PDA - Repetição

Estruturas de repetição

Slides adaptação de Deise Saccol

- Existem situações em que o processamento de um valor uma única vez não é suficiente para resolver o problema.

- Quando isto ocorre, deve-se utilizar uma estrutura de repetição.

- Estruturas de repetição permitem que uma ou mais instruções sejam executadas um número definido de vezes, ou enquanto uma determinada condição não é alcançada.

- As estruturas de repetição também são conhecidas como **Laços** ou **Loops**.
- As estruturas de repetição das linguagens de programação são de dois tipos:
  - Condicional
    - Repetem até satisfazer a condição de repetição.
    - São usadas quando não se sabe previamente quantas vezes deve-se executar as instruções do bloco de repetição.
  - Contada
    - Repetem um número contado (pré-definido) de vezes.
    - São usadas quando se sabe previamente quantas vezes deve-se executar as instruções do bloco de repetição.

- ESTRUTURAS DE REPETIÇÃO
  - Se uma ação se repete em um algoritmo, em vez de escrevê-la várias vezes, em certos casos podemos resumir anotando uma vez só e solicitando que ela se repita, usando uma das **estruturas de repetição**.
  - Podemos pedir que uma ação (ou um conjunto de ações) seja executada um número *definido* ou *indefinido* de vezes, ou *enquanto* um estado permanecer ou *até que* um estado seja atingido.
  - As principais estruturas de repetição são:
    - ENQUANTO...FAÇA // Condicionais
    - REPETA...ATE // Condicionais
    - PARA...ATÉ...FAÇA // Contada

- Normalmente, a utilização de uma estrutura de repetição requer o uso de dois tipos de variáveis:
  - Contadora: é inicializada antes da estrutura de repetição e é incrementada no interior desta por um valor constante.
  - Acumuladora: é inicializada antes da estrutura de repetição e é incrementada no interior desta por um valor variável.

- Exemplos:

Algoritmo variávelContadora;
...
cont = 0
<estrutura de repetição>
...
cont = cont + 1
... **constante**
<fim estrutura de repetição>
...

Algoritmo variávelAcumuladora;
...
soma = 0
<estrutura de repetição>
...
soma = soma + X
Variável
...
<fim estrutura de repetição>
...

– Com este tipo de instrução podemos fazer um *contador*. Veja como seria uma contagem até 10:

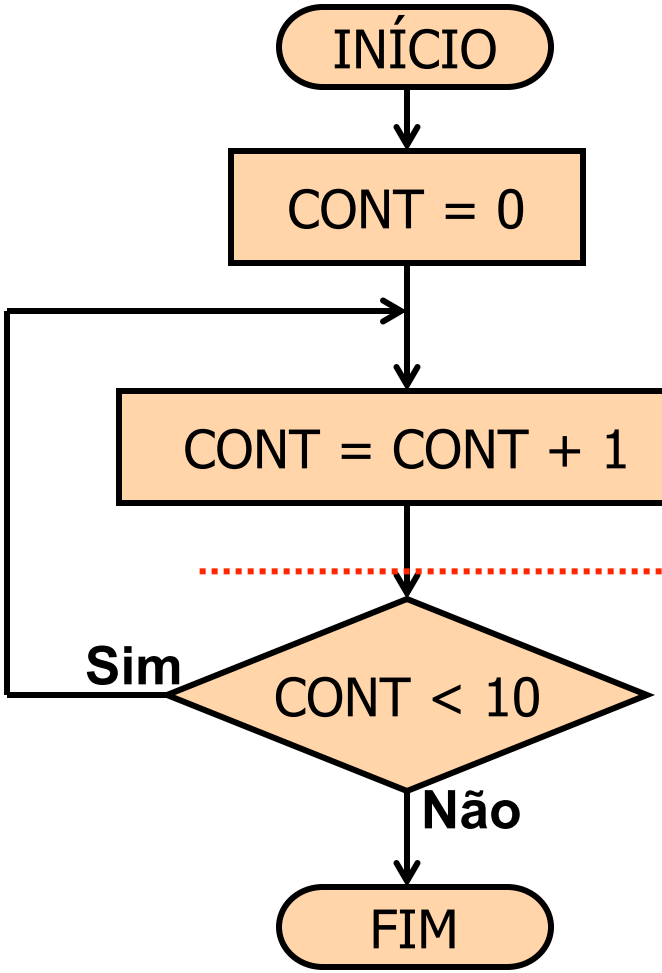
- CONTADOR = 0
- **Repetir**

  CONTADOR = CONTADOR + 1

  ate CONTADOR < 10

Isto será repetido 10 vezes.

