

Estruturas de controle – repetição / laços

1. Escreva um programa que mostre os números de 1 até 10, um número por linha.

```
#include <stdio.h>

int main() {

    int i;

    for(i = 1; i <= 10; i++){
        printf("%d\n", i);
    }

    return 0;
}
```

2. Escreva um programa que mostre na tela uma contagem regressiva de 10 até 0 para o lançamento de um foguete. Após o término da contagem regressiva, deve ser apresentada a palavra "Fogo!".

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

int main() {

    int i;

    for(i = 10; i >= 0; i--){
        printf("%d\n", i);
        Sleep(1000);
    }

    printf("Fogo!");

    return 0;
}
```

3. Escreva um programa que mostre os números múltiplos de 5, no intervalo de 1 até 100.

```
#include <stdio.h>

int main() {

    int i;

    for(i = 5; i <= 100; i = i + 5){
        printf("%d ", i);
    }

    return 0;
}
```

4. Escreva um programa que mostre na tela a soma obtida dos 100 primeiros números inteiros (1 + 2 + 3 + 4 + 5 + ... + 100). **Resultado:** 5050.

```
#include <stdio.h>

int main() {

    int i, soma;

    soma = 0;

    for(i = 1; i <= 100; i++){
        soma = soma + i;
    }

    printf("Resultado: %d", soma);

    return 0;
}
```

5. Escreva um programa que mostre todos os números pares existentes entre 1 e um número lido do teclado, bem como a quantidade de números apresentados.

```
#include <stdio.h>

int main() {

    int num, i, quant;

    quant = 0;

    printf("Informe número: ");
    scanf("%d", &num);

    for(i = 1; i <= num; i++){
```

```

        if(i % 2 == 0){
            printf("%d ", i);
            quant++;
        }
    }

    printf("\nQuantidade de números apresentados: %d", quant);

    return 0;
}

```

6. Escreva um programa que some os números fornecidos pelo usuário até que o número lido seja igual a zero. Ao final, mostre o resultado da soma.

```

#include <stdio.h>

int main() {

    int num, soma;
    soma = 0;

    do{
        printf("Informe número: ");
        scanf("%d", &num);

        soma = soma + num;
    }while(num != 0);

    printf("Resultado: %d", soma);

    return 0;
}

```

7. Escreva um programa que apresente na tela a tabuada de um número lido do teclado. Por exemplo, considerando que o número lido é 2, o formato da apresentação deve ser o seguinte:

```

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

```

```

#include <stdio.h>

```

```

int main() {

    int tabuada, i;

    printf("Informe tabuada desejada: ");
    scanf("%d", &tabuada);

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

    return 0;
}

```

8. Escreva um programa que mostre a tabuada de um número inteiro qualquer que será digitado pelo usuário, mas a tabuada não deve necessariamente iniciar em 1 e terminar em 10, ou seja, o valor inicial e final devem ser informados também pelo usuário, conforme mostrado no exemplo a seguir:

```

Mostrar a tabuada de: 5
Começar por: 4
Terminar em: 7

```

```

5 X 4 = 20
5 X 5 = 25
5 X 6 = 30
5 X 7 = 35

```

```

#include <stdio.h>

int main() {

    int tabuada, comecar, terminar, i;

    printf("Mostrar tabuada de: ");
    scanf("%d", &tabuada);

    printf("Começar por: ");
    scanf("%d", &comecar);

    printf("Terminar em: ");
    scanf("%d", &terminar);

    for(i = comecar; i <= terminar; i++){
        printf("\n%d x %d = %d", tabuada, i, tabuada * i);
    }

    return 0;
}

```

```
}
```

9. Escreva um programa que calcule e mostre na tela uma tabela de valores de temperatura em Celsius e Farenheit. A faixa de temperaturas que deve ser mostrada é de 0 a 100 graus Celsius, em intervalos de 5 graus.

```
#include <stdio.h>

int main() {

    int i;

    for(i = 0; i <= 100; i = i + 5){
        printf("\nCelsius: %d, Farenheit: %.1f", i, i * 1.8 + 32);
    }

    return 0;
}
```

10. Escreva um programa que calcule o fatorial de um número positivo qualquer. Por exemplo, se o número for 5, o resultado será 120 ($5 * 4 * 3 * 2 * 1 = 120$). É importante observar que o fatorial do número 0 é igual a 1.

