



## UNIDADE II

---

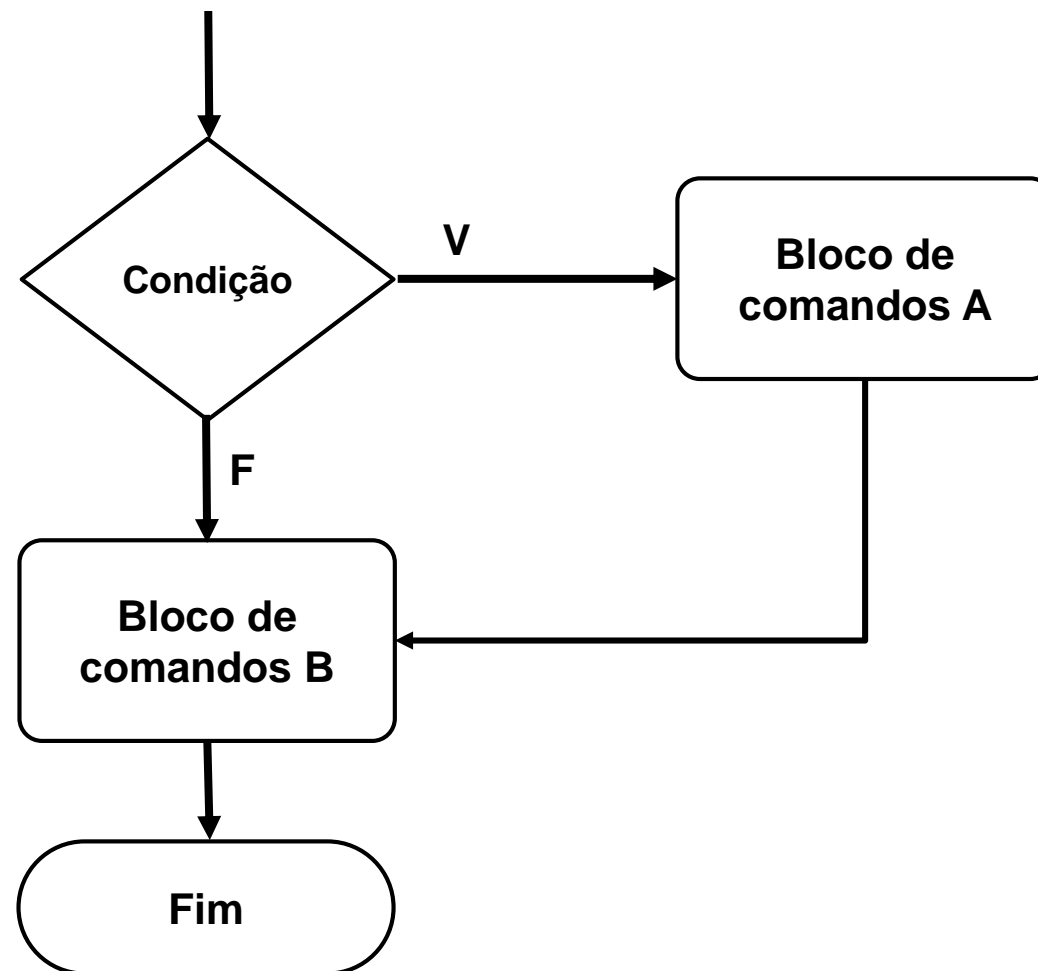
### Lógica de Programação e Algoritmos

Profa. Ma. Eliane Santiago

# Estrutura de Decisão Simples

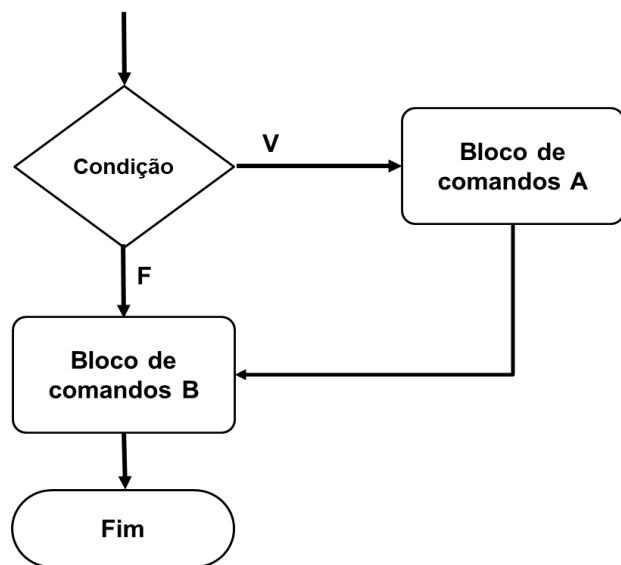
- O comando SE... ENTÃO é uma estrutura de decisão que avalia uma condição e, se ela for verdadeira, promove um desvio no curso de execução dos comandos, voltando ao bloco principal.

```
:  
se (condição1 = verdadeira) então  
    //faça alguma coisa  
fimse  
:
```



# Estrutura de Decisão Simples

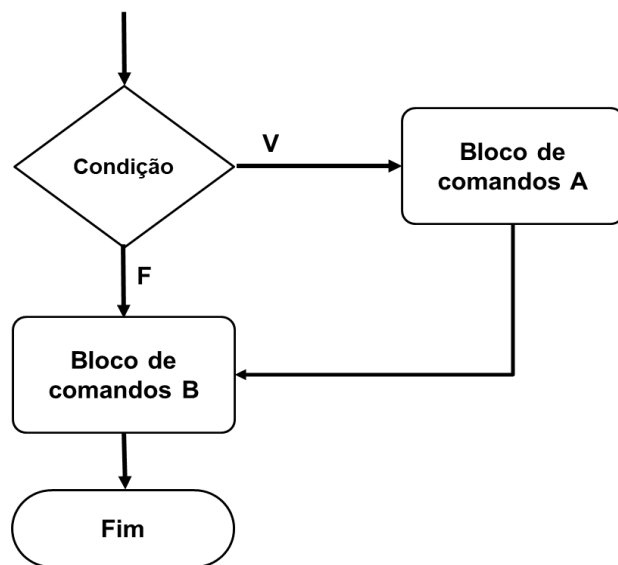
- O comando SE... ENTÃO é uma estrutura de decisão que avalia uma condição e, se ela for verdadeira, executa um comando ou bloco de comandos.



```
1. Algoritmo "Estrutura de Decisão Simples"
2. Var
3.     x, y : inteiro
4. Inicio
5.     //entrada
6.     escreva("Digite um valor para X: ")
7.     leia(x)
8.     escreva("Digite um valor para Y: ")
9.     leia(y)
10.    //processamento
11.    se (x mod y == 0) entao
12.        escreva(x, " é divisível por ", y)
13.    fimse
14.    Fimalgoritmo
```

# Estrutura de Decisão Simples

- O comando SE... ENTÃO é uma estrutura de decisão que avalia uma condição e, se ela for verdadeira, executa um comando ou bloco de comandos.