Fluxograma do contador:



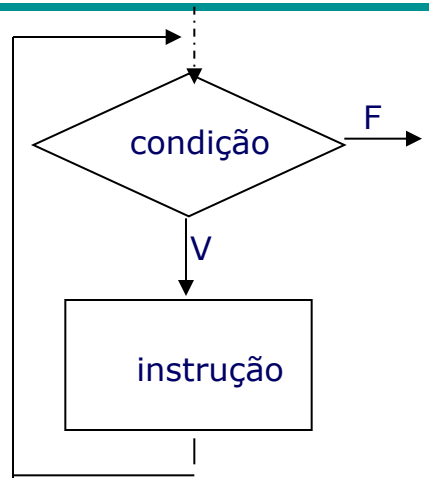
Neste ponto do algoritmo podemos incluir qualquer conjunto de instruções que quisermos repetir 10 vezes.

Estruturas de repetição

Enquanto...Faça

Enquanto (condição) faca

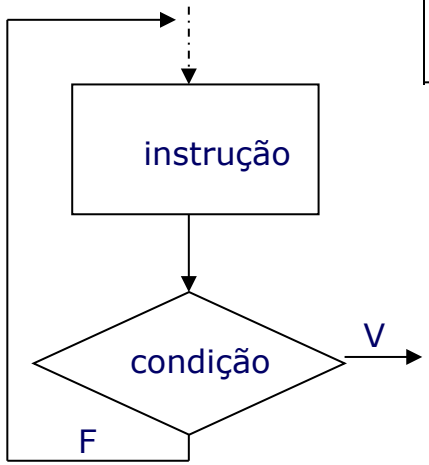
<instruções>



Repetir...Ate

Repetir <instruções>

ate (condição)



Para...ate...Faca

Para <variavel> = <inicio> até <fim> faca

<instruções>

Exemplo Enquanto

- **Ler 50 números fornecidos pelo usuário e calcular e exibir a média.**

```
Soma = 0;
Cont = 0;
Enquanto cont < 50 faca
  {
    ler num;
    soma = soma + num;
    cont = cont + 1;
   }
Media = soma / cont
Mostrar media
```

Exemplo Repita

Ler 50 números fornecidos pelo usuário e calcular e exibir a média.

```
Soma = 0;
Cont = 0;
Repita
  ler num;
  soma = soma + num;
  cont = cont + 1;
ate cont  <=50;
Media = soma / cont;
Mostrar media;
```

Exemplo Para

- **Ler 50 números fornecidos pelo usuário e calcular e exibir a média.**

```
Soma = 0;
Para cont =0 até 50 faca {
  ler num;
  soma = soma + num;
}
Media = soma / cont;
Mostrar media;
```

- Até agora para vários valores informados pelo usuário líamos cada valor de forma separada
- Por exemplo, no algoritmo para o cálculo da média de quatro números, líamos 4 vezes, 4 valores para dentro de 4 variáveis.

Mas também poderíamos:

- ler um valor para 1 variável e repetir isso 4 vezes, adicionando cada valor lido ao total em uma outra variável, a cada repetição.
- Após as 4 repetições, a soma dos 4 números estaria acumulada na outra variável, bastando uma instrução para dividi-la por 4 e assim obter a média.

Média de notas de alunos em uma turma

```
soma=0;

ler n1;
ler n2;
ler n3;
soma = n1+n2+n3;
media = soma/3;
exibir media;
```

```
soma=0;
i=0;
repita
    ler n;
    se (n>=0) entao soma = soma + n;
        i = i+1;
ate i<=3;
media = soma/i;
exibir media;
```

PARA...ATÉ...FACA

- Formato:

  Para <variável> **=** <valor inicial> **até** <valor final>
  faca <ações>

  - Significado: A <variável> é inicializada com <valor inicial>. Após cada execução das <ações>, soma-se 1 à <variável> e repete-se as <ações>, continuando assim até que a <variável> atinja o <valor final>.
  - Esta estrutura de repetição cria um *contador automático*, que nós não precisamos mandar incrementar.
  - Ao usar esta estrutura já está subentendido que a <variável> inicia com <valor inicial> e é incrementada a cada ciclo (podendo-se inclusive aproveitar seu valor dentro do ciclo), e que as <ações> serão repetidas até que a <variável> tenha o <valor final>.

