

.....

Aula 8 – Estrutura de Repetição (while / do..while)

.....

Prof. Fernando Zuher





Universidade
Anhembi Morumbi

LAUREATE INTERNATIONAL UNIVERSITIES



Estrutura de Repetição

- Conceito de repetição (looping) é utilizado quando se deseja repetir certo trecho de instruções por um número finito de vezes
- Dois caminhos:
 - Escrever o trecho N vezes 
 - OU
 - Utilizar o conceito de Looping 

Escrever o Trecho N vezes

```
double n1, n2, n3, n4, n5, n6, n7, n8, n9, n10;  
System.out.println("Informe nota do aluno 1: ");  
n1 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 2: ");  
n2 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 3: ");  
n3 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 4: ");  
n4 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 5: ");  
n5 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 6: ");  
n6 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 7: ");  
n7 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 8: ");  
n8 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 9: ");  
n9 = entrada.nextDouble();  
System.out.println("Informe nota do aluno 10:");  
n10 = entrada.nextDouble();
```



Estrutura de Repetição

- **Vantagens:**

- O algoritmo passa a ter um tamanho menor devido a estrutura de repetição e
- É possível aumenta a amplitude de processamento sem alterar o tamanho do código

Tipos de Repetição Finita

O **TOTAL** de repetições
já é **conhecido**

O **TOTAL** de repetições é
desconhecido

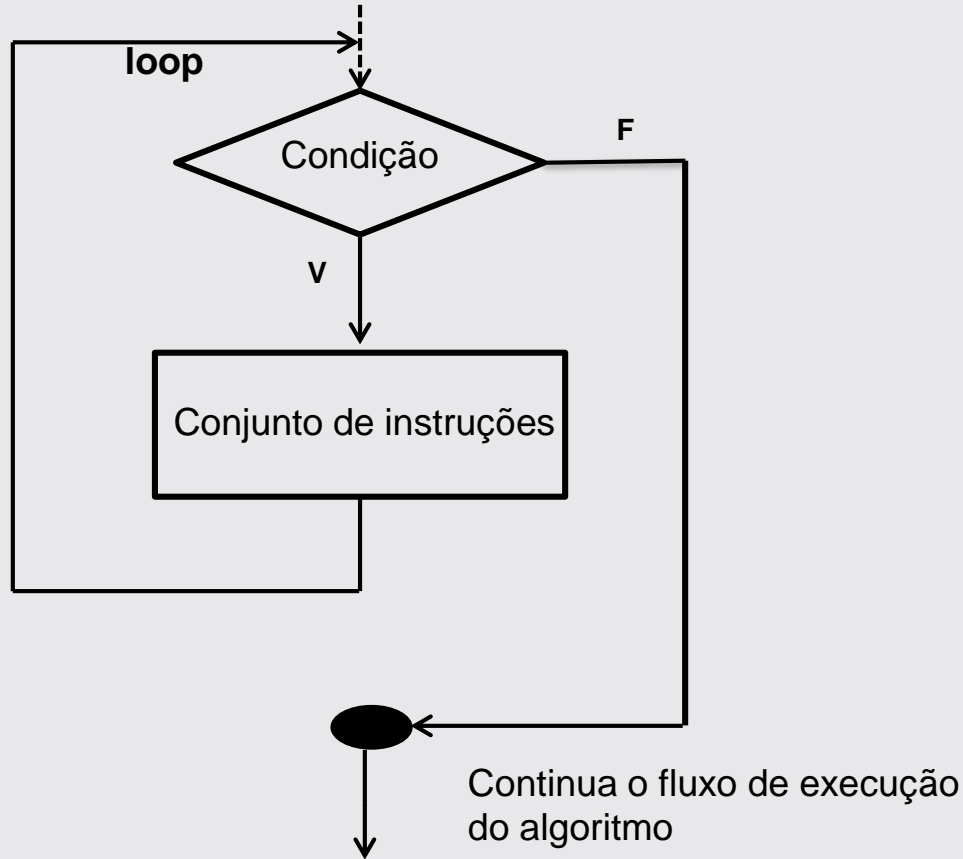
.....