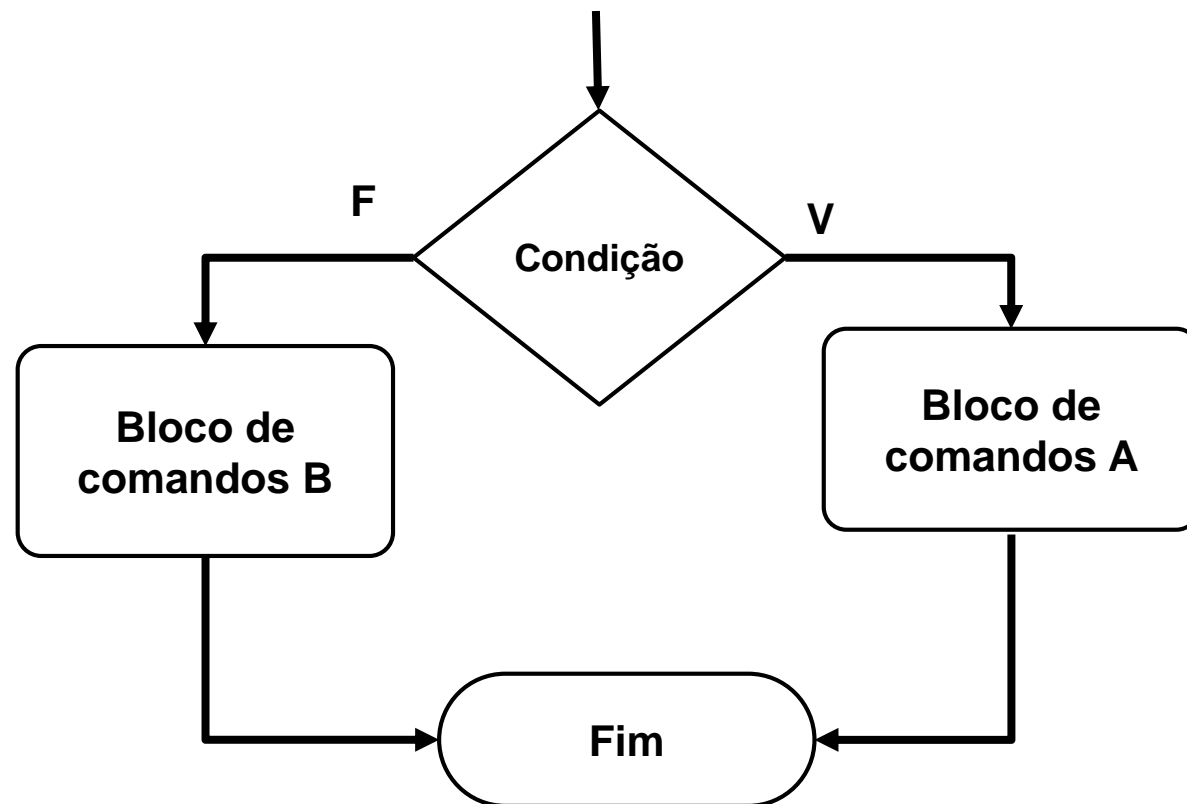


```
1. Algoritmo "Estrutura de Decisão Simples"
2. Var
3.     operacao : char
4.     x, y, r : inteiro
5.
6. Inicio
7.     //entrada
8.     escreval("Digite um valor para X: ")
9.     leia(x)
10.    escreval("Digite um valor para Y: ")
11.    leia(y)
12.    escreval("Qual operação deseja realizar?")
13.    leia(operacao)
14.
15.    //processamento
16.    se (operacao = "+") então
17.        r <- x+y
18.        escreva(x, " + ", y, " = ", r)
19.    fimse
20.    se (operacao = "-") então
21.        r <- x-y
22.        escreva(x, " - ", y, " = ", r)
23.    fimse
24. Fimalgoritmo
```

# Estrutura de Decisão Simples

- O comando SE... ENTÃO é uma estrutura de decisão que avalia uma condição e, se ela for verdadeira, promove um desvio para um comando ou bloco de comandos que deve ser executado.

```
:  
se (condição1 = verdadeira) então  
    //faça alguma coisa  
senão  
    //faça outra coisa  
fimse  
:
```



# Estrutura de Decisão Composta

O número é par ou ímpar?

```
1. Algoritmo "Estrutura de Decisão Composta"
2. Var
3.     x : inteiro
4. Inicio
5.     //entrada
6.     escreva("Digite um valor para X: ")
7.     leia(x)
8.
9.     //processamento
10.    se (x mod 2 = 0) entao
11.        escreva(x, " eh par")
12.    senão
13.        escreva(x, " eh impar" )
14.    fimse
15. Fimalgoritmo
```

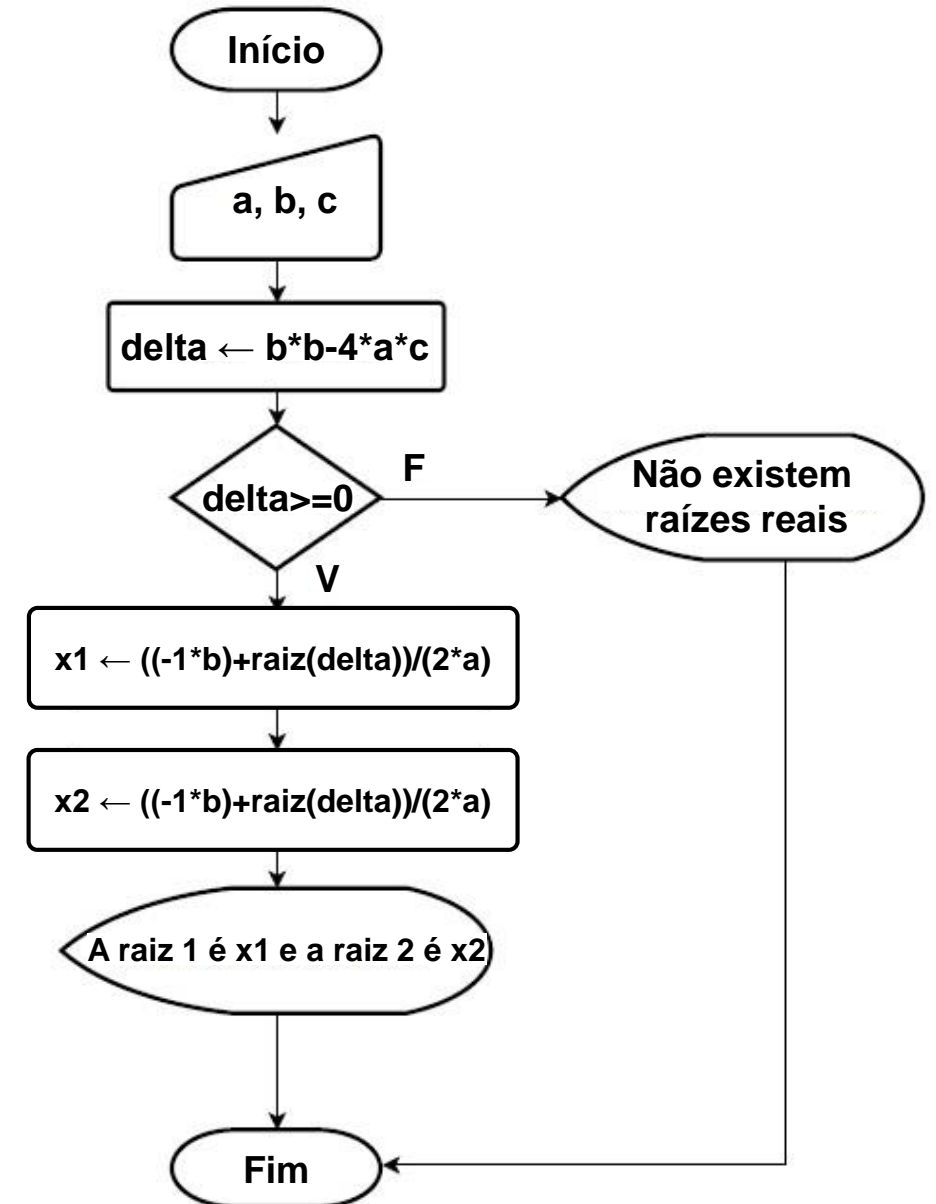
# Estrutura de Decisão Composta

Como estruturar um algoritmo para resolver a equação de segundo grau?

