

## AULA 5

### ESTRUTURA DE SELEÇÃO OU ESTRUTURA DE DECISÃO

#### ESTRUTURA DE SELEÇÃO SIMPLES

Escreva um algoritmo para calcular a média de um aluno a partir de duas notas. O programa de exibir a média e também uma mensagem informando se o aluno foi aprovado ou reprovado. Sabe-se que o aluno é reprovado se sua média é inferior a 6.

##### Saída:

Quais os resultados que o algoritmo deve fornecer?  
média e mensagem (aprovado ou reprovado).

##### Entrada:

Quais os insumos necessários para obter a saída? notas (n1 e n2)

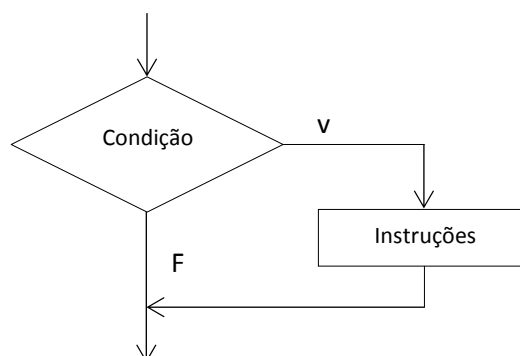
##### Processamento:

Como transformar os insumos na saída?  
 $media = (n1+n2)/2$

Como verificar se foi aprovado ou reprovado?

Estrutura de decisão é um desvio condicional que é usado para decidir se um conjunto de instruções deve ou não ser realizado. Dependendo de um resultado de um teste de condição executa ou não um conjunto de comando ou instruções.

#### Fluxograma de “estrutura de decisão”



Como é que se formula um teste de condição ?

Uma Condição é formulada com uso de operadores relacionais.

= (igual)

== (igual)

< (menor que)

<= (menor ou igual)

> (maior que)

>= (maior ou igual)

<> (diferente)

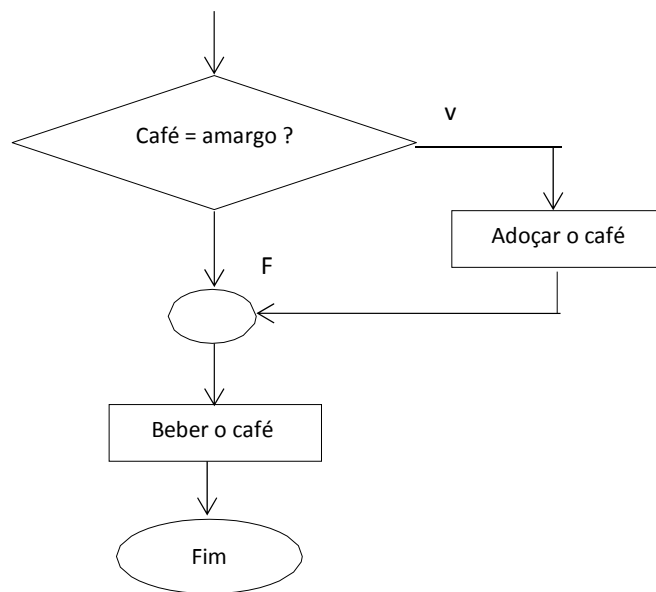
!= (diferente)

**Desvio condicional simples** uma condição é avaliada e se o resultado for verdadeiro um bloco de instrução é executado.

Exemplo: Verificar se o café está amargo para tomar ou não a decisão de adoçar.

##### Condição:

Café = amargo



Fluxograma é uma linguagem gráfica de se representar instruções. Podem ser utilizados em diversas áreas para facilitar a compreensão de um processo complexo.

Em programação o fluxograma podem ser ferramentas uteis para principalmente para representar decisões complexas.

É importante saber que a figura de LOSANGO é utilizada para TESTE DE CONDIÇÃO. E que o RETÂNGULO para BLOCOS DE INSTRUÇÕES.

As SETAS orientam o fluxo de execução do programa.

CIRCULOS PEQUENOS para conectar fluxos.

Como representar de forma textual uma estrutura de seleção simples ou desvio condicional simples na linguagem PORTUGOL ou pseudocódigo?

se (condição) entao

    instrução 1

    instrução 2

    instrução 3

    ...