Comando while

.....



while (Enquanto/Faça)



while – Sintaxe

- Sintaxe do comando while:

Bloco de instruções

```
while( condição )
```

```
{
```

```
    instruções_dentro_do_laço;
```

```
    instruções_dentro_do_laço;
```

```
    instruções_dentro_do_laço;
```

```
}
```

```
próxima_instrução_do_algoritmo;
```


while (Enquanto/Faça)

- A condição de repetição é verificada **antes** de entrar no laço
 - Enquanto o resultado da condição for **verdadeira** – executa o bloco de instruções
 - Quando a condição se tornar **falsa**, o laço é encerrado e o fluxo vai para a próxima instrução do algoritmo
- Portanto, a condição é testada a cada iteração (volta)

Laço é um **bloco de instruções** que será executado **repetidas vezes** e que está contido em uma **estrutura de repetição**

Contador

- Variável utilizada para contar cada repetição em uma estrutura de repetição
 - Utilizado quando o número de repetições é conhecido
 - Forma mais usual é incrementar ou decrementar o contador
 - Geralmente é uma variável do tipo **int**
 - Deve ser **inicializado** antes de ser utilizado!



Exemplo 1

- Escreva um algoritmo que imprima a sequência dos números inteiros de 1 até 100.

```
int contador; //declara o contador
contador = 1; //inicializa o contador
while(contador <= 100){
    System.out.println( contador + " ");
    contador = contador + 1; //incremento do contador
}
```

Exemplo 2

- Faça um algoritmo que imprima os números pares entre 0 e 10.

```
int contador, resto; //declara o contador
contador = 0; //inicializa o contador
while(contador <= 10){
    resto = contador % 2; //calcula o resto da divisão
    if( contador > 1 && resto == 0){
        System.out.println("O número " + contador + " é PAR!");
    }
    contador = contador + 1; //incrementa o contador
}
```

Acumulador

- Variável responsável por acumular valores dentro de um laço
- Similar a um contador, mas o valor a ser acumulado é **variável**
- Exemplo:
 - Calcular o somatório de um conjunto de valores;

Exemplo 3

- Faça um algoritmo que calcule a média das notas de 5 alunos.

```
1  int contador; //declara o contador
2  double nota, media, acmNotas; //declara o acumulador
3  contador = 0; //inicializa o contador
4  acmNotas = 0; //inicializa o acumulador
5  while(contador < 5){
6      System.out.println("Digite a " + (contador + 1) + "ª nota:");
7      nota = entrada.nextInt();
8      acmNotas = acmNotas + nota; //acumula a nota lida
9      contador = contador + 1; //incrementa o contador
10 }
11 media = acmNotas / contador; //Calcula a média das 5 notas
12 System.out.println("A média é : " + media);
```

Variável de Controle

- variável utilizada para controlar as repetições em uma estrutura de repetição
 - Utilizada quando o número de repetições não é conhecido
 - A variável terá o seu estado (valor) modificado

Exemplo 4

- Faça um algoritmo que leia o nome e a idade de um **conjunto de pessoas** e mostrar o nome da pessoa se ela for maior de idade.

```
1 char resp; //declara a variável de controle
2 String nome;
3 int idade;
4 resp = 's'; //inicializa a variável de controle
5 while(resp == 's'){
6     System.out.println("Digite seu nome e sua idade: ");
7     nome = entrada.nextLine();
8     idade = entrada.nextInt();
9     if(idade >= 18){ //verifica se é maior de idade
10         System.out.println( nome + " você é maior de idade! ");
11     }
12     System.out.println("Deseja continuar? (s/n): "); //Modifica a variável de controle
13     resp = entrada.next().charAt(0);
14 }
```


.....

