

Aula 04 - Estruturas de Repetição

1. Lógica

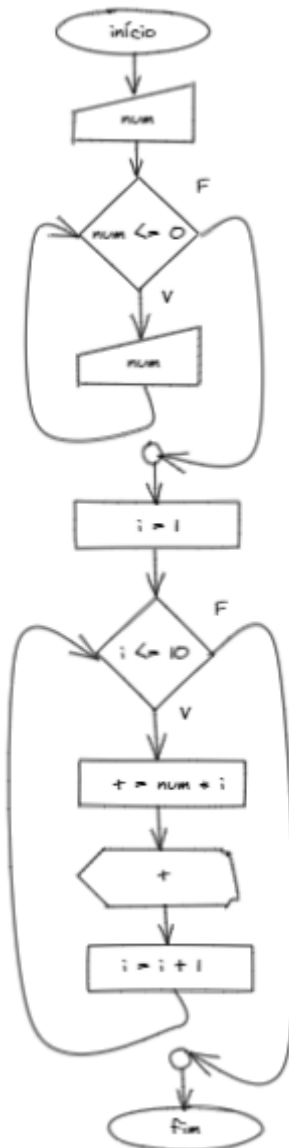
Em determinados programas, precisamos de entrada de dados, processamento e saída de dados. Entretanto, em alguns casos, precisamos também ter decisões e até mesmo **repetições** de determinados trechos de código. Neste caso, podemos utilizar as estruturas de repetição While, Do..While ou For no Java.

2. While

Exemplo:

Fazer um programa para receber um número e validar se esse número é positivo. Após isso, exibir a tabuada de 1 a 10 desse número:

Fluxograma:



Programação Java:

```

package estrutura_repeticao;

import java.util.Scanner;

public class Exemplo01 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int num, t, i;

        System.out.printf("Digite um número positivo: ");
        num = ler.nextInt();
    }
}

```

```
        while(num <= 0) {
            System.out.printf("Erro, número negativo! Digite um número
positivo: ");
            num = ler.nextInt();
        }

        i = 1;

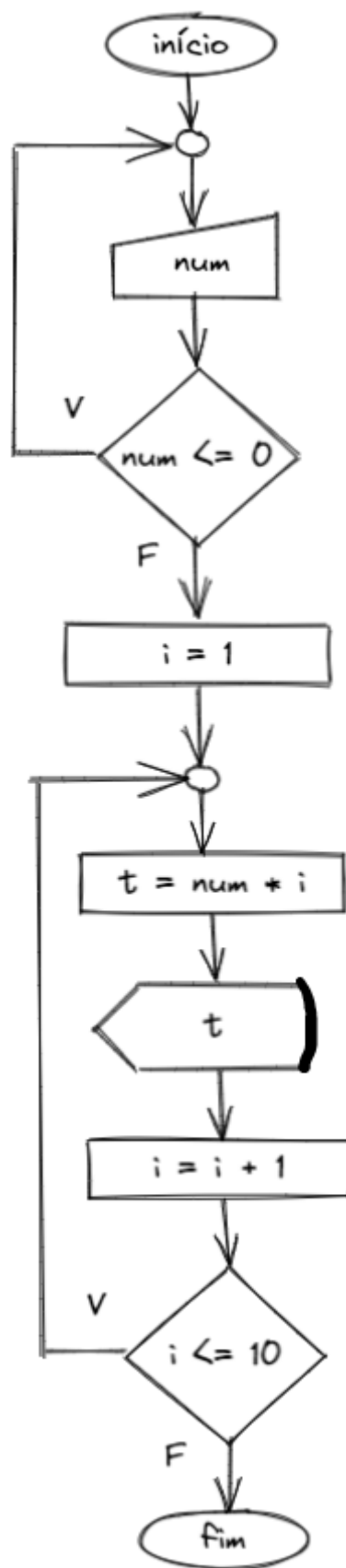
        while (i <= 10) {
            t = num * i;
            System.out.printf("\n%d X %d = %d", num, i, t);
            i++;
        }
    }
}
```

3. Do..While

Exemplo:

Fazer um programa para receber um número e validar se esse número é positivo. Após isso, exibir a tabuada de 1 a 10 desse número:

Fluxograma:



Programação Java:

```
package estrutura_repeticao;
```

```

import java.util.Scanner;

public class Exemplo03 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int num, t, i;

        do {
            System.out.printf("Digite um número positivo: ");
            num = ler.nextInt();
        }
        while(num <= 0);

        i = 1;

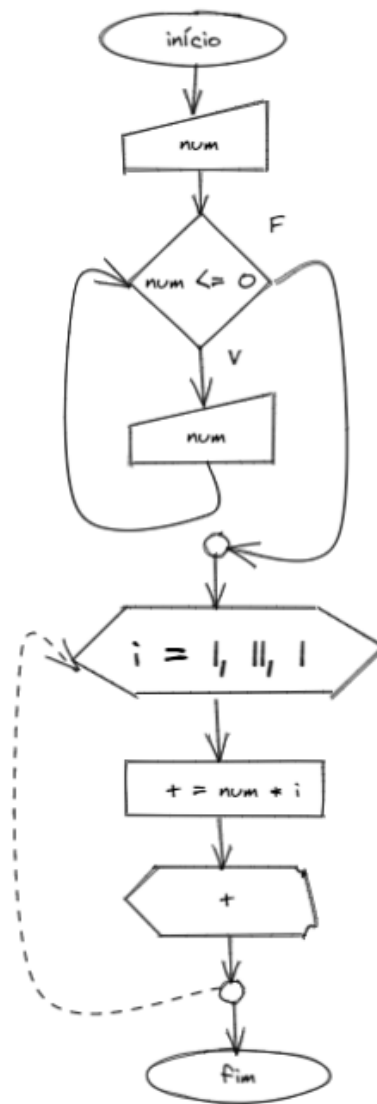
        do {
            t = num * i;
            System.out.printf("\n%d X %d = %d", num, i, t);
            i++;
        }
        while (i <= 10);
    }
}

```

4. For

Exemplo:

Fazer um programa para receber um número e validar se esse número é positivo. Após isso, exibir a tabuada de 1 a 10 desse número:



Programação Java:

```

package estrutura_repeticao;

import java.util.Scanner;

public class Exemplo02 {
    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int num, t, i;

        System.out.printf("Digite um número positivo: ");
        num = ler.nextInt();

        while(num <= 0) {

```

```

        System.out.printf("Erro, número negativo! Digite um número
positivo: ");
        num = ler.nextInt();
    }

    for (i=1; i<=10; i++) {
        t = num * i;
        System.out.printf("\n%d X %d = %d", num, i, t);
    }
}
}

```

5. Teste de Mesa

Na Lógica de Programação, podemos utilizar um recurso extremamente útil chamado Teste de Mesa. Ele consiste em anotar todos os valores das variáveis conforme o programa vai avançando passo a passo.

Utilizando o exemplo acima, podemos ter o seguinte teste de mesa:

Teste de Mesa:

num	i	t	mensagem
-4	1	5	erro, num negativo
-2	2	10	erro, num negativo
5	3	15	
	4		

Neste caso, podemos ver que a cada looping do programa os valores estão sendo alterados corretamente, tendo o programa exibindo o resultado da tabuada do número 5 corretamente.

6. Manipulando Loopings (While, Do While e For)

6.1. Break

Caso você queira interromper um looping, você pode utilizar o comando break para isso, veja:

```

package estrutura_repeticao;

import java.util.Scanner;

public class Exemplo01 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int num, t, i;

        System.out.printf("Digite um número positivo: ");
        num = ler.nextInt();

        while(num <= 0) {
            System.out.printf("Erro, número negativo! Digite um número
positivo: ");
            num = ler.nextInt();
        }

        i = 1;

        while (i <= 10) {
            t = num * i;
            System.out.printf("\n%d X %d = %d", num, i, t);

            if (i == 8)
                break;

            i++;
        }
    }
}

```

Neste caso, no mesmo exercício da tabuada, quando o $i = 8$, nós vamos “parar” o looping com o comando **break**.

