

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE

COMANDOS DE REPETIÇÃO (ENQUANTO - WHILE)

Disciplina: Algoritmos e POO

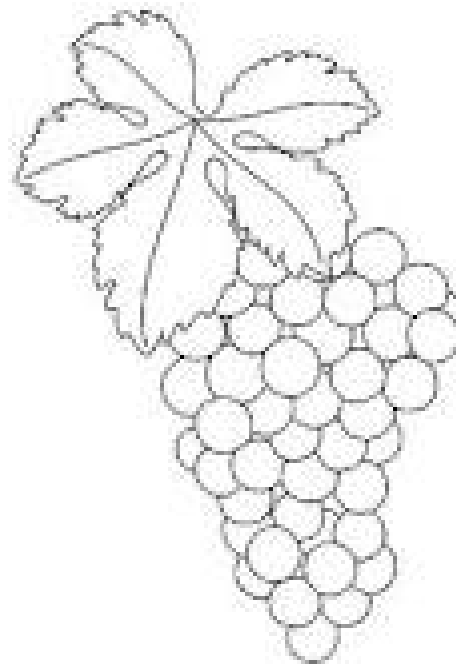
Professora: Alba Lopes

alba.lopes@ifrn.edu.br

<http://docente.ifrn.edu.br/albalopes>

COMANDOS DE REPETIÇÃO

- Usada quando se deseja que um trecho do algoritmo seja repetido várias vezes
- A quantidade de repetições pode ser fixa ou depender de uma condição



COMANDOS DE REPETIÇÃO

- As estruturas de repetição são classificadas em 3 tipos
 - Repetição com **teste no início** do laço
 - Repetição com **teste no final** do laço
 - Repetição com **variável de controle**
- São também chamadas de **laços de repetição!**



REPETIÇÃO COM TESTE NO INÍCIO

○ While

- Essa estrutura repete uma sequencia de comandos enquanto uma determinada condição for verdadeira
- Essa condição é determinada por uma expressão lógica



ENQUANTO...FAÇA

○ Sintaxe:

```
while (<expressao-logica> ){  
    <sequencia-de-comandos>  
}
```

<i><expressao_logica></i>	Essa expressão é avaliada antes de cada repetição do laço
}	Indica o fim da sequencia de comandos . Cada vez que a execução atinge esse ponto, volta-se ao início do laço para que a <expressão-logica> seja avaliada novamente Se o resultado dessa avaliação for TRUE (VERDADEIRO) , a <sequencia-de-comandos> será executada mais uma vez. Caso contrário, a execução prosseguirá a partir do primeiro comando após o }

EXEMPLO

- Criar um algoritmo para escrever 10x a mensagem “Programação Orientada a Objetos é muito legal!”

```
public class Enquanto {  
  
    public static void main(String [] args) {  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
        System.out.println("Programação Orientada a Objetos é muito legal!")  
    }  
}
```



EXEMPLO (NÚMERO FIXO DE VEZES)

- E se tivesse que escrever a mensagem 1000 vezes?
 - Melhor usar um comando de repetição!

```
public static void main(String [] args) {  
  
    int contador;  
    contador = 0;  
    while (contador < 1000){  
        System.out.println("Programação Orientada a Objetos é muito legal!");  
        contador = contador + 1;  
    }  
}
```



EXEMPLO (NÚMERO FIXO DE VEZES)

- E se tivesse que escrever a mensagem 1000 vezes?
 - Melhor usar um comando de repetição!

```
public static void main(String [] args) {  
  
    int contador;  
    contador = 0;  
    while (contador < 1000){  
        System.out.println("Programação Orientada a Objetos é muito legal!");  
        contador = contador + 1;  
    }  
}
```

É preciso declarar uma variável para ser utilizada no teste da condição do laço (variável de controle)



EXEMPLO (NÚMERO FIXO DE VEZES)

- E se tivesse que escrever a mensagem 1000 vezes?
 - Melhor usar um comando de repetição!

```
public static void main(String [] args) {  
  
    int contador;  
    contador = 0;  
    while (contador < 1000){  
        System.out.println("Programação Orientada a Objetos é muito legal!");  
        contador = contador + 1;  
    }  
}
```

É preciso
inicializar a
variável de
controle



EXEMPLO (NÚMERO FIXO DE VEZES)

- E se tivesse que escrever a mensagem 1000 vezes?
 - Melhor usar um comando de repetição!

```
public static void main(String [] args) {  
  
    int contador;  
    contador = 0;  
    while (contador < 1000){  
        System.out.println("Programação Orientada a Objetos é muito legal!");  
        contador = contador + 1;  
    }  
}
```

É preciso
IDENTIFICAR a
condição de
parada do laço



EXEMPLO (NÚMERO FIXO DE VEZES)

- E se tivesse que escrever a mensagem 1000 vezes?
 - Melhor usar um comando de repetição!

```
public static void main(String [] args) {  
  
    int contador;  
    contador = 0;  
    while (contador < 1000){  
        System.out.println("Programação Orientada a Objetos é muito legal!");  
        contador = contador + 1;  
    }  
}
```

É preciso
ATUALIZAR a
variável de
controle do laço



EXEMPLO (NÚMERO INDETERMINADO)



- Algoritmo para comer um cacho de uva
 - Não se sabe ao certo quantas uvas tem no cacho
 - Sempre é preciso identificar se há ou não uvas no cacho para continuar comendo...

```
public class Uvas {  
  
    public static void main(String [] args) {  
        Scanner teclado = new Scanner(System.in);  
        int ha_uvas_no_cacho;  
        System.out.println("Há uvas no cacho? Digite 1 para SIM e 2 para NÃO: ");  
        ha_uvas_no_cacho = teclado.nextInt();  
        while ( ha_uvas_no_cacho != 2){  
            System.out.println("Retire uma uva!");  
            System.out.println("Ainda há uvas no cacho? Digite 1 para SIM e 2 para NÃO: ");  
            ha_uvas_no_cacho = teclado.nextInt();  
        }  
        System.out.println("Acabou!...");  
    }  
}
```

EXERCÍCIOS

- Resolva os exercícios a seguir usando **comandos de repetição**
 1. Calcular a soma dos números de 1 a 20 usando comando de repetição
 2. Escrever os números pares existentes entre 1 e 20
 3. Ler um número e calcular seu fatorial
 - Ex: $5! = 5 * 4 * 3 * 2 * 1$
 4. Ler 10 números do usuário e calcule a média
 5. Ler a idade de várias pessoas e calcular a média das idades. Encerrar a execução quando valor negativo for digitado.



EXERCÍCIOS

6. Leia uma sequencia de números do usuário e escreva o dobro de cada número lido. Encerre a execução quando o número digitado for negativo
7. Escreva um algoritmo que calcule a soma dos números pares enquanto o valor da soma for menor do que 100.