$$f(x) = x^2 - 3x - 10$$

$$\Delta = b^2 - 4.a.c$$

$$x = \frac{-b \pm \sqrt{\Delta}}{2.a}$$



# Algoritmo para Resolução da Equação de 2º grau

$$\Delta \geq 0 ?$$

Se sim, calcule as raízes

Se não, escreva “não existem raízes reais”

```
1. Algoritmo "Equação de Segundo Grau"
2. Var
3.   a, b, c, delta, x1, x2 : real
4. Inicio
5.   //entrada
6.   escreval("A: ")
7.   leia(a)
8.   escreval("B: ")
9.   leia(b)
10.  escreval("C: ")
11.  leia(c)
12.
13.  //processamento
14.  delta <- b*b-4*a*c
15.
16.  se (delta>=0) entao
17.    x1 <- (-1*b+raiz(delta))/(2*a)
18.    x2 <- (-1*b-raiz(delta))/(2*a)
19.    escreval("A raiz 1 é ", x1, "e a raiz 2 é ", x2)
20.  senao
21.    escreval("Não existem raízes reais")
22.  fimse
23. Fimalgoritmo
```



# As estruturas de decisão ajudam a especificar regras

## Qual é o dia do rodízio?

O algoritmo em descrição narrativa para descrever a regra conforme a tabela é

### Algoritmo em descrição narrativa

leia a placa do veículo  
extraia o último dígito  
se é final 1 ou 2 então escreva “dia do rodízio: segunda-feira”  
senão se é final 3 ou 4 então escreva “dia do rodízio: terça-feira”  
senão se é final 5 ou 6 então escreva “dia do rodízio: quarta-feira”  
senão se é final 7 ou 8 então escreva “dia do rodízio: quinta-feira”  
senão escreva “dia do rodízio: sexta escreva “dia do rodízio: sexta-feira”

DIA DA SEMANA	FINAL DA PLACA
Segunda-feira	1 e 2
Terça-feira	3 e 4
Quarta-feira	5 e 6
Quinta-feira	7 e 8
Sexta-feira	9 e 0

# As estruturas de decisão ajudam a especificar regras

## Emissão do título de eleitor

### Algoritmo em descrição narrativa

leia nacionalidade, idade, sexo, situação com o serviço militar, país de residência, débito com a Justiça Eleitoral, foto

se é cidadão brasileiro, então

se idade é maior ou igual a 16 anos, então

se tem foto 3x4 então verificar se o tamanho é menor ou igual a 10MB.

se sexo masculino e idade maior 18 e idade menor 45 anos, então apresentar quitação do serviço militar.

se tem débito com a Justiça Eleitoral, então apresentar comprovante de quitação.

se residente no Brasil, então apresentar  
comprovante de residência recente.

# Interatividade

Fornecidas as regras da avaliação preliminar de potenciais doadores sangue, em termos de idade e peso, o trecho de algoritmo, em pseudocódigo, que verifica corretamente se os dados se enquadram nas restrições fornecidas é:

- Idade entre 18 e 69 anos
  - Peso acima de 50 kg
- a) se ((idade>=18) ou (idade<=69)) e ((peso >=50) e (peso<130)) então  
    escreva("Avaliação preliminar ok")  
    senão  
        escreva ("Candidato não apto para doação ")
- b) se ((18 <= idade <= 69)) ou (peso >= 50)) então  
    escreva("Avaliação preliminar ok")  
    senão  
        escreva ("Candidato não apto para doação ")
- c) se (((idade >=18) e (idade <=69)) e (peso > 50)) então  
    escreva("Avaliação preliminar ok")  
    senão  
        escreva ("Candidato não apto para doação ")

# Interatividade

Fornecidas as regras da avaliação preliminar de potenciais doadores sangue, em termos de idade e peso, o trecho de algoritmo, em pseudocódigo, que verifica corretamente se os dados se enquadram nas restrições fornecidas é:

- Idade entre 18 e 69 anos
- Peso acima de 50 kg

d) se (((idade>=18) e (idade<=69)) ou ((peso >=50) e (peso<130))) então  
    escreva("Avaliação preliminar ok")  
    senão  
        escreva ("Candidato não apto para doação ")

e) se (((idade >=18) ou (idade <=69)) ou (peso > 50)) então  
    escreva("Avaliação preliminar ok")  
    senão  
        escreva ("Candidato não apto para doação ")

# Resposta

Fornecidas as regras da avaliação preliminar de potenciais doadores sangue, em termos de idade e peso, o trecho de algoritmo, em pseudocódigo, que verifica corretamente se os dados se enquadram nas restrições fornecidas é:

- Idade entre 18 e 69 anos
- Peso acima de 50 kg

c) se (((idade >=18) e (idade <=69)) e (peso > 50)) então  
    escreva("Avaliação preliminar ok")  
    senão  
        escreva ("Candidato não apto para doação ")

# Estrutura de Decisão Encadeada

- O comando SE... SENÃO é uma estrutura de decisão que avalia uma condição e, se ela for verdadeira, promove um desvio para o comando ou bloco de comandos que deve ser executado.

```
:  
se (condição1 = verdadeira) então  
    //faça alguma coisa  
senão  
    se (condição2 = verdadeira) então  
        //faça alguma coisa  
    senão  
        se (condição3 = verdadeira) então  
            //faça alguma coisa  
        senão  
            se (condição4 = verdadeira) então  
                //faça alguma coisa  
            fimse  
        fimse  
    fimse  
fimse
```

# Estrutura de Decisão Encadeada

- O comando SE... SENÃO é uma estrutura de decisão que avalia uma condição e, se ela for verdadeira, promove um desvio para o comando ou bloco de comandos que deve ser executado.

```
:  
se (condição1 = verdadeira) então  
| //faça alguma coisa  
senão  
|  
| se (condição2 = verdadeira) então  
| | //faça alguma coisa  
| | senão  
| | | se (condição3 = verdadeira) então  
| | | | //faça alguma coisa  
| | | | senão  
| | | | | se (condição4 = verdadeira) então  
| | | | | | //faça alguma coisa  
| | | | | fimse  
| | | | fimse  
| | | fimse  
| | fimse  
| fimse
```

# Estrutura de Decisão Encadeada

- O comando SE... SENÃO pode ser encadeado para testar várias condições.

```
1. Algoritmo "Triangulo"
2. Var
3.     L1, L2, L3 : real
4. Inicio
5.     //entrada
6.     escreval("L1: ")
7.     leia(L1)
8.     escreval("L2: ")
9.     leia(L2)
10.    escreval("L3: ")
11.    leia(L3)
12.    //processamento
13.    se ((L1<(L2+L3)) e (L2<(L1+L3)) e (L3<(L1+L2))) entao
14.        //se é triângulo, então verifique qual o tipo
15.        se ((L1=L2) e (L2=L3))
16.            escreva("Triagulo Equilátero")
17.        senão
18.            se (((L1=L2) e (L1<>L3)) ou ((L1=L3) e (L1<>L2)) ou
                ((L2=L3) e (L2<>L1)))
19.                escreva("Triagulo Isósceles")
20.            senão
21.                escreva("Triagulo Escaleno")
22.            fimse
23.        fimse
24.    senão //se não é triângulo
25.        escreva("Não é triângulo")
26.    fimse
25. Fimalgoritmo
```