6.2. Continue

Caso você queira pular uma iteração do looping, você pode utilizar o comando “continue” para isso, veja:

```

package estrutura_repeticao;

```



```

import java.util.Scanner;

public class Exemplo02 {
    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int num, t, i;

        System.out.printf("Digite um número positivo: ");
        num = ler.nextInt();

        while(num <= 0) {
            System.out.printf("Erro, número negativo! Digite um número
positivo: ");
            num = ler.nextInt();
        }

        for (i=1; i<=10; i++) {
            if (i == 8)
                continue;

            t = num * i;
            System.out.printf("\n%d X %d = %d", num, i, t);
        }
    }
}

```

Neste caso, no mesmo exercício da tabuada, quando o $i = 8$, nós vamos “pular” aquela iteração do looping com o comando **continue**.

Exercícios

19W. Criar uma rotina de entrada que aceite somente um valor positivo.

```

package estrutura_repeticao;

import java.util.Scanner;

public class Ex19 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int num;
    }
}

```

```

        System.out.print("Digite um valor positivo: ");
        num = ler.nextInt();

        while (num < 0) {
            System.out.print("Erro! Digite apenas números positivos!");
            System.out.print("\nDigite um valor positivo: ");
            num = ler.nextInt();
        }

        System.out.print("Numero positivo foi digitado!");

        ler.close();
    }
}

```

20. Entrar com dois valores via teclado, onde o segundo deverá ser maior que o primeiro. Caso contrário solicitar novamente apenas o segundo valor.

```

package estrutura_repeticao;

import java.util.Scanner;

public class Ex20 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int a, b;

        System.out.print("Digite o primeiro número: ");
        a = ler.nextInt();

        System.out.print("Digite o segundo número: ");
        b = ler.nextInt();

        while (b <= a) {
            System.out.print("Erro! O segundo número deve ser maior que o primeiro!");
            System.out.print("\nDigite o segundo número: ");
            b = ler.nextInt();
        }
    }
}

```

```

        System.out.print("O segundo número foi maior do que o
primeiro!");

        ler.close();
    }
}

```

21. Exibir a tabuada do número cinco no intervalo de um a dez.

```

package estrutura_repeticao;

import java.util.Scanner;

public class Ex21 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int i = 1, num = 7, t;

        while(i <= 10) {
            t = num * i;
            System.out.printf("\n%d X %d = %d", num, i, t);
            i = i + 1;
        }

        ler.close();
    }
}

```

22. Entrar via teclado com um valor qualquer. Travar a digitação, no sentido de aceitar somente valores positivos. Após a digitação, exibir a tabuada do valor solicitado, no intervalo de um a dez.

```

package estrutura_repeticao;

import java.util.Scanner;

public class Ex22 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int num, i=1, t;
    }
}

```

```

        System.out.print("Digite o número que você quer a tabuada: ");
        num = ler.nextInt();

        while (num <= 0) {
            System.out.print("Erro! Digite um número positivo!");
            System.out.print("\nDigite o número que você quer a
tabuada: ");
            num = ler.nextInt();
        }

        while (i <= 10) {
            t = num * i;
            System.out.printf("\n%d X %d = %d", num, i, t);
            i = i + 1;
        }

        ler.close();
    }
}

```

23. Entrar via teclado com um valor (X) qualquer. Travar a digitação, no sentido de aceitar somente valores positivos. Solicitar o intervalo que o programa que deverá calcular a tabuada do valor digitado, sendo que o segundo valor (B), deverá ser maior que o primeiro (A), caso contrário, digitar novamente somente o segundo. Após a validação dos dados, exibir a tabuada do valor digitado, no intervalo decrescente, ou seja, a tabuada de X no intervalo de B para A.

```

package estrutura_repeticao;

import java.util.Scanner;

public class Ex23 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int num, a, b, t;

        System.out.printf("Digite um número para obter a tabuada: ");
        num = ler.nextInt();

        while (num <= 0) {