- Exemplo da estrutura PARA...ATÉ...FAÇA:
  "Mostrar os quadrados dos inteiros de 3 a 11."
  - Pseudocódigo:

```
Para CONT = 3 até 11 faca
    Mostrar (CONT ** 2)
```

Isto será repetido 9 vezes.

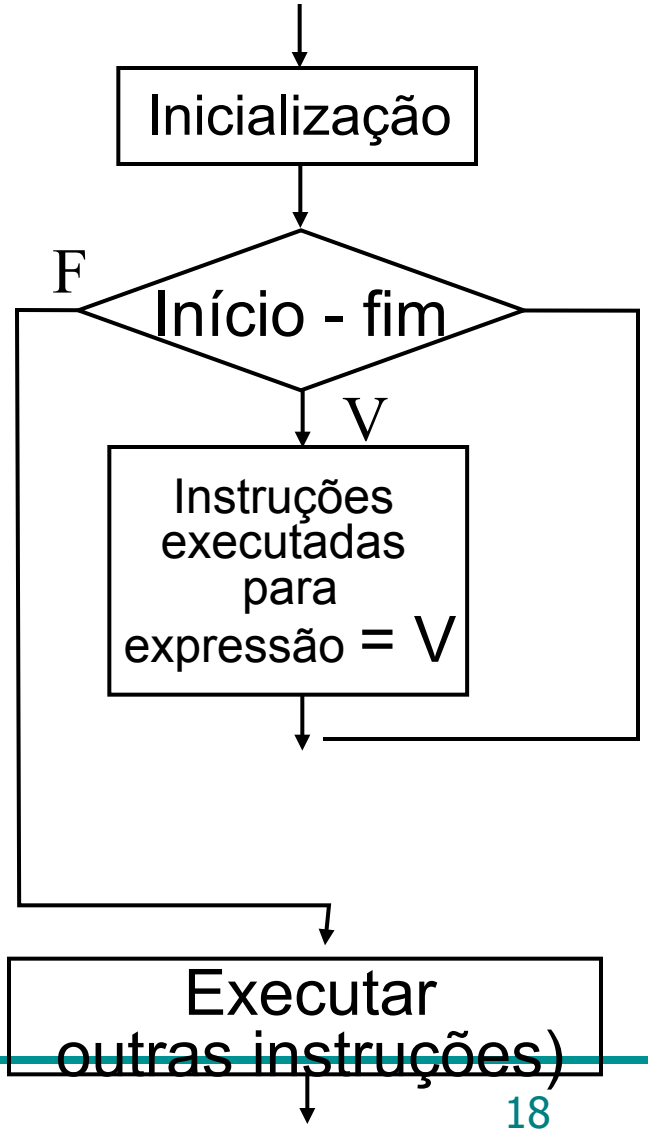
  - Usamos esta estrutura quando sabemos quantas vezes temos de repetir certas ações, mesmo que o número de vezes só seja conhecido *durante a execução*. Por exemplo:
    "Perguntar ao usuário de quantos valores ele quer calcular a média. Ler os números e calcular a média."

E se eu quisesse calcular a média de N números?

- Para esse problema construímos um algoritmo que será genérico, ou seja, que poderá ser usado para calcular a média de quantos números se quiser!