    Instrução n

fimse

Em outras linguagem o ENTAO equivale o **ABRE-CHAVE**: { , no VISUALG usa-se o ENTAO.

Assim o programa completo para exibir a média do aluno e se ele foi aprovado ou reprovado é:

Inicialmente declara as variáveis:

Media, nota1, nota2: real

Em seguida as instruções de entradas de dados:

escreva("Digite a primeira nota: ")

leia(nota1)

escreva("Digite a segunda nota: ")

leia(nota2)



**Entrada de dados**

Temos as instruções de entradas de dados **leia(nota1)** e **leia(nota2)**.

Conforme aprenderam antes de cada leia recomenda-se enviar uma mensagem de orientação ao usuário indicando o que ele deve digitar. Por isso temos escreva("Digite a primeira nota:") e escreva("Digite a segunda nota:").

Na sequência temos o cálculo da média através da expressão  $media = (nota1 + nota2)/2$ . Note que a expressão está com parentes para indicar a prioridade do cálculo das duas notas antes da divisão. A soma precisa ser feita antes da divisão.

A próxima instrução é a primeira SAÍDA de DADOS:

Escreva("Média do aluno: ", media)

Ou seja, a primeira exibição de dado, utilizando o comando escreva.

Por fim, temos a estrutura de seleção com teste de condição

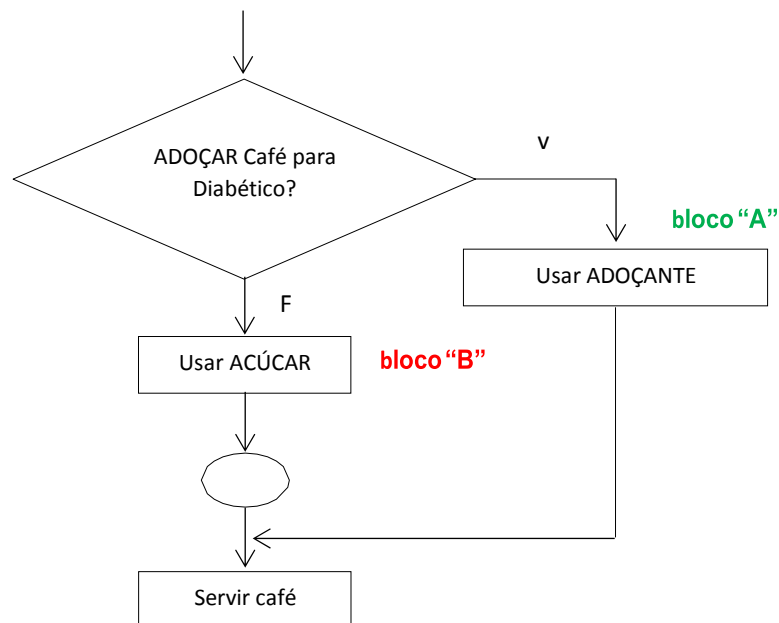
```
se (media >= 6) então  
    escreva(" aprovado ")  
fimse
```

Sendo o resultado de teste de condição verdadeiro a mensagem " aprovado " será impresso por meio do comando ESCREVA.

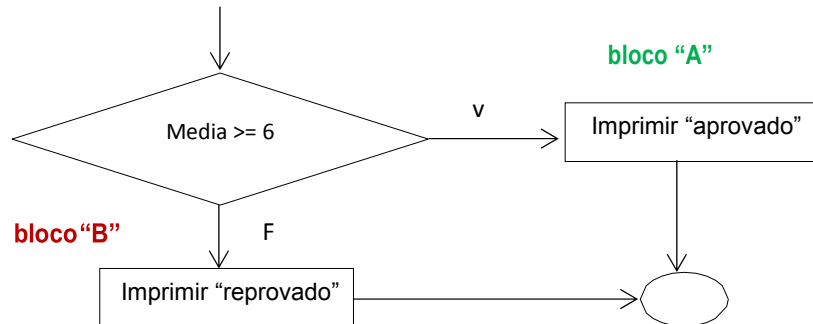
### ESTRUTURA DE SELEÇÃO COMPOSTA

No exemplo anterior Café igual amargo, dependendo do resultado o bloco adoçar café ou será executado ou desviado. No entanto podemos nos deparar com situações em que haja mais de um bloco para ser executado ou desviado de acordo com o resultado do teste de condição, neste caso precisamos fazer uso de estrutura de seleção composta.

