

LISTA 5 – 03/05

Questão 1

A recursividade é uma função que chama, ao fim, ela mesma, assim diminuindo a criação de variáveis para um código. O retorno da função sob ela mesma faz com que o programa entenda o código retornando ao final a mesma função do início, como parâmetro de parada.

Questão 2

Pilha de memória é onde são armazenados os dados para a conclusão da função, quando cada uma das ações do código, que corresponde a determinada memória é executada, essa memória é 'excluída' e seu resultado transmitido para que o próximo espaço na pilha será executado até que segue a conclusão. Na recursividade a memória é utilizada para preservar os valores anteriores dos mesmos no retorno dos níveis de recursão.

Questão 3

```
#include <stdio.h>
int multiplica(int n1, int n2) {
    if (n2== 0)
        return n2;
    else
        return n1 + multiplica(n1,n2-1);
}
int main(){
    int n1, n2, resul;
    printf("Digite dois valores: ");
    scanf("%i %i", &n1, &n2);
    resul=multiplica(n1,n2);
    printf("Resultado: %i",resul);
    return 0;
}
```

Questão 4

```
#include <stdio.h>
int divide(int n1, int n2) {
    if (n2 == 1)
        return n1;
    else if (n1<n2)
        return 0;
    else
        return 1 + divide(n1-n2,n2);
}
int main(){
    int n1, n2, resultado;
```

```

printf("Digite dois valores: ");
scanf("%i %i", &n1, &n2);
resultado=divide(n1,n2);
printf("Resultado: %i",resultado);
return 0;
}

```

Questão 5 e 6

```

#include <stdio.h>
int denominador(int num){
    int resp=1;
    if(num>1)
    {resp = num*denominador(num-1);}
    else{resp *= num;}
    return resp;
}
float soma(int n1, int i){
    float var=0;
    float total, den;
    int n=1;
    den = denominador((i-1)*2);
    if(i%2==0){n=-1;}
    if (i>1){total=(1/den)*n; var+= total+soma(n1,i-1);}
    else var=1;
    return var;
}
int main(void){
    int n1;
    float resultado;
    printf("Digite a quantidade de termos: ");
    scanf("%i", &n1);
    resultado=soma(n1,n1);
    printf("Resultado igual a: %.2f",resultado);
    return 0;
}

```

Questão 7

```

#include <stdio.h>
int soma(int n) {
    if(n == 1) {
        return 1;
    }
    return n + soma(n - 1);
}
int main(){
    int n,resultado;

```

```

printf("Digite um valor: ");
scanf("%i", &n);
resultado=soma(n);
printf("Soma dos valores entre 0 e %i: %i",n,resultado);
}

```

Questão 8

```

#include <stdio.h>
float serie(float n){
    if (n==1)
        return 1;
    else {
        return (1/(n)+serie(n-1));
    }
}
int main(){
    float n;
    int num=1, den;
    printf("Digite a quantidade de termos: ");
    scanf("%f",&n);
    for (den=1;den<=n;den++){
        printf("%i/%i | ",num,den);
    }
    printf("\nSoma dos termos: %f",serie(n));
    return 0;
}

```

Questão 9

```

#include <stdio.h>
int mdc(int n1, int n2){
    if(n2 == 0)
        return n1;
    else
        return mdc(n2, n1%n2);
}
int teste(int n1, int n2){
    while(n2!=0){
        int resto=n1%n2;
        n1=n2;
        n2=resto;
    }
    return n1;
}
int main() {
    int n1,n2,resultado;
    printf("Digite dois valores: ");
    scanf("%i%i",&n1,&n2);
}

```

```
    resultado=mdc(n1,n2);
    printf("mdc(%i,%i) = %i\n",n1,n2,resultado);
    return 0;
}
```

Questão 10

```
#include <stdio.h>
int fib(int n){
    if (n==0)
        return 0;
    else if (n==1)
        return 1;
    else
        return fib(n-1)+fib(n-2);
}
int main(){
    int n, resultado;
    printf("Digite o número da sequência: ");
    scanf("%i", &n);
    resultado=fib(n);
    printf("Resultado no termo %i: %i",n,resultado);
}
```