

Programação 1

(TP5: *aula 2*)

Estruturas de controlo: decisão

Plano

Estruturas de controlo – decisão (*Livro, pág. 175-180*)

Tipos de dados `boolean`

Operadores relacionais e lógicos (*Livro, pág. 134-135*)

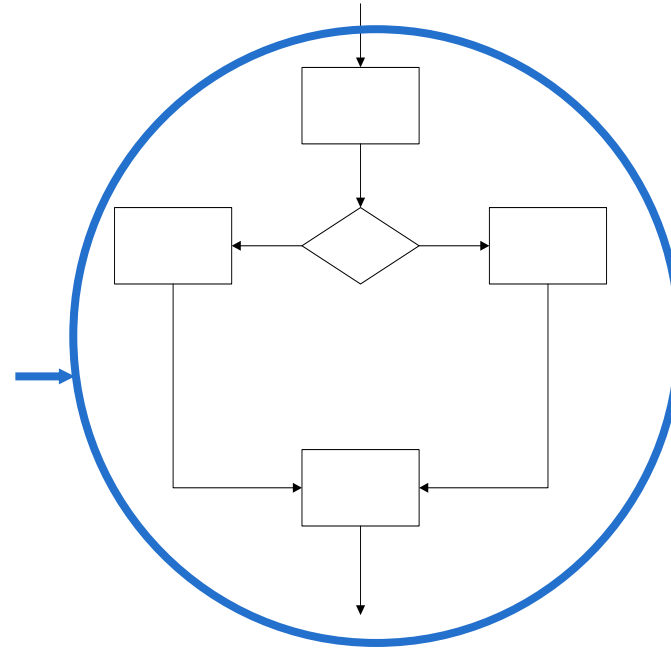
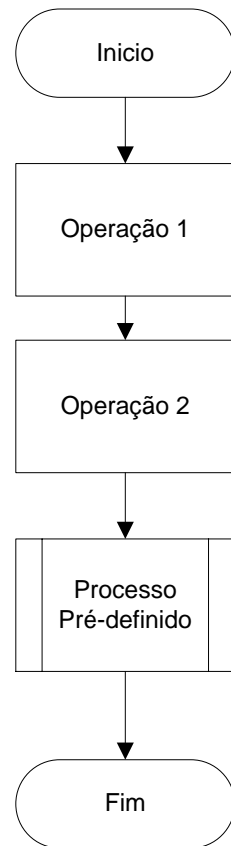
Estrutura de decisão `if`

Estrutura de decisão múltipla `switch`

Estruturas de controlo: decisão

- Um das particularidades de um computador é a capacidade de repetir tarefas ou executar tarefas consoante determinadas condições.
- Para implementar programas mais complexos, temos a necessidade de executar instruções de forma condicional.
- Determinadas instruções só podem/devem ser executadas depois da avaliação de determinadas condições.
- As instruções que permitem condicionar a execução de outras designam-se por **estruturas de controlo**. Nestes slides vamos apresentar as **estruturas de decisão**.
- Temos em JAVA (e na maioria das linguagens) dois tipos de instruções de decisão: `if` e `switch`.

Diagramas de Fluxo – *Flowcharts* (decisão)



Alguns conceitos essenciais...

Tipo de dados `boolean`– Podem assumir os valores `true` e `false` (verdadeiro e falso).

Operadores relacionais: `<`, `<=`, `>`, `>=`, `==`, `!=`

Operadores lógicos: `!`, `||`, `&&`

Exemplos:

```
boolean cond1, cond2, cond3, cond4, cond5;  
cond1 = 3 > 0;           // cond1 fica com true  
cond2 = 5 != 5;          // cond2 fica com false  
cond3 = cond1 || cond2;  // cond3 fica com true  
cond4 = cond1 && cond2;   // cond4 fica com false  
cond5 = !cond4;          // cond5 fica com true
```

Instrução de decisão: if

```
if (condição) instrução;
```

- a condição é avaliada;
- tem que ser uma expressão cujo resultado seja do tipo booleano;
- se verdadeira, é executada a instrução;
- se falsa, o programa continua na linha seguinte;
- exemplo:

```
int x;  
System.out.print("Um valor inteiro:");  
x = sc.nextInt();  
if( x < 0) x = -x;  
System.out.println("O valor absoluto é " + x);
```

Instrução de decisão: `if` (2)

```
if (condição)
{
    bloco1
}
else
{
    bloco2
}
```

- a condição é avaliada;
- se verdadeira, é executado o bloco1;
- se falsa, é executado o bloco2.