Comando do..while

.....



do..while

- A condição é testada no final!
- Sempre após a execução do bloco do laço!
- A diferença básica para o comando while é que o bloco a ser repetido sempre executa ao menos **uma vez!**

do..while - Sintaxe

- Sintaxe do comando do..while

Bloco de instruções

do

{

instruções_dentro_do_laço;

instruções_dentro_do_laço;

instruções_dentro_do_laço;

}while (condição) ;

próxima_instrução_do_algoritmo;

Exemplo 1

- Faça um algoritmo para ler uma nota e validar se ela está entre 0 e 10.

```
double nota;  
do{  
    System.out.println("Digite uma nota: ");  
    nota = entrada.nextDouble();  
    if( nota < 0 || nota > 10){ //verifica a nota para dar mensagem de erro!  
        System.out.println("Valor inválido. A nota deve estar entre 0 e 10 !");  
        System.out.println("Digite novamente a nota");  
    }  
}while( ! (nota >= 0 && nota <= 10) ) ; //Valida se a nota é válida!
```

Exemplo 2

- Ler 10 números inteiros e encontrar o maior número lido

```
int num, maior, contador; //declara o contador e a variável p/ guardar o maior número
contador = 0; //Inicializa o contador com zero
do{
    System.out.println("Digite um número: ");
    num = entrada.nextInt();
    if( contador == 0) {    //Verifica se é a 1ª passagem no laço para iniciar a variável maior
        maior = num; //guarda o valor do 1º número lido em maior
    }
    if(num > maior){    //verifica se o número lido é maior que maior
        maior = num; //se for, atribui o número lido a variável maior
    }
    contador = contador + 1; //incrementa o contador
}while(contador < 10);
System.out.println("O maior número é: " + maior);
```

.....

Vamos praticar?

.....

Exercícios

1 – Escreva um algoritmo que imprima a sequencia do número 100 ao 200.

Exemplo: 100 101 102 103 ... 198 199 200

2 – Faça um algoritmo que imprima a **metade** de cada número no intervalo de 10 a 20.

Exemplo:

a metade de 10 é 5.0

a metade de 11 é 5.5

a metade de 12 é 6.0

...

a metade de 18 é 9.0

a metade de 19 é 9.5

a metade de 20 é 10.0

Exercícios

3 - Faça um algoritmo que **leia 200 números** inteiros e imprima quantos são pares e quantos são ímpares.

Exemplo:

Digite o 1º número:

4

Digite o 2º número:

-201

.....

Digite o 199º número:

976

Digite o 200º número:

20002

O total de pares é: _____

O total de ímpares é: _____

Exercícios

- 4 – Faça um algoritmo que **leia dez números** inteiros e imprima o **menor** número da lista.
- 5 – Faça um algoritmo que imprima a tabuada do número 5.

Exemplo:

Tabuado do 5

$$5 \times 0 = 0$$

$$5 \times 1 = 5$$

$$5 \times 2 = 10$$

...

$$5 \times 10 = 50$$

Exercícios

6 - Construir um algoritmo que leia um número inteiro e imprimir a sequência: 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024... enquanto o valor for menor ou igual ao valor lido.

Exemplo:

Digite um número: **82**

Sequência impressa: **1, 2, 4, 8, 16, 32, 64**

Exercícios

7 - O IMC (índice de Massa Corporal) é uma medida do grau de obesidade de uma pessoa. Faça um algoritmo que leia a altura e o peso de **20 pessoas**. Calcular o IMC de cada pessoa e verificar quantas pessoas estão com o IMC entre 18,5 e 24,9 que é considerado sem obesidade.

$$\text{IMC} = \text{peso} / \text{altura}^2$$

Exercícios

- 8 – Dado o cardápio de uma lanchonete:

| PRODUTO | CÓDIGO | PREÇO |
|-----------------|--------|-------|
| Cachorro quente | 100 | 1,20 |
| Bauru Simples | 101 | 1,30 |
| Bauru com ovo | 102 | 1,50 |
| Hambúrguer | 103 | 1,20 |
| Cheeseburger | 104 | 1,30 |
| Refrigerante | 105 | 1,00 |

- Faça um algoritmo que leia o código do produto e a quantidade que o cliente deseja comprar.
- O cliente pode **comprar mais de um** produto! Perguntar se o cliente deseja comprar mais produtos.
- Calcular o valor total a ser pago por cada produto e valor total da compra.

Exercícios

9 – Faça um algoritmo que leia duas notas de 5 alunos, Calcule a média aritmética de cada aluno e calcule a média da classe.

A entrada de **cada nota** deve ser validada! Ou seja, o programa somente avança se a entrada de nota estiver entre 0 e 10. Caso contrário, solicita novamente! Utilize o do..while para validar as notas!!