nota do Simulado 2 de Julio: 1

```

```

_____  

Digite a matrícula do 2º aluno: 13
Digite o nome do 2º aluno: Joao
Digite a nota da AV de Joao: 6
Digite a nota do Simulado 1 de Joao: 0.5
Digite a nota do Simulado 2 de Joao: 0.3

```

```

_____  

Alunos cadastrados:  

Nome: Julio  

Matrícula: 14  

Nota AV: 10.000000  

Nota Simulado 1: 1.000000  

Nota Simulado 2: 1.000000

```

```

_____  

Nome: Joao  

Matrícula: 13  

Nota AV: 6.000000  

Nota Simulado 1: 0.500000  

Nota Simulado 2: 0.300000

```

```

_____  

...Program finished with exit code 0  

Press ENTER to exit console.

```

6) Fazer um programa que solicite ao usuário na main um número para calcular a tabuada. Crie uma função que faz o cálculo da tabuada. (Fonte de estudo slide 56). OBS: a única diferença é que a geração da tabuada tem que ser em função.

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

```

int calcularTabuada(int numero) {

    int multiplicacao;

    for (int i = 1; i <= 10; i++) {
        multiplicacao = numero * i;
        printf("\n%d x %d = %d",numero,i,multiplicacao);
    }

}

int main(){

    int num;

    printf("\nDigite um número para a tabuada desejada: ");
    scanf("%d",&num);

    calcularTabuada(num);

    return 0;

}

```

```

Digite um número para a tabuada desejada: 2

```

```

2 x 1 = 2
2 x 2 = 4
2 x 3 = 6
2 x 4 = 8
2 x 5 = 10
2 x 6 = 12
2 x 7 = 14
2 x 8 = 16
2 x 9 = 18
2 x 10 = 20

```

```

...Program finished with exit code 0
Press ENTER to exit console.

```

7) Criar um programa em linguagem C que calcule o Índice de Massa Corporal (IMC) de um indivíduo e classifique sua situação de acordo com os seguintes parâmetros:

- **Abaixo do peso:** IMC menor que 18,5
- **Peso normal:** IMC entre 18,5 e 24,9
- **Sobrepeso:** IMC entre 25 e 29,9
- **Obesidade grau I:** IMC entre 30 e 34,9
- **Obesidade grau II:** IMC entre 35 e 39,9
- **Obesidade grau III:** IMC maior ou igual a 40

O programa deve solicitar ao usuário que digite seu peso em quilogramas e sua altura em metros. Em seguida, o programa deve calcular o IMC utilizando a seguinte fórmula:

$$\text{IMC} = \text{peso} / (\text{altura} * \text{altura})$$

```

#include <stdio.h>

int calcularImc(float weight, float height) {

    float imc = weight / (height * height);

    if (imc < 18.5) {
        printf("\nAltura: %.2f\nPeso: %2.f\nIMC: %.2f\nCondição: Abaixo do
peso",height,weight,imc);
        return 0;
    }

    if (imc < 24.9) {
        printf("\nAltura: %.2f\nPeso: %2.f\nIMC: %.2f\nCondição: Peso normal",height,weight,imc);
        return 0;
    }

    if (imc < 29.9) {
        printf("\nAltura: %.2f\nPeso: %2.f\nIMC: %.2f\nCondição: Sobrepeso",height,weight,imc);
        return 0;
    }

    if (imc < 34.9) {
        printf("\nAltura: %.2f\nPeso: %2.f\nIMC: %.2f\nCondição: Obesidade grau
I",height,weight,imc);
        return 0;
    }

    if (imc < 39.9) {
        printf("\nAltura: %.2f\nPeso: %2.f\nIMC: %.2f\nCondição: Obesidade grau
II",height,weight,imc);
        return 0;
    }

    if (imc >= 40.0) {
        printf("\nAltura: %.2f\nPeso: %2.f\nIMC: %.2f\nCondição: Obesidade grau
III",height,weight,imc);
        return 0;
    }

}

int main()
{
    float peso,altura;

    printf("\nDigite sua altura em (metros): ");
    scanf("%f",&altura);

    printf("\nDigite seu peso em (quilogramas): ");
    scanf("%f",&peso);

    printf ("_____");

    calcularImc(peso,altura);
}

```

```
    return 0;  
}
```

```
Digite sua altura em (metros): 1.78  
Digite seu peso em (quilogramas): 76.0  
-----  
Altura: 1.78  
Peso: 76  
IMC: 23.99  
Condição: Peso normal  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```