Exemplo

```
Scanner kb = new Scanner(System.in);  
double nota;  
System.out.print("Nota do aluno: ");  
nota = kb.nextDouble();  
if (nota >= 9.5) {  
    System.out.println("Aluno aprovado");  
}  
Else {  
    System.out.println("Aluno reprovado");  
}
```


Combinação de vários if

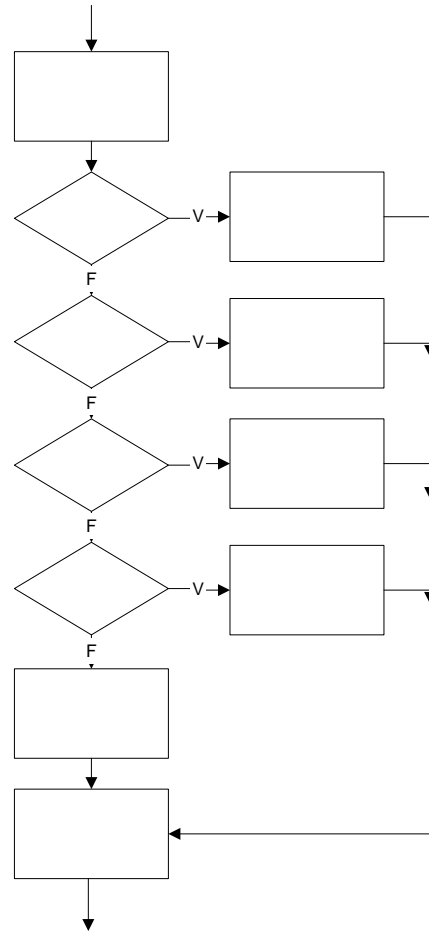
```
if(condiçao1)
{
    if(condição2)
    {
        bloco1;
    }
    else
    {
        bloco2;
    }
}
else
{
    bloco3;
}
```

```
if(condiçao1)
{
    bloco1;
}
else if(condição2)
{
    bloco2;
}
else
{
    bloco3;
}
```

Exemplo

```
if(sexo == 'F' || sexo == 'f'){  
    if(imc <= 18.5)  
        System.out.println("Excessivamente magra");  
    else  
        System.out.println("Normal");  
}  
else if(sexo == 'M' || sexo == 'm'){  
    if(imc < 20)  
        System.out.println("Excessivamente magro");  
    else  
        System.out.println("Jovem ok");  
}  
else{  
    System.out.println("Sexo inválido");  
}
```

Diagramas de Fluxo – *Flowcharts* (decisão múltipla – switch)



Instrução de decisão múltipla `switch`

```
switch (expressão)
{
    case valor1:
        bloco1;
        break;
    case valor2:
        bloco2;
        break;
    default:
        bloco3;
}
```

- As constantes que constituem a lista de alternativas são do mesmo tipo da expressão.
- Primeiro é calculada a expressão e depois o seu valor é pesquisado na lista de alternativas existentes em cada case, pela ordem com que são especificados.
- Se a pesquisa for bem sucedida, o bloco de código correspondente é executado.
- Caso não exista na lista e se o `default` existir, o bloco de código correspondente é executado.
- A execução do `switch` só termina com o aparecimento da instrução `break`.

Exemplo com switch

```
Scanner kb = new Scanner(System.in);
int a, b, r = 0;
int op;
char opChar = ' ';
boolean ok = true;
System.out.print("Operando 1: ");
a = kb.nextInt();
System.out.print("Operando 2: ");
b = kb.nextInt();
System.out.print("Operação: ");
op = kb.nextInt();
switch(op) {
    case 1:
        r = a + b; opChar = '+'; break;
    case 2:
        r = a - b; opChar = '-'; break;
```

```
    case 3:
        r = a * b; opChar = '*'; break;
    case 4:
        if(b != 0){
            r = a / b;
            opChar = '/';
            break;
        }
    default:
        ok = false;
        System.out.println("Operação
        invalida");
}
if(ok) {
    System.out.printf("%d %c %d = %d",
        a, opChar, b, r);}
```

a seguir...

Estruturas de controlo: repetição