```
#include <stdio.h>

int main() {

    int fatorial, num, i;

    fatorial = 1;

    printf("Informe número: ");
    scanf("%d", &num);

    for(i = 2; i <= num; i++){
        fatorial *= i;
    }

    printf("O fatorial de %d é %d", num, fatorial);

    return 0;
}
```

11. Escreva um programa que escreva os 10 primeiros termos da série 1, 3, 9, 27, ...

```
#include <stdio.h>

int main() {
```

```

int i, num;

num = 3;

for(i = 1; i <= 10; i++){
    printf("%d ", num);
    num = num * 3;
}

return 0;
}

```

12. Escreva um programa que mostre o resultado da série: $1/100 + 2/99 + 3/98 + 4/97 + \dots + 100/1$.

Resultado: 423.93.

```

#include <stdio.h>

int main() {

    int i;
    float numerador, denominador, soma;

    numerador = 1;
    denominador = 100;
    soma = 0;

    for(i = 1; i <= 100; i++){
        soma = soma + numerador / denominador;
        numerador++;
        denominador--;
    }

    printf("Resultado: %.2f", soma);

    return 0;
}

```

13. Escreva um programa que mostre o resultado dos primeiros 20 termos da série: $1/1 + 3/2 + 5/4 + 7/8 + \dots$ **Resultado:** 6.00.

```

#include <stdio.h>

int main() {

    int i;
    float numerador, denominador, soma;

```

```

numerator = 1;
denominator = 1;
soma = 0;

for(i = 1; i <= 20; i++){
    soma = soma + numerator / denominator;
    numerator = numerator + 2;
    denominator = denominator * 2;
}

printf("Resultado: %.2f", soma);

return 0;
}

```

14. Escreva um programa que mostre o resultado dos primeiros 50 termos da série: $1000/1 - 997/2 + 994/3 - 991/4 + \dots$ **Resultado:** 685.30.

```

#include <stdio.h>

int main() {

    float numerator, denominator, soma;
    int i;

    numerator = 1000;
    denominator = 1;
    soma = 0;

    for(i = 1; i <= 50; i++){

        if(i % 2 != 0){
            soma = soma + numerator / denominator;
        }
        else{
            soma = soma - numerator / denominator;
        }

        numerator = numerator - 3;
        denominator = denominator + 1;
    }

    printf("A soma é: %.2f", soma);

    return 0;
}

```

15. Escreva um programa que escreva os 10 primeiros números da sequência de Fibonacci. **Resultado:** 1, 1, 2, 3, 5, 8, 13, 21, 34, 55.

```

#include <stdio.h>

int main() {

    int anterior, atual, i;

    anterior = 1;
    atual = 1;

    printf("%d %d ", anterior, atual);

    for(i = 3; i <= 10; i++){
        atual = atual + anterior;
        anterior = atual - anterior;

        printf("%d ", atual);
    }

    return 0;
}

```

16. Escreva um programa que forneça quantos números devem existir em sequência a partir do 1 (1, 2, 3, 4,...) para que a sua soma ultrapasse o valor 100 o mínimo possível.

```

#include <stdio.h>

int main() {

    int num, soma, quant;

    num = 1;
    soma = 0;
    quant = 0;

    while(soma <= 100){
        soma = soma + num;
        num++;
        quant++;
    }

    printf("\nQuantidade de números: %d", quant);
    printf("\nResultado da soma: %d", soma);

    return 0;
}

```


17. A definição de "amplitude" em Estatística é dada pela grandeza numérica resultante da diferença entre o maior valor e o menor valor do conjunto de valores de uma amostra. Escreva programa que leia uma sequência de números reais positivos terminada em zero (o número zero não deve ser processado pois serve para marcar o final da entrada de dados). O programa deve determinar e mostrar o valor da amplitude estatística dos valores. **Exemplo:** para a sequência 4.5, 5.2, 1.7, 1.3, 1.9, 2.2, 8.3, 9.1, 5.4 e 0, teremos o resultado de 7.8 como amplitude.