# Qual o seu índice de massa corpórea (IMC)

Resultado	Situação
Abaixo de 17	Muito abaixo do peso
Entre 17 e 18,50	Abaixo do peso
Entre 18,50 e 25	Peso normal
Entre 25 e 30	Acima do peso
Entre 30 e 35	Obesidade I
Entre 35 e 40	Obesidade II (severa)
Acima de 40	Obesidade III (mórbida)

```
1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.
18.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.
34.
35.
36.
37.
38.
39.
40.

Algoritmo "IMC"
Var
    altura, massa, imc : real
Inicio
    //entrada
    escreva("Peso.....: ")
    leia(massa)
    escreva("Altura....: ")
    leia(altura)

    //processamento
    imc <- massa/(altura*altura)

    //saida
    se (imc<17) então
        escreva("Muito abaixo do peso")
    senão
        se ((imc>=17) e (imc<=18.5)) então
            escreva("Abaixo do peso")
        senão
            se ((imc>18.5) e (imc<25)) então
                escreva("Peso Normal")
            senão
                se ((imc>=25) e (imc<=30)) então
                    escreva("Acima do peso")
                senão
                    se ((imc>30) e (imc<35)) então
                        escreva("Obesidade I")
                    senão
                        se ((imc>=35) e (imc<=40)) então
                            escreva("Obesidade II")
                        senão
                            escreva("Obesidade III")
                    fimse
                fimse
            fimse
        fimse
    fimse
Fimalgoritmo
```

# Estruturas de decisão encadeadas podem ser agrupadas com o operador E lógico

```
Se (condição1 = verdadeira) então  
    Se (condição2 = verdadeira) então  
        //faça alguma coisa  
    fimse  
fimse
```

```
Se (condição1 = verdadeira) e (condição2 = verdadeira) então  
    //faça alguma coisa  
fimse
```

# Leia três números e escreva-os em ordem crescente

## Entradas possíveis

N1	N2	N3
1	2	3
1	3	2
2	1	3
2	3	1
3	1	2
3	2	1

```
1.  Algoritmo "Ordenar_3_Numeros"
2.  Var
3.  n1, n2, n3: inteiro
4.  Inicio
5.      escreva("Número 1: ")
6.      leia(n1)
7.      escreva("Número 2: ")
8.      leia(n2)
9.      escreva("Número 3: ")
10.     leia(n3)
11.
12.     se (n1<n2) e (n1<n3) e (n2<n3) então
13.         escreva(n1, " < ", n2, " < " , n3)
14.     senão
15.         se (n1<n2) e (n1<n3) e (n3<n2) então
16.             escreva(n1, " < ", n3, " < " , n2)
17.         senão
18.             se (n2<n1) e (n2<n3) e (n1<n3) então
19.                 escreva(n2, " < ", n1, " < " , n3)
20.             senão
21.                 se (n2<n1) e (n2<n3) e (n3<n1) então
22.                     escreva(n2, " < ", n3, " < " , n1)
23.                 senão
24.                     se (n3<n1) e (n3<n2) e (n1<n2) então
25.                         escreva(n3, " < ", n1, " < " , n2)
26.                     senão
27.                         escreva(n3, " < ", n2, " < " , n1)
28.             fimse
29.         fimse
30.     fimse
31. fimse
32. fimse
33. fimalgoritmo
```

# Interatividade

Considere o algoritmo dado abaixo e responda qual será o conceito atribuído caso as notas do aluno sejam 9,5 e 5,5 para n1 e n2, respectivamente.

- a) Insuficiente.
- b) Regular.
- c) Bom.
- d) Muito bom.
- e) Excelente.

```
1. Algoritmo "Conceito de Aproveitamento Escolar"
2. Var
3.   n1, n2, media : real
4. Inicio
5.   leia(n1, n2)
6.   media <- (n1+n2)/2
7.   se ((media>=0) e (media<4)) entao
8.     escreva("Insuficiente")
9.   senão
10.    se ((media>=4) e (media<6)) entao
11.      escreval("Regular")
12.    senao
13.      se ((media>=6) e (media<7,5)) entao
14.        escreval("Bom")
15.      senão
16.        se ((media>=7,5) e (media<9)) então
17.          escreval("Muito Bom")
18.        senão
19.          escreval("Excelente")
20.        fimse
21.      fimse
22.    fimse
23.  fimse
24. Fimalgoritmo
```

# Resposta

Considere o algoritmo dado abaixo e responda qual será o conceito atribuído caso as notas do aluno sejam 9,5 e 5,5 para n1 e n2, respectivamente.

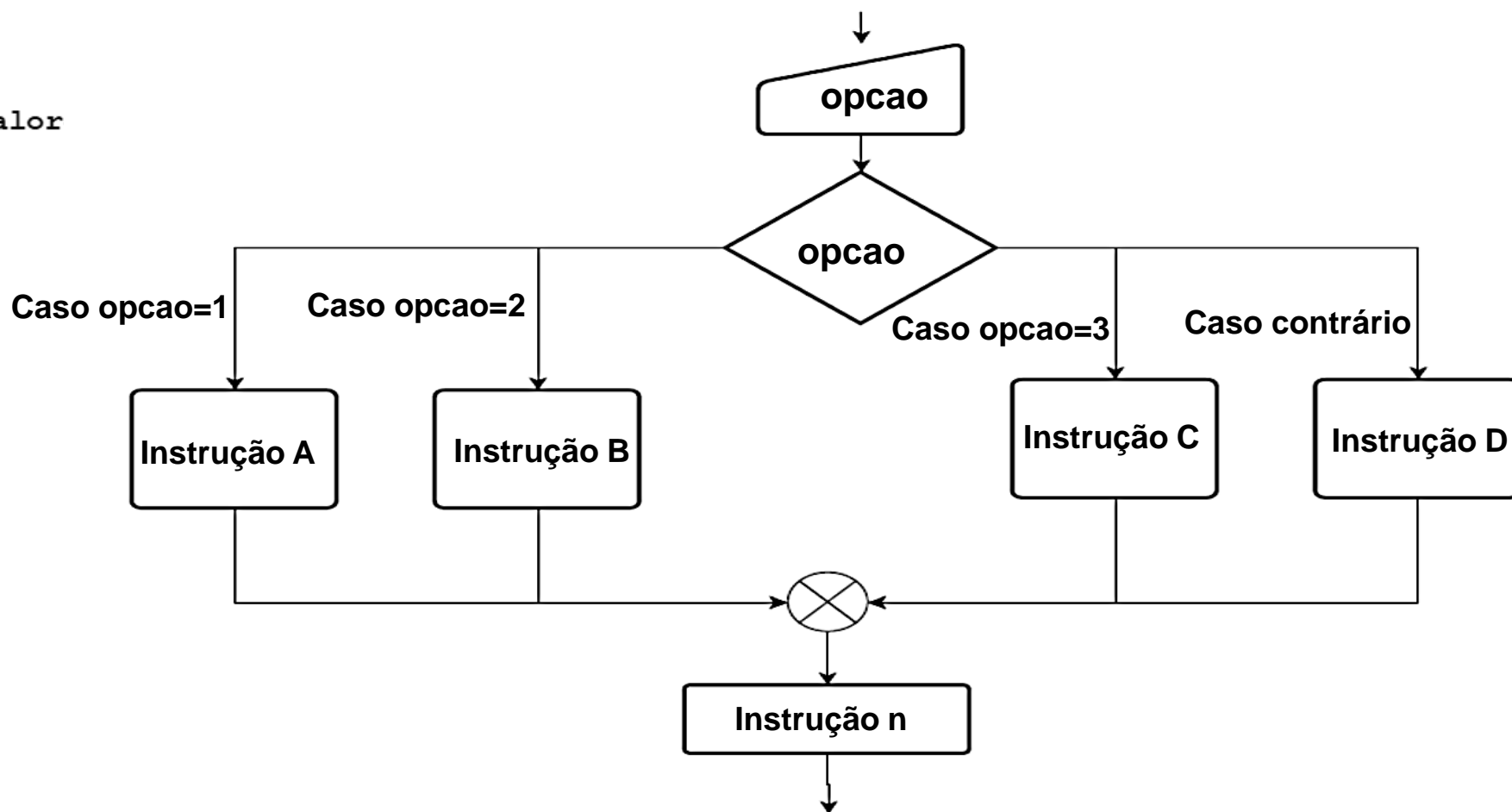
- a) Insuficiente.
- b) Regular.
- c) Bom.
- d) **Muito bom.**
- e) Excelente.

```
1. Algoritmo "Conceito de Aproveitamento Escolar"
2. Var
3.   n1, n2, media : real
4. Inicio
5.   leia(n1, n2)
6.   media <- (n1+n2)/2
7.   se ((media>=0) e (media<4)) entao
8.     escreva("Insuficiente")
9.   senão
10.    se ((media>=4) e (media<6)) entao
11.      escreval("Regular")
12.    senao
13.      se ((media>=6) e (media<7,5)) entao
14.        escreval("Bom")
15.      senão
16.        se ((media>=7,5) e (media<9)) então
17.          escreval("Muito Bom")
18.        senão
19.          escreval("Excelente")
20.        fimse
21.      fimse
22.    fimse
23.  fimse
24. Fimalgoritmo
```