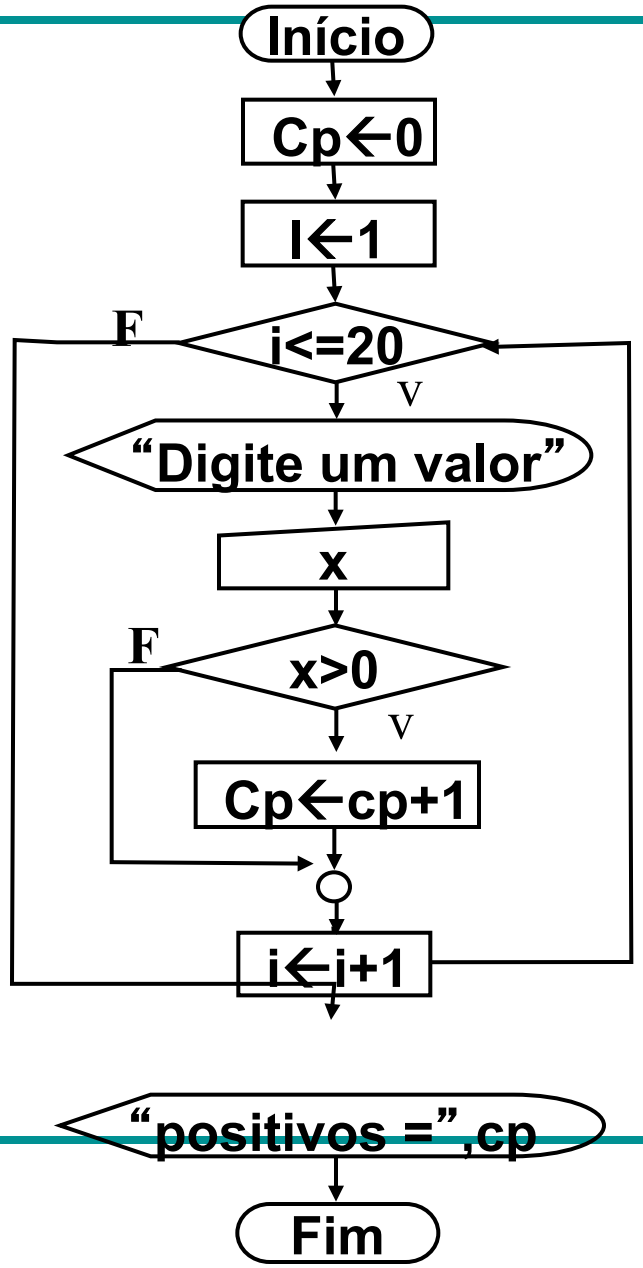
```
..
Exibir (“De quantos valores você quer calcular a média?”);
Ler QUANT ; //(aqui se descobre quantas repetições)
SOMA = 0;
Para CONT = 1 até QUANT faca
 {
    Ler N; // (aqui é lido cada número, um em cada ciclo)
    SOMA = SOMA + N;
 }
MEDIA = SOMA / QUANT;
Mostrar MEDIA;
```

- Estrutura de repetição Para/Faça
  - Semântica:
    - Repete as instruções enquanto a variável contador não atingir o valor final. Ressalta-se que a variável contador é previamente inicializada e incrementada ou decrementada de uma constante a cada repetição.



- Estrutura de repetição Para/Faça
- Resumindo...
  - Sabe-se de antemão quantas vezes o bloco de repetição será executado. Isto é, repete enquanto o valor final não atingir o valor final da variável de controle.
  - Incrementa automaticamente a variável de controle cada vez que o bloco é executado (incremento 1 é o padrão).
  - A variável de controle deve ser um número inteiro.
  - A variável de controle não pode ser modificada dentro do bloco.

```
Algoritmo ExemploParaFaça
Início
contP=0;
Para i=1 até 20 faça
      Escreva("Digite um valor");
      Leia (x);
      Se (x > 0) Então
            contP = contP + 1;
Fim Para;
Escreva ("positivos = "+(contP));
Fim.
```



```
Algoritmo ExemploParaFaçaComRepita
Início
contP=0; i=1;
Repita
    Escreva("Digite um valor");
    Leia (x);
    Se (x > 0) Então
        contP = contP + 1;
            i = i+1; //{Incremento explícito da variável
      contadora}
Até (i > 20);
Escreva ("positivos = "+contP);
Fim.
```

ENQUANTO ... FACA ...

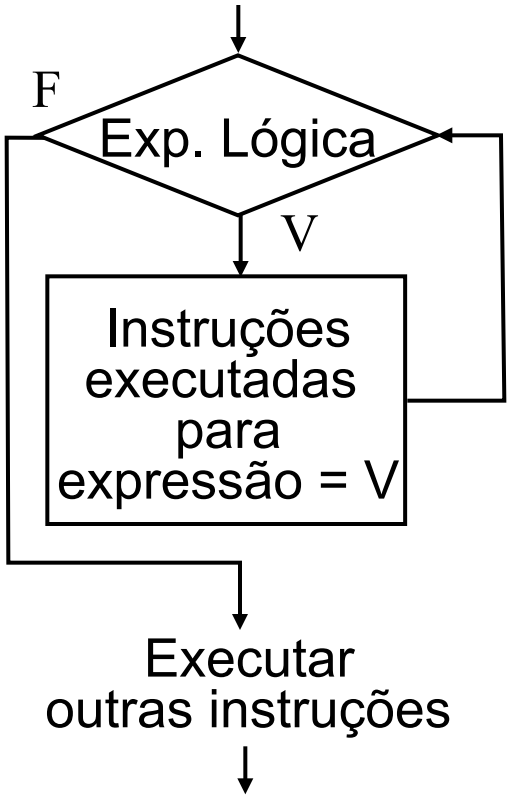
– Formato:

Enquanto <operação lógica> **faca** <ações>

- Significado: A <operação lógica> é testada. Se for verdadeira, então executar <ações> e em seguida testar novamente a op. lógica. Este ciclo prossegue até que em algum teste a op. lógica resulte em falso.
- Nesta estrutura temos novamente, assim como nas estruturas de decisão, uma *operação lógica* determinando se devemos *continuar* a repetir (resultado V) ou *parar* de repetir (resultado F) as ações.
- Devemos garantir que o *dado testado* na op. lógica tenha seu *valor modificado* por alguma das ações repetidas, senão nunca teremos um resultado F no teste e a repetição permanecerá num ciclo infinito (loop)!

ENQUANTO ... FACA ...

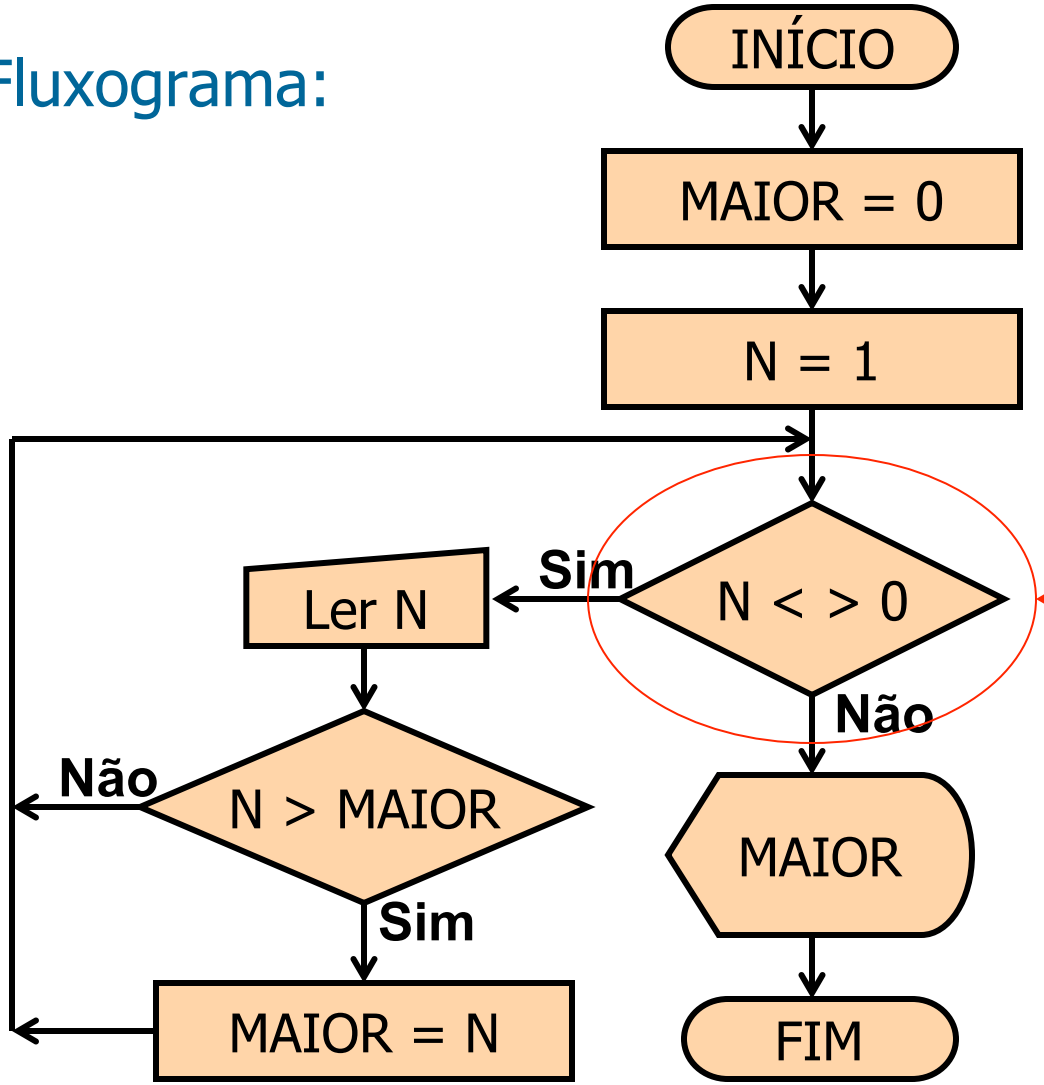
- Semântica:
  - Faz o teste no início do laço. Se o resultado for V as instruções do laço são executadas. Volta-se para o início do laço e testa-se novamente a sua condição. Isto é repetido enquanto a condição testada for V.