Por exemplo se formos servir café para uma pessoa, devemos antes verificar se a mesma é ou não diabética.



De acordo com o fluxograma apresentado se o teste de condição for verdadeiro o bloco “A” Usar ADOÇANTE será executado, caso o resultado do teste seja falso o bloco “B” com a instrução usar ACÚCAR será executado. Independentemente de se executar o Bloco “A” ou o bloco “B” a instrução “Servir café” será executada na sequencia como ponto dos dois blocos. Temos exemplo com a possibilidade de um ou outro bloco ser executado, isso é o que chamamos de DESVIO CONDICIONAL COMPOSTO ou ESTRUTURA DE SELEÇÃO COMPOSTA.



Veja, como representar uma estrutura de seleção composta em representação textual em linguagem portugal ou pseudocódigo:

```

se (condição) entao
    instrução A1
    instrução A2
    instrução A3
    ...
    Instrução An
senao
    instrução B1
    instrução B2
    instrução B3
    ...
    Instrução Bn
fimse
  
```

Exemplo como aplicar em linguagem de programação qualquer, como **linguagem C**:

```

if (condição) {
    instrução A1
    instrução A2
    instrução A3
    ...
    Instrução An
} else {
    instrução B1
    instrução B2
    instrução B3
    ...
    Instrução Bn
}
  
```

Conforme exemplo da média em portugal:

```

Se (media>=6) entao
  
```

```

        escreva("aprovado")
senao
        escreva("reprovado")
fimse

```

Assim, para programa completo para imprimir a media e a mensagem se ele foi aprovado ou reprovado é a seguinte:

Inicialmente declara as variáveis:  
 nota1, nota2, media: real

Em seguida as instruções de entradas de dados:

```

escreva("Digite a primeira nota: ")
leia(nota1)
escreva("Digite a segunda nota: ")
leia(nota2)

```

} **Entrada de dados**

Na sequencia temos o calculo da média pela expressão:  
 $media = (nota1 + nota2)/2$

A próxima instrução é a primeira saída de dados ou seja:

```

escreva("Média do aluno: ",media)

```

Por fim a estrutura de seleção com o teste de seleção:

```

se(media>=6) entao
    escreva("aprovado")
senao
    escreva("reprovado")
fimse

```

Se o teste de condição for verdadeiro, ou seja, maior ou igual a 6 imprime aprovado. Caso contrário, ou seja, se media menor que 6 imprime reprovado. Simulando a execução do programa para um teste de condição falso, teremos o seguinte:

nota1, nota2, media: real

```

escreva("Digite a primeira nota: ")
leia(nota1)
escreva("Digite a segunda nota: ")
leia(nota2)

```

```

media = (nota1 + nota2)/2
escreva("Média do aluno: ",media)

```

```

se(media>=6) entao
    escreva("aprovado")
senao
    escreva("reprovado")
fimse

```



nota1	nota2
4.0	6.0

media
5.0

Média do aluno: 5.0
---------------------

reprovado
-----------

Conforme exemplo de estrutura de seleção composta, ela pode ser usada sempre que houver dois blocos de instruções distintos e excludentes entre si, ou seja, apenas um deles será executado a partir do resultado de um teste de condição.

#### **Exercício de estrutura de seleção composta:**

Escreva um programa para calcular o peso ideal de uma pessoa, utilizando as seguintes formulas:

- para homens  $(72.7 * altura) - 58$
- para mulheres  $(62.1 * altura) - 47.7$

Aplicando a técnica de interpretação de enunciados vamos identificar qual saída de dados, ou seja, quais resultados o algoritmo deve fornecer.

**SAÍDA:** Quais os resultados que o algoritmo deve fornecer?

Escreva um programa para calcular o **peso ideal** de uma pessoa

**SAÍDA:** pesoIdeal

Assim o resultado que o algoritmo deve fornecer é o peso ideal.

O próximo passo é identificar a entrada de dados:

**ENTRADA:** Quais os insumos necessários para obter a saída.

O enunciado não explicita claramente a entrada de dados, portanto fornece duas fórmulas para o peso ideal, uma para homens e outra para mulheres. Percebemos que existe uma incógnita que é a altura que desconhecemos. Portanto é necessário conhecer a altura da pessoa para calcular seu peso ideal.