# Escolha Caso

- A estrutura ESCOLHA-CASO oferece uma forma organizada para agrupar os comandos. A lógica consiste em verificar o valor da variável que controlará a decisão e uma ação diferente será executada para cada valor que a variável poderá assumir.

```
escolha (<variavel>
  caso <valor1>
    <comandos>
  caso <valor2>, <valor3>, <valor
    <comandos>
  caso <valor5>
    <comandos>
  outrocaso
    <comandos>
fimescolha
```



# Escolha Caso

## Qual o horário das aulas?

Dia	Dia da Semana	Disciplina
1	Domingo	Não tem aula
2	Segunda-feira	LPA
3	Terça-feira	Lógica Matemática
4	Quarta-feira	IHC
5	Quinta-feira	IPE
6	Sexta-feira	PDM
7	Sábado	autoestudo

```
1. Algoritmo "Dias de Aula"
2. Var
3.     dia_da_semana : inteiro
4. Inicio
5.     escreva("=====")
6.     escreva("      DIA DA SEMANA      ")
7.     escreva(" [1] Domingo              ")
8.     escreva(" [2] Segunda-feira              ")
9.     escreva(" [3] Terça-feira                ")
10.    escreva(" [4] Quarta-feira              ")
11.    escreva(" [5] Quinta-feira              ")
12.    escreva(" [6] Sexta-feira              ")
13.    escreva(" [7] Sábado                    ")
14.    escreva("=====")
15.    leia(dia_da_semana)
16.    escolha(dia_da_semana)
17.        caso 1
18.            escreva("Domingo não tem aula")
19.        caso 2
20.            escreva("Aulas da Segunda-feira")
21.            escreva("Lógica de Programação e Algoritmos")
22.        caso 3
23.            escreva("Aulas da Terça-feira")
24.            escreva("Lógica Matemática")
25.        caso 4
26.            escreva("Aulas da Quarta-feira")
27.            escreva("Interação Humano-Computador")
28.        caso 5
29.            escreva("Aulas da Quinta-feira")
30.            escreva("Introdução à Programação Estruturada")
31.        caso 6
32.            escreva("Aulas da Sexta-feira")
33.            escreva("Progr. Dispositivos Móveis")
34.        caso 7
35.            escreva("Aulas do Sábado")
36.            escreva("autoestudo")
37.        outrocaso
38.            escreva("Dia inválido!")
39.    fimescolha
40. Fimalgoritmo
```

# Escolha Caso

- Desenvolva um algoritmo que receba dois valores numéricos inteiros e o símbolo da operação conforme tabela abaixo, calcule e mostre a operação efetuada.

Operador	Menu
+	Adição
-	Subtração
*	Multiplicação
/	Divisão

Algoritmo "Operacoes Inteiros"

Var

x, y : inteiro  
operacao : caractere  
resultado : real

Inicio

escreva("Informe o primeiro número: ")  
leia(x)  
escreva("Informe o segundo número: ")  
leia(y)  
escreva("Informe a operação que será efetuada: ")  
leia(operacao)

escolha(operacao)

caso "+"

resultado <- x + y  
escreva("x + y = ", resultado)

caso "-"

resultado <- x - y  
escreva("x - y = ", resultado)

caso "\*"

resultado <- x \* y  
escreva("x \* y = ", resultado)

caso "/"

se (x>y)  
resultado <- x / y  
senao  
resultado <- y - x  
fimse  
escreva("x / y = ", resultado)

outrocaso

escreva("Operador Inválido")

fimescolha

Fimalgoritmo



# Escolha Caso

- Desenvolva um algoritmo para calcular a área das 4 figuras geométricas da tabela.

Opção	Menu
Q	Quadrado
T	Triângulo
R	Retângulo
Z	Trapézio

Algoritmo "Area das Figuras"

Var

b, h, bm : inteiro  
figura: caractere  
area : real

Inicio

escreval ("Informe a base : ")  
leia (b)  
escreval ("Informe a altura: ")  
leia (h)  
escreval ("Informe a letra da figura que deseja calcular a área")  
escreval ("[Q] Quadrado")  
escreval ("[T] Triângulo")  
escreval ("[R] Retângulo")  
escreval ("[Z] Trapézio")  
leia (figura)

escolha (figura)

caso "Q"

area <- b \* h

escreva ("Area do quadrado = b \* h = ", area)

caso "T"

area <- (b \* h) / 2

escreva ("Area do triangulo = (b \* h)/2 = ", area)

caso "R"

area <- b \* h

escreva ("Area do retangulo = b \* h = ", area)

caso "Z"

area <- ((b + bm) \* h) / 2

escreva ("Area do trapézio = (b + bm) \* h / 2 = ", area)

outrocaso

escreva ("Figura Inválida")

fimescolha

Fimalgoritmo

# Escolha Múltipla

- A escolha múltipla permite escolher um algoritmo a partir de várias entradas. Neste exemplo, a seleção de vários estados converge para o mesmo algoritmo.

Região	Estados
Sul	RS, SC, PR
Sudeste	SP, RJ, MG, ES
Norte	AM, RR, AP, PA, TO, RO, AC
Nordeste	MA, PI, CE, RN, PE, PB, SE, AL, BA

Algoritmo "Regioes do Brasil"

Var

uf : caractere

Inicio

//entrada

escreva ("Digite a UF: ")

leia (uf)

//processamento

escolha (uf)

caso "RS", "SC", "PR"

escreva ("Região Sul")

caso "SP", "RJ", "MG", "ES"

escreva ("Região Sudeste")

caso "MT", "MS", "GO"

escreva ("Região Centro-Oeste")

caso "AM", "RR", "AP", "PA", "TO", "RO", "AC"

escreva ("Região Norte")

outrocaso

escreva ("Norte ou Nordeste")

fimescolha

Fimalgoritmo

# Escolha Caso Encadeado

- Fornecidos os critérios de risco de acordo com o peso e a idade, escreva um algoritmo que defina qual o grupo de risco de acordo com a tabela.

Idade	Peso		
	Até 60 kg	Entre 61 kg e 90 kg	Acima de 60 kg
0 a 17 anos	Grupo A	Grupo B	Grupo C
18 a 59 anos	Grupo D	Grupo E	Grupo F
Acima de 60 anos	Grupo G	Grupo H	Grupo I

```
1.  Algoritmo "Grupo de Risco"
2.  Var
3.      grupo : caractere
4.      idade, peso : inteiro
5.  Inicio
6.      //entrada
7.      escreva("Idade.....: ")
8.      leia(idade)
9.
10.     escreva("Peso (parte inteira).....: ")
11.     leia(peso)
12.
13.     //processamento e saída
14.     escolha (idade)
15.         caso 0 .. 17
16.             escolha (peso)
17.                 caso 0 .. 60
18.                     escreva("Grupo A")
19.                 caso 61 .. 90
20.                     escreva("Grupo B")
21.             outrocaso
22.                 escreva("Grupo C")
23.         fimescolha
24.     caso 18 .. 59
25.         escolha (peso)
26.             caso 0 .. 60
27.                 escreva("Grupo D")
28.             caso 61 .. 90
29.                 escreva("Grupo E")
30.             outrocaso
31.                 escreva("Grupo F")
32.         fimescolha
33.     outrocaso
34.         escolha (peso)
35.             caso 0 .. 60
36.                 escreva("Grupo G")
37.             caso 61 .. 90
38.                 escreva("Grupo H")
39.             outrocaso
40.                 escreva("Grupo I")
41.         fimescolha
42.     fimescolha
43.  Fimalgoritmo
```

# Interatividade

Considerando que as entradas para o algoritmo são as iniciais da região e do vendedor, analise as afirmativas e escolha a assertiva correta.