- Exemplo da estrutura ENQUANTO...FACA:

```
..
MAIOR = 0; // (o maior por enquanto é o menor valor)
N = 1; // (só para o 1.o teste funcionar...)
Enquanto (N <> 0) faca
{
    Ler N; // (aqui o valor de N muda, é a entrada do usuário)
    Se (N > MAIOR) então
    MAIOR = N;
}
Mostrar MAIOR; // (isto só executa qdo. o ciclo parar)
```

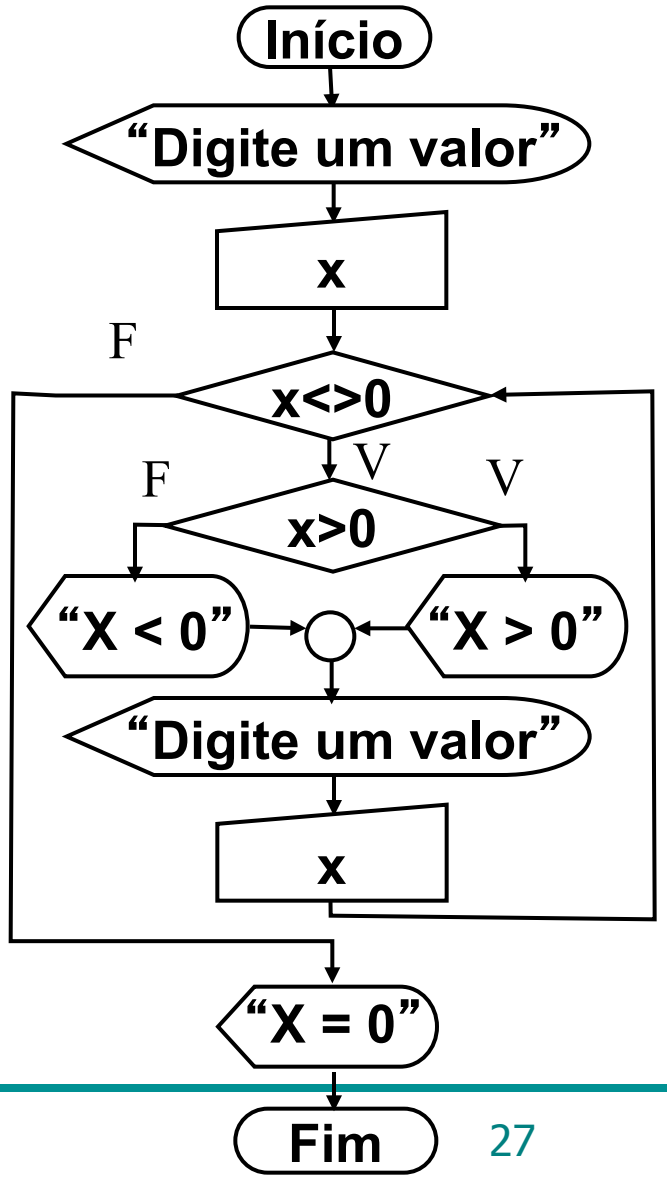
– Fluxograma:



Na estrutura ENQUANTO...FACA, veja que o teste é feito no *início* do ciclo.