```

```

        System.out.printf("Erro! Digite apenas números
positivos!");
        System.out.printf("\nDigite um número para obter a tabuada:
");
        num = ler.nextInt();
    }

    System.out.printf("Digite o intervalo inicial da tabuada: ");
    a = ler.nextInt();

    System.out.printf("Digite o intervalo final da tabuada: ");
    b = ler.nextInt();

    while(b <= a) {
        System.out.printf("Erro! O intervalo final deve ser maior
que o inicial!");
        System.out.printf("\nDigite o intervalo final da tabuada:
");
        b = ler.nextInt();
    }

    while (b>=a) {
        t = num * b;
        System.out.printf("%d X %d = %d\n", num, b, t);
        b = b - 1;
    }

    ler.close();
}
}

```

24. Exibir a tabuada dos valores de um a vinte, no intervalo de um a dez. Entre as tabuadas, solicitar que o usuário pressione uma tecla.

```

package estrutura_repeticao;

import java.util.Scanner;

public class Ex24 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int i, t=1, r;
    }
}

```

```

        while(t <= 20) {
            i = 1;
            while (i <= 10) {
                r = t * i;
                System.out.printf("%d X %d = %d\n", t, i, r);
                i = i + 1;
            }
            System.out.println("Pressione uma tecla para
continuar...");
            ler.nextLine();

            t = t + 1;
        }

        ler.close();
    }
}

```

25. Exibir a soma dos números inteiros positivos do intervalo de um a cem.

```

package estrutura_repeticao;

public class Ex25 {

    public static void main(String[] args) {
        int total=0, i=1;
        String texto="";

        while (i <= 100) {
            total = total + i;
            if (i == 100)
                texto = texto + i + " = " + total;
            else
                texto = texto + i + " + ";
            i = i + 1;
        }

        System.out.print(texto);
    }
}

```

26. Entrar via teclado com dez valores positivos. Consistir a digitação e enviar mensagem de erro, se necessário. Após a digitação, exibir:

- a) O maior valor;
- b) A soma dos valores;
- c) A média aritmética dos valores;

```
package estrutura_repeticao;

import java.util.Scanner;

public class Ex26 {

    public static void main(String[] args) {
        Scanner ler = new Scanner(System.in);

        int i=1, num, maiorNumero=1;
        double somaNumeros=0, mediaNumeros;

        while (i <= 10) {
            System.out.printf("Digite o %dº valor: ", i);
            num = ler.nextInt();

            while (num <= 0) {
                System.out.print("O número digitado negativo! Digitar apenas positivos!");
                System.out.printf("\nDigite o %dº valor: ", i);
                num = ler.nextInt();
            }

            somaNumeros = somaNumeros + num;

            if (num > maiorNumero)
                maiorNumero = num;

            i = i + 1;
        }

        mediaNumeros = somaNumeros / 10;

        System.out.println("O maior número é: " + maiorNumero);
        System.out.println("A soma dos números é: " + somaNumeros);
        System.out.println("A média dos números é: " + mediaNumeros);

        ler.close();
    }
}
```

```
}
```

27. Entrar via teclado com “N” valores quaisquer. O valor “N” (que representa a quantidade de números) será digitado, deverá ser positivo, mas menor que vinte. Caso a quantidade não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. Após a digitação dos “N” valores, exibir:

- a) O maior valor;
- b) O menor valor;
- c) A soma dos valores;
- d) A média aritmética dos valores;
- e) A porcentagem de valores que são positivos;
- f) A porcentagem de valores negativos;

28. Entrar via teclado com “N” valores quaisquer. O valor “N” (que representa a quantidade de números) será digitado, deverá ser positivo, mas menor que vinte. Caso a quantidade não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. Após a digitação dos “N” valores, exibir:

- a) O maior valor;
- b) O menor valor;
- c) A soma dos valores;
- d) A média aritmética dos valores;
- e) A porcentagem de valores que são positivos;
- f) A porcentagem de valores negativos;

Após exibir os dados, perguntar ao usuário se deseja ou não uma nova execução do programa. Consistir a resposta no sentido de aceitar somente “S” ou “N” e encerrar o programa em função dessa resposta.