- I. Caso a região seja a Oeste e, no caso do vendedor ser a Lisa, o sistema vai escrever a mensagem ***Vendas: R\$ 9.500,00.***
- II. Caso a região seja a Norte e, no caso do vendedor ser o Ronaldo, o sistema vai escrever a frase “***Este vendedor não atua nesta região***”.
- III. No caso do usuário digitar “S” para a Região Sul, o sistema escreverá a mensagem ***Região sem atuação por esta empresa.***
- IV. No caso do usuário digitar N para região Norte e o vendedor ser o Jaime, o sistema escreverá a mensagem ***Vendas: R\$ 5.000,00.***

# Interatividade

São verdadeiras as afirmativas:

- a) I, II e III, apenas.
- b) I e III, apenas
- c) I e II, apenas.
- d) II e IV, apenas.
- e) III e IV, apenas.

```
Algoritmo "Vendas Regiao"
Var
    regiao, vendedor : caractere
Inicio
    escreva("As regioes sao: Leste, Oeste e Norte")
    escreva("Informe a primeira letra da regiao: ")
    leia(regiao)

    escolha(regiao)
    caso "I"
        escreva("Os vendedores sao: Rafael, Joao e Mariana")
        escreva("Informe a primeira letra do nome do vendedor: ")
        leia(vendedor)
        escolha(vendedor)
        caso "R"
            escreva("Vendas: R$ 10.000,00")
        caso "J"
            escreva("Vendas: R$ 12.000,00")
        caso "M"
            escreva("Vendas: R$ 14.000,00")
        outrocaso
            escreva("Este vendedor não atua nesta região")
    fimescolha
caso "O"
    escreva("Os vendedores sao: Ronaldo, Lisa e Hilton")
    escreva("Informe a primeira letra do nome do vendedor: ")
    leia(vendedor)
    escolha(vendedor)
    caso "R"
        escreva("Vendas: R$ 10.000,00")
    caso "L"
        escreva("Vendas: R$ 9.500,00")
    caso "H"
        escreva("Vendas: R$ 13.000,00")
    outrocaso
        escreva("Este vendedor não atua nesta região")
    fimescolha
caso "N"
    escreva("Os vendedores sao: Thomas, Jaime e Raquel")
    escreva("Informe a primeira letra do nome do vendedor: ")
    leia(vendedor)
    escolha(vendedor)
    caso "T"
        escreva("Vendas: R$ 5.000,00")
    caso "J"
        escreva("Vendas: R$ 9.000,00")
    caso "R"
        escreva("Vendas: R$ 14.000,00")
    outrocaso
        escreva("Este vendedor não atua nesta região")
    fimescolha
outrocaso
    escreva("Região sem atuação desta empresa")
fimescolha
Fimalgoritmo
```

# Resposta

São verdadeiras as afirmativas:

- a) I, II e III, apenas.
- b) I e III, apenas
- c) I e II, apenas.
- d) II e IV, apenas.
- e) III e IV, apenas.

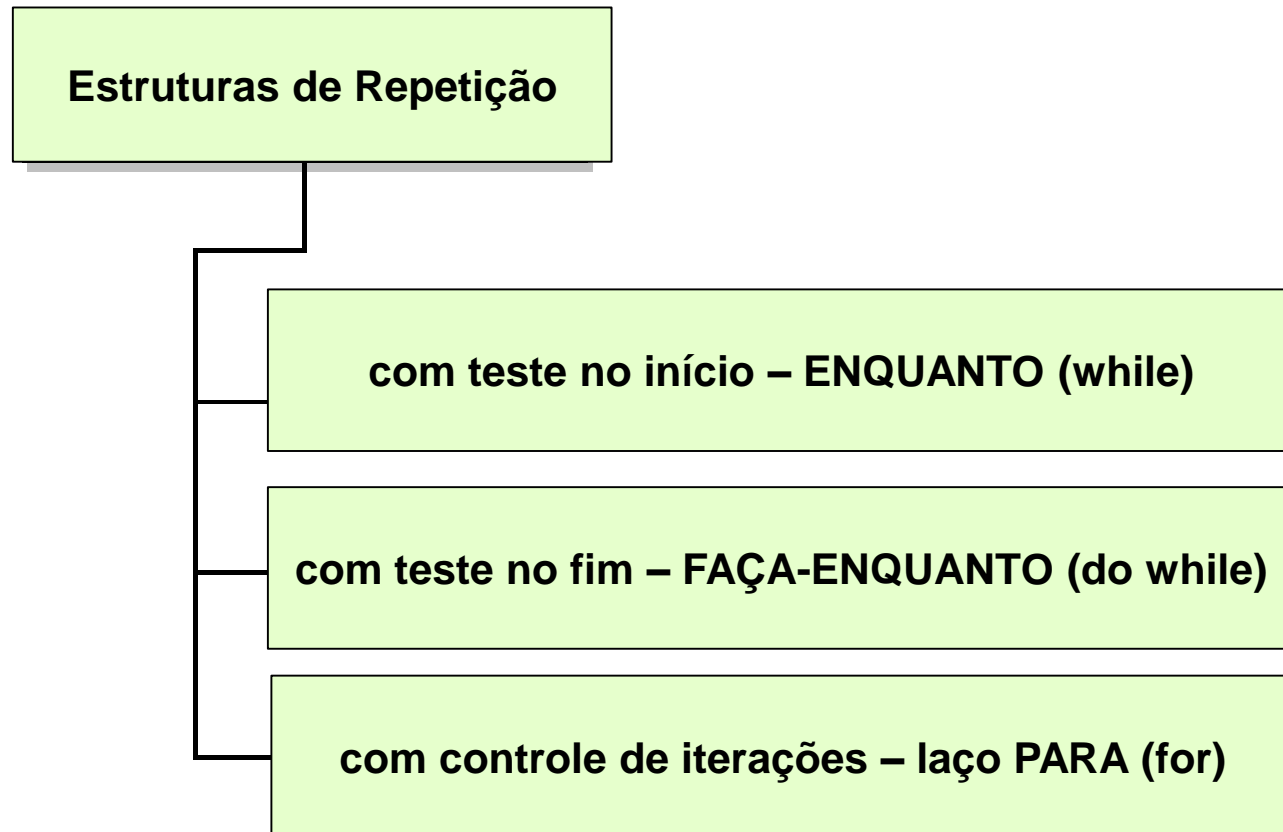
```
Algoritmo "Vendas Regiao"
Var
    regiao, vendedor : caractere
Inicio
    escreva("As regioes sao: Leste, Oeste e Norte")
    escreva("Informe a primeira letra da regiao: ")
    leia(regiao)

    escolha(regiao)
    caso "I"
        escreva("Os vendedores sao: Rafael, Joao e Mariana")
        escreva("Informe a primeira letra do nome do vendedor: ")
        leia(vendedor)
        escolha(vendedor)
        caso "R"
            escreva("Vendas: R$ 10.000,00")
        caso "J"
            escreva("Vendas: R$ 12.000,00")
        caso "M"
            escreva("Vendas: R$ 14.000,00")
        outrocaso
            escreva("Este vendedor não atua nesta região")
    fimescolha
caso "O"
    escreva("Os vendedores sao: Ronaldo, Lisa e Hilton")
    escreva("Informe a primeira letra do nome do vendedor: ")
    leia(vendedor)
    escolha(vendedor)
    caso "R"
        escreva("Vendas: R$ 10.000,00")
    caso "L"
        escreva("Vendas: R$ 9.500,00")
    caso "H"
        escreva("Vendas: R$ 13.000,00")
    outrocaso
        escreva("Este vendedor não atua nesta região")
    fimescolha
caso "N"
    escreva("Os vendedores sao: Thomas, Jaime e Raquel")
    escreva("Informe a primeira letra do nome do vendedor: ")
    leia(vendedor)
    escolha(vendedor)
    caso "T"
        escreva("Vendas: R$ 5.000,00")
    caso "J"
        escreva("Vendas: R$ 9.000,00")
    caso "R"
        escreva("Vendas: R$ 14.000,00")
    outrocaso
        escreva("Este vendedor não atua nesta região")
    fimescolha
outrocaso
    escreva("Região sem atuação desta empresa")
fimescolha
Fimalgoritmo
```