Assim, uma entrada obrigatória é altura.

**ENTRADA de DADOS:**

ALTURA

Como foi informado duas fórmulas, o programa deve saber quando usar uma ou outra.

Como, essas fórmulas são para o peso ideal para homens e outra para peso ideal para mulheres, deve-se informar o sexo da pessoa ao programa. Ou seja, o sexo também é uma entrada de dados.

**ENTRADA de DADOS:**

ALTURA e SEXO

Por fim, vamos identificar os passos necessários para a etapa do processamento para transformar os insumos, ou seja, as entradas de dados em saídas de dados. Como foi fornecido as fórmulas, basta aplica-las de acordo com o sexo informado.

Devemos fazer uso de uma estrutura de seleção.

**SAÍDA:** pesoIdeal

**ENTRADA:** altura e sexo

**PROCESSAMENTO:**

se (sexo="M") entao

    pesoIdeal =  $(72.7 * altura) - 58$

senao

    pesoIdeal =  $(62.1 * altura) - 47.7$

fimse

Assim o programa completo para calculo do peso ideal de uma pessoa seria o seguinte:

**DECLARAÇÃO de VARIÁVEIS**

Altura, pesoIdeal: real

Sexo: caracter

Lembrando que tipos de dados CHARACTER é utilizado para armazenar letras do alfabeto, dígitos e caracteres especiais.

Sendo a altura para receber a entrada de dado fornecida pelo usuário e peso ideal para armazenar o resultado do cálculo.

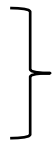
Em seguida:

```
escreva("Digite sexo [M ou F]: ")
```

```
leia(sexo)
```

```
escreva("Digite a altura: ")
```

```
leia(altura)
```



Temos as instruções de entrada de dados `leia(sexo)` e `leia(altura)`.

**LEMBRANDO** que antes de cada "leia" devemos passar uma mensagem ao usuário indicando o que ele deve digitar.

Por isso temos essas instruções acima, escreva e leia.

```
se (sexo="M") entao
```

```
    pesoIdeal = (72.7 * altura) - 58
```

```
senao
```

```
    pesoIdeal = (62.1 * altura) - 47.7
```

```
fimse
```

Mais adiante veremos como obrigar o usuário digitar M ou F.

Lembrando que em português estruturado, português ou pseudocódigo no teste de condição (`sexo = "M"`) no VisualG usa o operador de atribuição (`=`), em outros software ou linguagem de programação usa-se o operador de comparação (`==`).

Observa-se que no teste de condição (`sexo = "M"`) as aspas duplas ou simples são necessárias pois SEXO é uma variável do tipo caracter. Se não tivéssemos comparando o conteúdo da variável SEXO com uma variável M, nesse caso do exemplo M não é uma variável, mas uma constante para representar o sexo masculino.

A próxima instrução é a saída de dados, ou seja a saída do resultado:

```
escreva("Seu peso ideal é: ", pesoIdeal)
```

Note que "Seu peso ideal" esta entre aspas duplas, que será exibido exatamente dessa forma. Já pesoIdeal esta fora das aspas e separado por virgula, no caso servindo de concatenação entre a string e a variável pesoIdeal, sendo esta substituída pelo conteúdo que nesse momento terá o resultado do cálculo.

Simulando a execução do programa teremos o seguinte:

Declaração de Variáveis:

altura, pesoIdeal: real

sexo: caracter

```
escreva("Digite sexo [M ou F]: ")
```

```
leia(sexo)
```

```
escreva("Digite a altura: ")
```

sexo	pesoIdeal	altura
M		1.75

leia(altura)

se (sexo="M") então

    pesoldeal = (72.7 \* altura) – 58

senao

    pesoldeal = (62.1 \* altura) – 47.7

fimse

escreva("Seu peso ideal é: ",pesoldeal)

pesoldeal

69.22

Seu peso ideal é: 69.22

### ESTRUTURAS DE SELEÇÃO ENCADEADA

O uso de estruturas de seleção encadeada envolve normalmente problemas com diversas condições.

