



#R4E

Software Developer

Algoritmia e Programação

Estruturas de Repetição - Ciclos - Do While
Menus



Conteúdo



- Estruturas de Repetição
 - Ciclo Do While
- Menus
- Ciclos Aninhados
- Ciclos Infinitos
- Incrementos e Decrementos
- Resumo

Ciclo Fazer... Enquanto... - do... while...

- O ciclo fazer... enquanto executa uma instrução ou bloco de instruções enquanto uma determinada condição é verdadeira.

```
do {  
    instrução1;  
    instrução2;  
    ...  
} while (condição)
```

- Funcionamento:
 - São executadas as instruções associadas ao ciclo.
 - A condição é avaliada:
 - Se Falsa, o ciclo termina e o algoritmo continua na instrução imediatamente a seguir.
 - Se Verdadeira, volta-se a executar as instruções associadas ao ciclo.

Ciclo Fazer... Enquanto... - do... while...

- Neste tipo de ciclo o bloco de instruções é executado **pelo menos uma vez**, enquanto que nos ciclos **enquanto (while)** e **para (for)** pode o bloco de instruções nunca ser executado.
- Não deve ser usado se não tivermos a certeza que o facto de executarmos as ações antes de testarmos a veracidade da condição não é prejudicial
- O ciclo **fazer... enquanto...** é bastante usado para criar menus de programas



Ciclo Fazer... Enquanto... - do... while...

```
int numero = 1;
```

```
do {  
    System.out.println(numero);  
    numero++;  
} while (numero <= 10);
```

- Este programa escreve os números inteiros de 1 a 10 inclusive

Menus



- A apresentação de menus de opções ao utilizador tira partido do modo de funcionamento do ciclo **do... while...**, uma vez que o Menu é apresentado pelo menos uma vez.
- Estrutura de um menu:

```
do {  
    do {  
        Apresentação das opções;  
    } while (opção inválida)  
    Executar opção escolhida;  
} while (opção != saída)
```

Menus

- Exemplo:

```
do {  
    do {  
        System.out.println("1 - Primeira Opção");  
        System.out.println("2 - Segunda Opção");  
        System.out.println("0 - Saída");  
    } while (opcao < 0 || opcao > 2);  
    switch (opcao) {  
        case 1: System.out.println("Escolheu a primeira opção");  
        break;  
        case 2: System.out.println("Escolheu a segunda opção");  
        break;  
        default: System.out.println("Escolheu a opção sair!");  
    }  
} while (opcao != 0);
```

Ciclos Aninhados

- Uma ação de um ciclo de qualquer tipo pode ser outro ciclo de qualquer tipo (**ciclo aninhado ou encadeado**).

```
int num1 = 1, num2 = 1;
while (num1 <=5) {
    while (num2 <= num1) {
        System.out.print( num2);
        num2++;
    }
    System.out.println(" ");
    num1++;
}
```

Resultado:

1
12
123
1234
12345

Ciclos Infinitos

- Chama-se **ciclo infinito** a um ciclo que **nunca termina**.
- São, quase sempre, resultado de **erros de programação**, muito embora sejam, por vezes, criados inadvertidamente, e outras vezes, por má codificação do algoritmo por parte do programador.

- Exemplo:

```
int numero = 1;
```

```
while (numero <= 10) {  
    System.out.println(numero);  
    numero--;  
}
```



Incrementos e Decrementos

- Como foi possível observar, a maioria dos exemplos de ciclos incluem instruções do género:
 - `numero++;`
 - `numero--;`
- Estas instruções designam-se **incremento** e **decremento** de variáveis.
- São usadas para fazer "avançar" (incrementar) ou "recuar" (decrementar) uma variável em + ou - 1 unidade, respetivamente.
- São bastante utilizadas em blocos de instruções de ciclos.



Ciclos (Resumo)

- A escolha do ciclo a ser usado deve ser feita com cuidado, devendo adotar-se a estrutura mais adequada ao contexto do problema que se pretende resolver.

	while	for	do... while
Sintaxe	<pre>while (condição) { ação; }</pre>	<pre>for (expressão1; condição; expressão2) { ação; }</pre>	<pre>do { ação; } while (condição)</pre>
Executa a instrução	Zero ou mais vezes	Zero ou mais vezes	Uma ou mais vezes
Testa a condição	Antes da ação	Antes da ação	Depois da ação
Utilização	Frequente	Frequente	Pouco Frequente



#R4E

Software Developer

Algoritmia e Programação

Estruturas de Repetição - Ciclos - Do While
Menus