- Estrutura de repetição Enquanto/Faça
- Resumindo...
  - Não se sabe de antemão quantas vezes o bloco de repetição será executado. Isto é, ele pode ser executado várias vezes ou nenhuma vez.
  - Testa a condição antes de entrar na estrutura de repetição
  - Repete a execução do bloco de instruções toda vez que condição for V.
  - O bloco de instruções é finalizado quando a condição for F.

```
Algoritmo ExemploEnquantoFaça
Início
Escrever (“Digite um valor”);
Ler (x);
Enquanto (x<>0) faça {
      Se (x > 0) Então
            Escrever (X “> 0”);
      Senão
            Escrever (X “< 0”);
      Escrever (“Digite um valor”);
      Ler (x);
}
Escrever (X “ = 0”);
Fim.
```



REPITA ... ATÉ

- Estrutura de repetição Repita/Até

- Sintaxe:

Repita

<Comando1>;

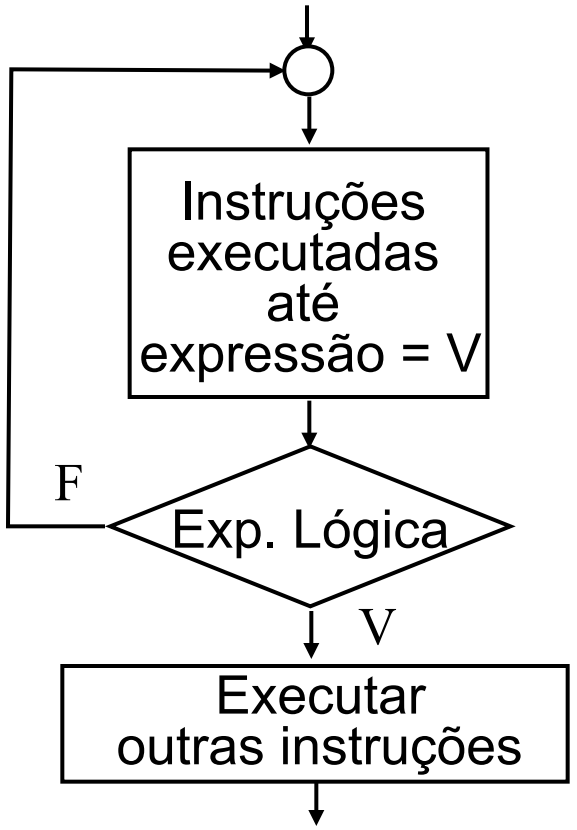
. . .

<ComandoN>;

Até <expressão lógica>;

REPITA ... ATÉ

- Semântica:
  - Efetua um teste lógico no fim do laço, garantindo que pelo menos uma vez as instruções deste são executadas. Ao contrário do enquanto, esta só repete o laço se o resultado do teste for F.