# Estruturas de Repetição

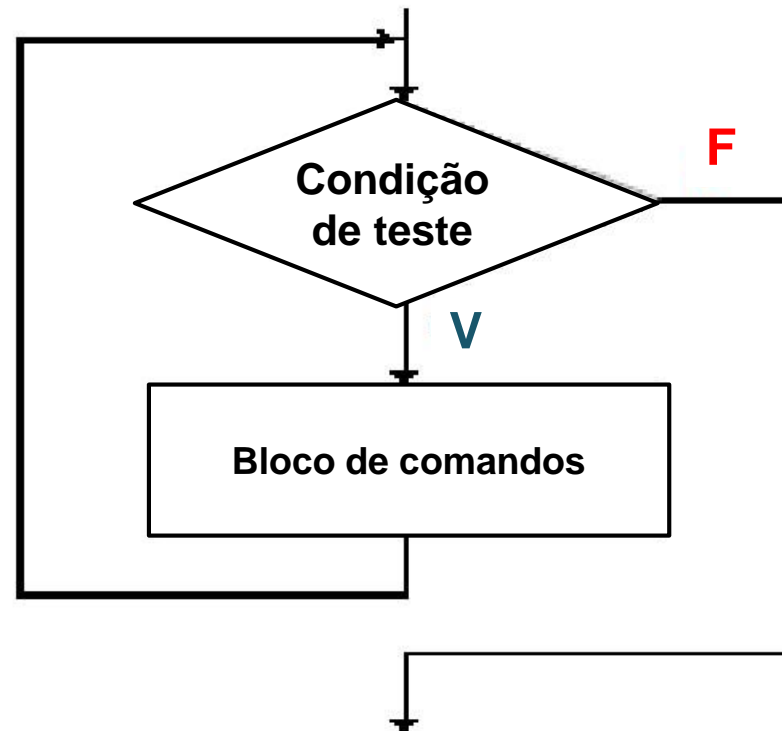
- Evita a repetição de códigos.
- Torna a lógica mais coesa.

Há três estruturas de repetição:



# Estrutura de Repetição com teste no início

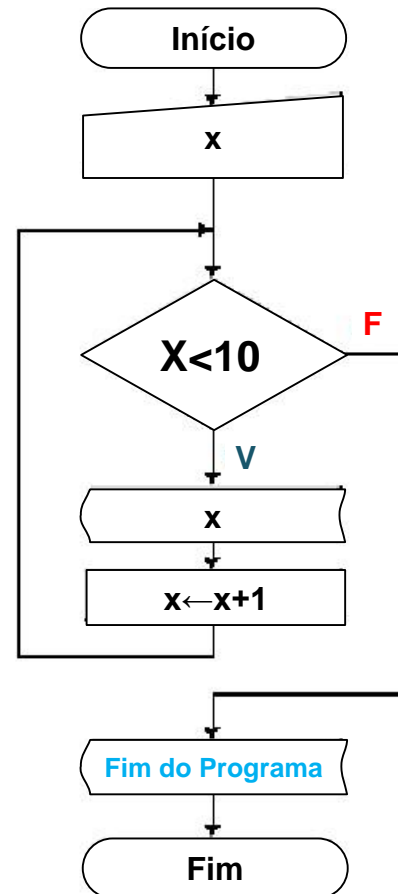
- A estrutura de repetição com teste no início estabelece o começo do laço com uma condição de teste.
- Se o teste resultar verdadeiro, então o bloco de comandos dentro do laço é executado e, ao término, a condição é novamente testada.
- O bloco interno será repetido enquanto a condição for verdadeira.





# Laço Enquanto – estrutura de repetição com teste no início

- Na estrutura de repetição com teste no início do algoritmo abaixo, a condição é satisfeita enquanto x for menor que 10. O valor de x é incrementado dentro do laço.
- O valor de entrada na linha 6 for 10, o bloco dentro do laço não será executado.
- O valor de entrada na linha 6 for 8, o bloco dentro do laço será executado duas vezes.

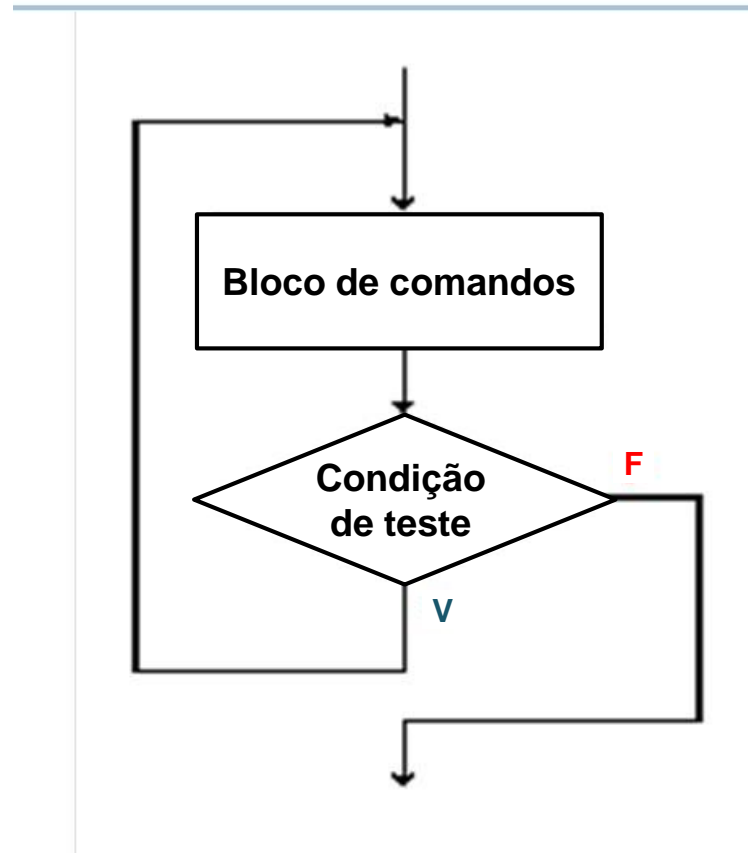


```
1.  Algoritmo "Repetição com teste no início"
2.  Var
3.      x : inteiro
4.  Início
5.      escreva("Digite um número inteiro: ")
6.      leia(x)
7.      enquanto (x < 10) faça
8.          escreval(x)
9.          x ← x + 1
10.     fimenquanto
11.     escreva("Fim do Programa")
12. Fimalgoritmo
```