```
#include <stdio.h>

int main() {

    float num, menor, maior, amplitude;

    printf("Informe número: ");
    scanf("%f", &num);

    menor = num;
    maior = num;

    while(num != 0){

        if(num < menor){
            menor = num;
        }

        if(num > maior){
            maior = num;
        }

        printf("Informe número: ");
        scanf("%f", &num);
    }

    amplitude = maior - menor;

    printf("A amplitude é: %.1f", amplitude);

    return 0;
}
```

18. Escreva um programa que apresente o seguinte menu:

```
# MENU PRINCIPAL #
```

```
[1] Inserir  
[2] Excluir  
[3] Consultar  
[4] Listar  
[5] Sair
```

Se as opções 1, 2, 3 ou 4 forem escolhidas, mostrar na tela o nome da opção (Inserir, Excluir, Consultar, Listar). Se a opção 5 for escolhida, encerrar a execução do programa.

```
#include <stdio.h>

int main() {

    int opcao;

    do{

        printf("# MENU PRINCIPAL #\n");
        printf("\n[1] Inserir.");
        printf("\n[2] Excluir.");
        printf("\n[3] Consultar.");
        printf("\n[4] Listar.");
        printf("\n[5] Sair\n");
        scanf("%d", &opcao);

        switch(opcao){
            case 1:
                printf("Opção escolhida: Inserir\n");
                break;
            case 2:
                printf("Opção escolhida: Excluir\n");
                break;
            case 3:
                printf("Opção escolhida: Consultar\n");
                break;
            case 4:
                printf("Opção escolhida: Listar\n");
                break;
        }

    }while(opcao != 5);

    return 0;
}
```

19. Escreva um programa que apresente na tela o seguinte menu:

```
# MENU PRINCIPAL #
```

```
[1] Digitar um número par  
[2] Digitar um número ímpar  
[3] Sair
```

Caso o usuário escolha a opção 1, ele deve digitar um número par para voltar ao menu. O mesmo deve ser feito na opção 2, mas para um número ímpar. Caso o usuário escolha a opção 3, o programa deve apresentar a quantidade de números pares digitados na opção 1, a quantidade de números ímpares digitados na opção 2 e encerrar o programa.

```
#include <stdio.h>

int main() {

    int opcao, num, quant_pares, quant_impares;

    quant_pares = 0;
    quant_impares = 0;

    do{
        printf("# MENU PRINCIPAL #\n");
        printf("\n[1] Digitar um número par.");
        printf("\n[2] Digitar um número ímpar.");
        printf("\n[3] Sair\n");
        scanf("%d", &opcao);

        switch(opcao){
            case 1:
                do{
                    printf("Digite um número par: ");
                    scanf("%d", &num);
                }while(num % 2 != 0);

                quant_pares++;
                break;

            case 2:
                do{
                    printf("Digite um número ímpar: ");
                    scanf("%d", &num);
                }while(num % 2 == 0);

                quant_impares++;
                break;
        }

    }while(opcao != 3);
```

```

printf("\nQuantidade números pares: %d", quant_pares);
printf("\nQuantidade números ímpares: %d", quant_impares);

return 0;
}

```

20. Escreva um programa que verifique a validade de uma senha fornecida pelo usuário (a senha válida é o número 2009). Se a senha informada pelo usuário for inválida, a mensagem "ACESSO NEGADO" deve ser mostrado na tela e deve ser solicitada uma nova senha até que ela seja válida. Caso contrário deve ser mostrado na tela a mensagem "ACESSO PERMITIDO" junto com um número de vezes que a senha foi informada.

```

#include <stdio.h>

int main() {

    int senha, tentativas;

    tentativas = 0;

    do{
        printf("Informe senha: ");
        scanf ("%d", &senha);

        if(senha != 2009){
            printf("\nACESSO NEGADO!\n");
        }

        tentativas++;

    }while (senha != 2009);

    printf("\nACESSO PERMITIDO!");
    printf("\nNúmero de tentativas: %d", tentativas);

    return 0;
}

```