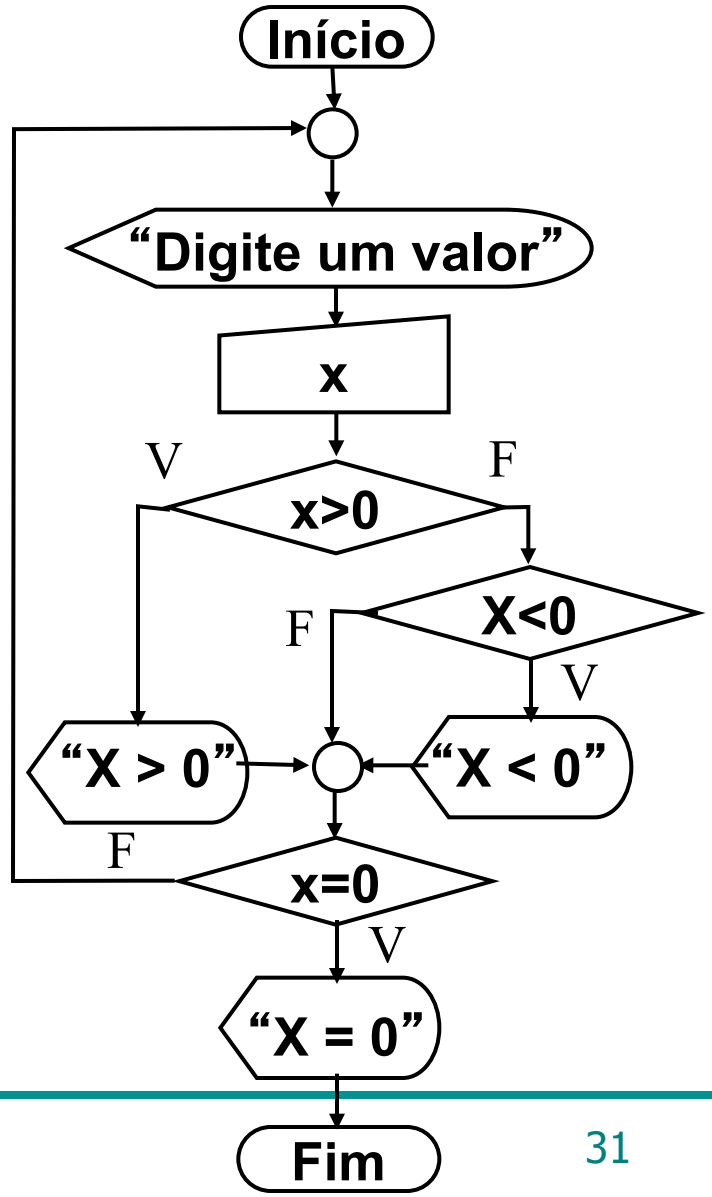


REPITA ... ATÉ

- Estrutura de repetição Repita/Até
- Resumindo...
  - Não se sabe de antemão quantas vezes o bloco de repetição será executado. Todavia é garantido que ele será executado pelo menos uma vez.
  - Testa a condição depois de entrar na estrutura de repetição.
  - Repete a execução do bloco de instruções toda vez que a condição for F.
  - A execução do bloco é finalizada quando a condição for V.

REPITA ... ATÉ

```
Algoritmo ExemploRepitaAté
Início
Repita
    Escrever (“Digite um valor”);
    Ler (x);
    Se (x > 0) Então
        Escrever (X “> 0”);
    Senão
        Se (x < 0) Então
            Escrever (X "< 0”);
Até (x = 0);
Escrever (X "= 0”);
Fim.
```



Nos laços condicionais a variável que é testada deve estar sempre associada a uma instrução que a atualize no interior do laço, pois, se isto não ocorrer, o algoritmo ficará repetindo indefinidamente este laço, gerando uma situação conhecida como “laço/loop” infinito”.

```
Algoritmo ExemploEnquantoFaça_LoopInfinito
Início
Escrever (“Digite um valor”);
Ler (x);
Enquanto (x<>0) faça {
    Se (x > 0) Então
        Escrever (X “> 0”);
    Senão
        Escrever (X “< 0”);
    Escrever (“Digite um valor”);
    //{Ler (x); - sem este Ler (x), o laço se repete infinitamente!}
}
Escrever (X "= 0”);
Fim.
```

- As Estruturas de Repetição são assim comparadas:

	Enquanto/Faça	**Repita/Até**	**Para/Faça**
Tipo da Estrutura	Condicional	Condicional	Contada
Ocorrência do Teste	Início	Fim	Início
Quantidade de Repetições	0 ou muitas	Mínimo 1	((fim-início) div passo) + 1
Condição para Repetir	V	F	Início <= Fim ou Início >= Fim

- Semelhante às estruturas de decisão composta, as estruturas de repetição também podem ser encadeadas/aninhadas.

- Esta abordagem é usada quando há a necessidade de se usar laços dentro de laços.
  - Por exemplo: fazer um algoritmo para gerar toda a tabuada de soma de 1 a 10.

```
Algoritmo TabuadaSoma
Início
n1=1;
Enquanto (n1<=10) faça {
    Para n2 = 1 até 10 faça {
        r = n1 + n2;
        Escrever ((n1),“ + ”,(n2),“ = ”,(r));
    }
     n1=n1+1;
}
Fim.
```

Também pode-se encadear estruturas de repetição com estruturas de decisão.

```
Algoritmo TabuadaSomaComParidade
Início
n1=1;
Enquanto (n1<=10) faça {
    Para n2 =1 até 10 faça {
        r = n1 + n2;
        Se (r mod 2 == 0)
                Escrever ((n1),“ + ”,(n2),“ = ”,(r), “ = Par”);
        Senão
                Escrever ((n1),“ + ”,(n2),“ = ”,(r), “ = Impar”);
    }
    n1=n1+1;
}
Fim.
```

Indentação

=

Organização hierárquica das estruturas e suas instruções.

Facilita visualizar o que está contido em que.
Auxilia no entendimento do código
e na busca de erros.

Exercícios

Leia uma lista de números inteiros positivos e escreva a média aritmética de todos os números lidos que são pares. O algoritmo finaliza quando é digitado zero, o qual deve ser excluído para cálculo da média.

Leia um código de votação e escreva a ordem de classificação e o percentual de votos de cada candidato. Considere: a) F = fim da eleição; b) X,Y,Z = códigos dos candidatos; c) N = voto nulo e d) B = voto em branco.