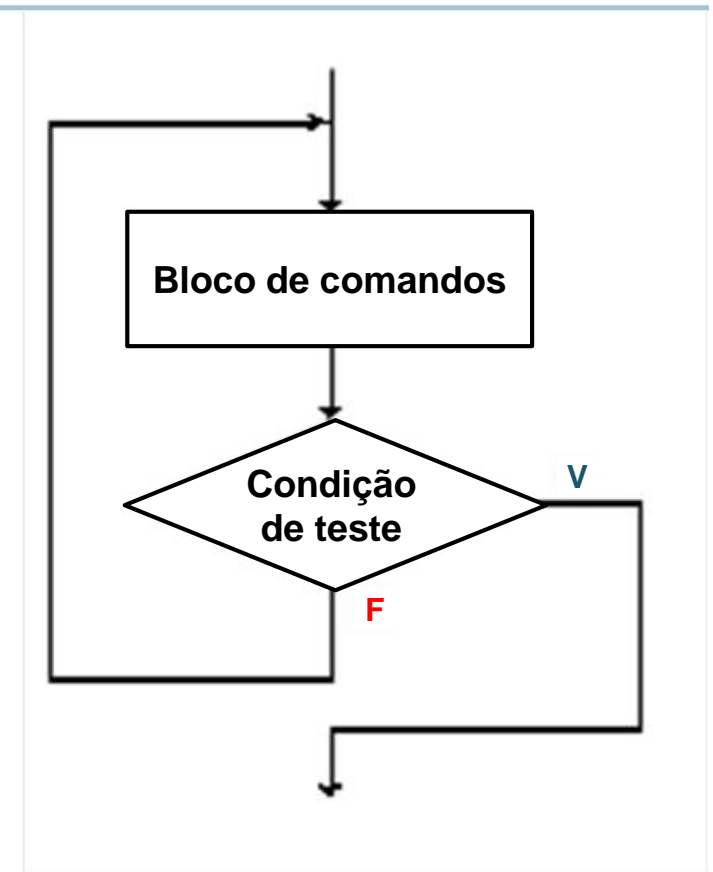
# Estrutura de Repetição com teste no Fim

- Diferente da estrutura de repetição com o teste no início, em que os processos são executados apenas quando a condição é satisfeita, na estrutura de repetição com teste no final o processo é executado pelo menos uma vez, independentemente de a condição ser ou não satisfeita.

FAÇA... ENQUANTO



REPITA... ATÉ



# Estrutura de Repetição com teste no Fim

```
1. Algoritmo "MenorMaior"
2. var
3.   n, numero, maior, menor : inteiro
4. Inicio
5.   //entrada do primeiro valor
6.   escreva("Digite um numero: ")
7.   leia(numero)
8.
9.   //inicialização das variáveis
10.  maior ← numero
11.  menor ← numero
12.  n ← -9
13.  //processamento
14.  repita
15.    escreva("Digite um numero: ")
16.    leia(numero)
17.    se (numero > maior) então
18.      maior ← numero
19.    fimse
20.    se (numero < menor) então
21.      menor ← numero
22.    fimse
23.    n ← n-1
24.  até n=0
25.  //saída
26.  escreva("O maior número é ", maior)
27.  escreva("O menor número é ", menor)
28. Fimalgoritmo
```

A)

```
1. Algoritmo "MenorMaior2"
2. var
3.   n, numero, maior, menor : inteiro
4. Inicio
5.   //entrada do primeiro valor
6.   escreva("Digite um numero: ")
7.   leia(numero)
8.
9.   //inicialização das variáveis
10.  maior ← numero
11.  menor ← numero
12.  n ← -9
13.  //processamento
14.  faça
15.    escreva("Digite um numero: ")
16.    leia(numero)
17.    se (numero > maior) então
18.      maior ← numero
19.    fimse
20.    se (numero < menor) então
21.      menor ← numero
22.    fimse
23.    n ← n-1
24.  enquanto (n > 0)
25.  //saída
26.  escreva("O maior número é ", maior)
27.  escreva("O menor número é ", menor)
28. Fimalgoritmo
```

B)

# Estrutura de Repetição – Laço Para

```
1.  Algoritmo "Tabuada de N usando o laço para"
2.  Var
3.      x, n : inteiro
4.  Inicio
5.      escreva("Digite um Número: ")
6.      leia(n)
7.
8.      para x de 0 ate 10 passo 1 faça
9.          escreval(n, " * ", x, " = ", n*x)
10.     fimpara
11. Fimalgoritmo
```

n	x	n*x
3	x 0	= 0
3	x 1	= 3
3	x 2	= 6
3	x 3	= 9
3	x 4	= 12
3	x 5	= 15
3	x 6	= 18
3	x 7	= 21
3	x 8	= 24
3	x 9	= 27
3	x 10	= 30

# Estrutura de Repetição – Laço Para

- Elabore um algoritmo que efetue a soma de números ímpares entre um dado intervalo informado pelo usuário.

```
1.  Algoritmo "Soma entre números ímpares"
2.  Var
3.      x, y, menor, maior, soma : inteiro
4.
5.  Inicio
6.      //entrada
7.      escreva("Digite um valor....: ")
8.      leia(x)
9.      escreva("Digite outro valor.: ")
10.     leia(y)
11.
12.     //processamento
13.     soma <- 0
14.
15.     se ( x > y ) entao
16.         maior <- x
17.         menor <- y
18.     senão
19.         maior <- y
20.         menor <- x
21.     fimse
22.
23.     enquanto (menor <= maior) faca
24.         se (menor mod 2 <> 0) entao
25.             escreva(" + ", menor, " ")
26.             soma <- soma+menor
27.         fimse
28.         menor <- menor+1
29.     fimenquanto
30.
31.     //saida
32.     escreva(" ")
33.     se (x<y) entao
34.         escreva("Somatório dos ímpares entre ", x, " e ", y, " é: ")
35.     senão
36.         escreva("Somatório dos ímpares entre ", y, " e ", x, " é: ")
37.     fimse
38. Fimalgoritmo
39.
40.
```

# Interatividade

Dado o algoritmo ao lado, analise as afirmativas corretas:

- I. O laço de repetição será executado 10 vezes.
  - II. A seguinte sequência de números será exibida na tela.  
0;1;2;3;4;5;6;7;8;9.
  - III. O laço de repetição será executado 0 vezes.
  - IV. A seguinte sequência de números será exibida na tela:  
10;9;8;7;6;5;4;3;2;1;0.
- a) Estão corretas I e II.
  - b) Estão corretas I e IV.
  - c) Apenas a III está correta.
  - d) Apenas a IV está correta.
  - e) Nenhuma das alternativas anteriores.

**Algoritmo** “Exibe Numero”

**Var**

num : **inteiro**

**Inicio**

num  $\leftarrow$  0

**enquanto** (num > 10) faça

    escreva(num)

    num  $\leftarrow$  num + 1

**fimenquanto**

**Fimalgoritmo**

# Resposta

Dado o algoritmo ao lado, analise as afirmativas corretas:

- I. O laço de repetição será executado 10 vezes.
- II. A seguinte sequência de números será exibida na tela.  
0;1;2;3;4;5;6;7;8;9.
- III. O laço de repetição será executado 0 vezes.
- IV. A seguinte sequência de números será exibida na tela:  
10;9;8;7;6;5;4;3;2;1;0.

- a) Estão corretas I e II.
- b) Estão corretas I e IV.

- c) Apenas a III está correta.
- d) Apenas a IV está correta.
- e) Nenhuma das alternativas anteriores.

**Algoritmo** “Exibe Numero”

**Var**

num : **inteiro**

**Inicio**

num  $\leftarrow$  0

**enquanto** (num > 10) faça

    escreva(num)

    num  $\leftarrow$  num + 1

**fimenquanto**

**Fimalgoritmo**

**ATÉ A PRÓXIMA!**