

ESTRUTURAS DE CONDIÇÃO

Por vezes nos veremos situações em que queremos que o algoritmo tome uma decisão e para isso usaremos as estruturas de condição e repetição que serão vistas na próxima aula.

Observe, a imagem abaixo ilustra em um diagrama de blocos, a estrutura if/else, se a média for maior ou igual a 7, o aluno está aprovado, se não, ele está reprovado.

```
IF MEDIA >= 7 THEN
    RESULTADO = "APROVADO";
ELSE
    RESULTADO = "REPROVADO";
```

Estruturas IF

A estrutura de decisão “*if*” ou **instrução de seleção única** constitui uma construção simples que permite que uma instrução ou um bloco de instruções seja executado após a análise de determinada condição que deverá ser satisfeita para que a execução ocorra. O que ela faz é avaliar se uma expressão é verdadeira (*true*) ou falsa (*false*).

É importante ressaltar que em Java essa expressão deve ser sempre um *boolean*. Assim, fragmentando o exemplo acima, “media do aluno”, se a média do aluno for maior ou igual a 7, ou seja, se a expressão for verdadeira, imprima “APROVADO”.

Mas se a expressão for falsa, a instrução ***print* (System.out.println)** será ignorada e a próxima instrução será executada.

Sintaxe da estrutura condicional

Simple:

```
if ( <condição> ) {
    <comando 1>
    <comando 2>
}
```

Estruturas ELSE

Espero ter deixado evidenciado que a instrução de decisão de seleção única *if*, somente processará a ação indicada se a condição for verdadeira (*true*); caso contrário, a ação é pulada.

Mas se desejarmos definir uma ação para quando a condição for verdadeira e outra ação distinta para quando a condição for falsa, então usaremos a instrução de seleção dupla, *if ... else*, que nos permitirá executar uma ação fora da ordem da sequência

Sintaxe da estrutura condicional

Composta:

```
if ( <condição> ) {  
    <comando 1>  
    <comando 2>  
}  
else {  
    <comando 3>  
    <comando 4>  
}
```

Ou podemos usar o **else if** assim também.

```
if ( condição 1 ) {  
    comando 1  
    comando 2  
}  
else {  
    if ( condição 2 ) {  
        comando 3  
        comando 4  
    }  
    else {  
        comando 5  
        comando 6  
    }  
}
```

Sintaxe opcional: operadores de atribuição cumulativa

Operadores de atribuição cumulativa

<code>a += b;</code>	<code>a = a + b;</code>
<code>a -= b;</code>	<code>a = a - b;</code>
<code>a *= b;</code>	<code>a = a * b;</code>
<code>a /= b;</code>	<code>a = a / b;</code>
<code>a %= b;</code>	<code>a = a % b;</code>

Problema exemplo

Uma operadora de telefonia cobra R\$ 50.00 por um plano básico que dá direito a 100 minutos de telefone. Cada minuto que exceder a franquia de 100 minutos custa R\$ 2.00. Fazer um programa para ler a quantidade de minutos que uma pessoa consumiu, daí mostrar o valor a ser pago.

Entrada	Saída
22	Valor a pagar: R\$ 50.00

Entrada	Saída
103	Valor a pagar: R\$ 56.00

Dia 6 – Estrutura De Condição

O código do exemplo ficaria:

```
import java.util.Locale;

import java.util.Scanner;

public class Main { public static void main(String[] args) {

    Locale.setDefault(Locale.US);

    Scanner sc = new Scanner(System.in);

    int minutos = sc.nextInt();

    double conta = 50.0;

    if (minutos > 100) { conta += (minutos - 100) * 2.0;

    }

    System.out.printf("Valor da conta = R$ %.2f%n", conta);

    sc.close();

    }

}
```

Estrutura switch-case

Quando se tem várias opções de fluxo a serem tratadas com base no valor de uma variável, ao invés de várias estruturas if-else encadeadas, alguns preferem utilizar a estrutura switch-case, o comando break é usado para parar o código.

Sintaxe do switch-case

```
switch ( expressão ) {
    case valor1:
        comando1
        comando2
        break;
    case valor2:
        comando3
        comando4
        break;

    default:
        comando5
        comando6
        break;
}
```


Expressão condicional ternária


Estrutura opcional ao if-else quando se deseja decidir um VALOR com base em uma condição.

Sintaxe:

```
( condição ) ? valor_se_verdadeiro : valor_se_falso
```

Exemplos:

```
( 2 > 4 ) ? 50 : 80       80
```

```
( 10 != 3 ) ? "Maria" : "Alex"       "Maria"
```