

LISTA 3

TEÓRICAS:

1 – Para o paradigma de linguagem estruturada, em especial, a Linguagem C, o que é uma subrotina/módulo/função? Para que ela serve?

Uma função é um bloco de código reutilizável e serve para facilitar a vida do programador. Por exemplo, usar o .h cria um bloco de código que pode ser chamado em qualquer outro código.

2 – O que é uma variável local? Dê um exemplo em código.

É uma variável criada dentro de qualquer função que só pode ser usada dentro dessa função específica em que ela foi criada.

exemplo:

```
#include <stdio.h>
#include "bibli_4.h"

int main()
{
    int a=questao_4();

    printf("resultado: %i",a);

    return 0;
}
```

3 – O que é uma variável global? Dê um exemplo em código.

É um variável criada fora de todas as funções, podendo ser utilizada em qualquer momento do código.

```
#include <stdio.h>

int x_global = 10; // Variável global

void exemploGlobal() {
    printf("Valor da variável global: %d\n", x_global);
}

int main() {
```

```
    exemploGlobal();  
    return 0;  
}
```

PRÁTICAS:

EX4:

BIBLIOTECA

```
#ifndef BIBLI_4_H  
#define BIBLI_4_H  
  
int questao_4()  
{  
  
    int num = 0;  
    printf("Digite um numero: ");  
    scanf("%i", &num);  
  
    if (num % 2 == 0)  
    {  
        return 0;  
    }else  
    {  
        return 1;  
    }  
  
}  
  
#endif
```

CHAMANDO A BIBLIOTECA:

```
#include <stdio.h>  
#include "bibli_4.h"
```

```
int main()
{
    int a=questao_4();

    printf("resultado: %i",a);

    return 0;
}
```

EX5:

Biblioteca

```
#ifndef BIBLI_5_H_
#define BIBLI_5_H_

int x, y; // var global

int soma()
{
    return x + y;
}

int sub()
{
    return x - y;
}

int mult()
{
    return x * y;
}

float divisao()
{
    if (y != 0)
    {
        return (float)x / y;
    }
    else
    {
        printf("invalido: divisao por zero!");
    }
}
```

```

    }
}

#endif

```

programa principal para ler e imprimir os resultados

```

#include <stdio.h>
#include "bibli_5.h"

int main()
{
    // entrada de dados
    printf("digite o valor x: ");
    scanf("%i", &x);

    printf("\ndigite o valor y: ");
    scanf("%i", &y);

    // printando os resultados

    printf("soma: %i", soma());
    printf("\nsubtracao: %i", sub());
    printf("\nmultiplicacao: %i", mult());
    printf("\ndivisao: %.2f", divisao());

    return 0;
}

```

EX6:

```

#ifndef BIBLI_CAL_H
#define BIBLI_CAL_H

#include <stdio.h>

int hora;

int saudacao()
{
    if (hora > 6.1 && hora < 12.0)

```

```

{
    printf("\nBom dia");
}
else if (hora > 12.1 && hora < 18)
{
    printf("\nBoa tarde");
}
else if ((hora > 18 && hora < 23) || (hora >= 0 && hora <= 5))
{
    printf("\nBoa noite");
}
else
{
    printf("\n valor invalido");
}
}

#endif

```

programa principal

```

#include <stdio.h>
#include "bibli_cal.h"

int main()
{
    printf("digite hora (0-23): ");
    scanf("%i", &hora);

    saudacao();

    return 0;
}

```

EX7:

```

#ifndef BIBLI_7FIBONACC_H
#define BIBLI_7FIBONACC_H

int n;
int fibonacci()

```

```

{

    long long calcularFibonacci(int n);

}

#endif

```

programa principal

```

#include <stdio.h>
#include "bibli_7fibonacci.h"

long long calcularFibonacci(int n) {
    if (n <= 1) {
        return n;
    }

    long long a = 0;
    long long b = 1;
    long long resultado = 0;

    for (int i = 2; i <= n; ++i) {
        resultado = a + b;
        a = b;
        b = resultado;
    }

    return resultado;
}

int main()
{
    printf("Digite o valor de N: ");
    scanf("%d", &n);

    // Calcula o enésimo termo da sequência de Fibonacci
    long long resultado = calcularFibonacci(n);

    printf("O %dº termo da sequência de Fibonacci é: %lld\n", n,
resultado);

    return 0;
}

```

EX8:

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

int main()
{
    // entrada das notas
    printf("Digite nota 1: ");
    scanf("%f", &num1);

    printf("Digite nota 2: ");
    scanf("%f", &num2);

    printf("Digite nota 3: ");
    scanf("%f", &num3);

    // escolha do tipo de media e impressao da media escolhida
    char tipo_media;
    printf("digite 'p' para media ponderada / 'a' para media
aritmética: ");
    scanf(" %c", &tipo_media);

    if (tipo_media == 'p')
    {
        printf("media ponderada: %f", media_ponderada());
    }
    else if (tipo_media == 'a')
    {
        printf("media aritmética: %f", media_aritmetica());
    }
    else
    {
        printf("invalido");
    }
    return 0;
}
```

BIBLIOTECA CRIADA:

```
#ifndef CALCULO_MEDIA_H
#define CALCULO_MEDIA_H
```

```

float num1, num2, num3, p, a;
float media_aritmetica()
{
    return (num1 + num2 + num3)/3;
}
float media_ponderada()
{
    return ((num1*5)+(num2*3)+(num3*2))/10;
}

#endif

```

EX9:

```

#include <stdio.h>

typedef struct
{
    int matricula;
    float notas[3];
} Aluno;

Aluno Turma[5]; // Variável global
int alunoAtual = 0; // Variável global para controlar o aluno atual

void LeAluno()
{
    for (int i = 0; i < 5; i++)
    {
        printf("Digite a matrícula do aluno %d: ", i + 1);
        scanf("%i", &Turma[i].matricula);

        for (int j = 0; j < 3; j++)
        {
            printf("Digite a nota %i do aluno: ", j + 1);
            scanf("%f", &Turma[i].notas[j]);
        }
    }
}

```



```

void ImprimeTurma()
{
    for (int i = 0; i < 5; i++)
    {
        printf("Aluno %d - Matricula: %i\n", i + 1,
Turma[i].matricula);
        for (int j = 0; j < 3; j++)
        {
            printf("Nota %i: %.2f\n", j + 1, Turma[i].notas[j]);
        }
    }
}

float CalculaMediaAluno()
{
    float soma = 0.0;

    // Calcula a média do aluno atual
    for (int j = 0; j < 3; j++)
    {
        soma += Turma[alunoAtual].notas[j];
    }

    alunoAtual++; // Passa para o próximo aluno da turma
    return soma / 3.0;
}

int main()
{
    float media;
    int i;

    for (i = 0; i < 5; i++)
    {
        LeAluno(); // Função que lê os dados de cada aluno
    }

    ImprimeTurma(); // Função que imprime os dados de todos os alunos

    for (i = 0; i < 5; i++)
    {
        media = CalculaMediaAluno(); // Função
que calcula a média das notas de um aluno
    }
}

```

```
        printf("Aluno %d - Media = %.2f\n", i + 1, media); // Corrigido
para exibir o número do aluno e sua média
    }

    return 0;
}
```