Veja a necessidade de seu uso a partir do seguinte problema:

Escreva um algoritmo para imprimir o conceito final de um aluno a partir de uma média. A tabela de conversão a ser utilizada é a seguinte:

Conceito "A" se media >= 9	Conceito "C" se media >= 6
Conceito "B" se media >= 7	Conceito "P" se media < 6

Aplicando a técnica de interpretação de enunciado:

#### **SAÍDA:**

Quais os resultados que o algoritmo deve fornecer?

SAÍDA: Conceito final

#### Próximo passo **ENTRADA:**

Quais os insumos necessários para se obter a saída?

Precisa conhecer a média do aluno. Dá para supor que a media será fornecida automaticamente pelo programa, mas vamos fazer que seja fornecida pelo usuário diretamente.

#### **ENTRADA de DADOS:**

Media

Passos para obter os insumos necessários para saída de dados:

Conceito "A" se media >= 9;

Conceito "B" se media >= 7;

Conceito "C" se media >= 6;

Conceito "P" se media < 6;

Estrutura de seleção:

se (media>=9) então

    escreva("Conceito A")

fimse

se (media>=7) então

    escreva("Conceito B")

fimse

Se (media>=6) então



```
    escreva("Conceito C")
fimse
```

```
se (media<6) entao
    escreva("Conceito P")
fimse
```

**Simulando a execução do programa teremos o seguinte:**

Declaração de Variáveis

Media: real

Instruções de Entrada de dados:

```
escreva("Digite a media: ") // mensagem de orientação ao usuário para digitar o valor da média
leia(media)
```

Na sequencia temos as estrutura de seleção com os teste de condição.

```
se (media>=9) entao
    escreva("Conceito A")
fimse
```

```
se (media>=7) entao
    escreva("Conceito B")
fimse
```

```
Se (media>=6) entao
    escreva("Conceito C")
fimse
```

```
se (media<6) entao
    escreva("Conceito P")
fimse
```

Simulando a execução do programa qual conceito obteríamos como saída, se o usuário digitar valor 9.5 para media ?

A resposta teria que ser Conceito A, no entanto a solução imprime três mensagens:

```
Digite a media: 9.5
Conceito A
Conceito B
Conceito C
```

} Resultados de saída

Assim executa os três primeiros blocos de instruções devido ser verdadeiro. Isso ocorre devido ao uso de quatro estruturas de seleção simples ao invés de seleção composta com senao. Outra possibilidade seria fazer uso de seleção composta com operadores lógicos verificando os intervalos.

Por exemplo:

```
se (media>=9) entao
    escreva("Conceito A")
fimse
se (media>=7 e media <9) entao
    escreva("Conceito B")
fimse
```

```

se (media >= 6 e media < 7) entao
    escreva("Conceito C")
fimse
se (media < 6) entao
    escreva("Conceito P")
fimse

```

Com essa estrutura somente o conceito "A" será exibido o qual está correto.  
 No entanto essa estrutura ainda não é conveniente. Ela é eficaz mas ainda não é ideal.  
 Nesse caso como temos vários SEs precisamos de seleção encadeada.  
 ESTRUTURAS DE SELEÇÃO ENCADEADAS são sequências de estruturas SE...ENTAO...SENAO...  
 usado para testes de múltiplos casos encadeados.

```

se (media >= 9) entao
    escreva("Conceito A")
senao
    se (media >= 7) entao
        escreva("Conceito B")
    senao
        se (media >= 6) entao
            escreva("Conceito C")
        senao
            escreva("Conceito P")
        fimse
    fimse
fimse

```

Escreva um algoritmo (em português, pseudocódigo) para ler o preço de etiqueta de um produto e calcular o preço a pagar pelo cliente de acordo com a seguinte tabela:

CÓDIGO	CONDIÇÃO DE PAGAMENTO	DESCONTO
0	A VISTA	25%
1	Cheque (30 dias)	20%
2	Cartão de Crédito (2x)	10%
3	Cartão de Crédito (3x)	5%
outra	Negociada com vendedor	Sem desconto

SAÍDA: preço a pagar

ENTRADA: Quais os insumos necessários para obter a saída?

Preço de etiqueta do produto [precoOriginal] e código do tipo de pagamento

```

se (codigo=0) entao
    precoPagar = precoOriginal * 0.75
senao
    se (codigo=1) entao
        precoPagar = precoOriginal * 0.80
    senao
        .
        .
        .
fimse

```