

Estruturas de Repetição 1

MATA37: Introdução à Lógica de Programação

Prof.: Rafael A. Melo (melo@dcc.ufba.br)

Departamento de Ciência da Computação

Instituto de Matemática

Universidade Federal da Bahia



Linguagens de Alto Nível

- Linguagem com um nível de abstração relativamente elevado, longe do código de máquina e mais próximo à linguagem humana.
- Ex.: Pascal, C, C++, C#, Delphi, Visual Basic, Java, etc.

Estrutura de Repetição

- Estrutura de controle que permite repetir diversas vezes um mesmo trecho do algoritmo.
- Os trechos repetidos são chamados de
 - Laços
 - *Loops*

Estruturas de Repetição

- Repetições condicionais
 - a repetição ocorre condicionada a uma condição lógica.
- Repetições incondicionais
 - que têm um número pré-definido de repetições.

Repetição com Teste no Início

- Permitir repetir um mesmo trecho do algoritmo, mas sempre verificando **antes** de cada execução se é “permitido” executar o trecho.
- Trecho do algoritmo = Bloco de comandos

Repetição com Teste no Início

*Enquanto houver itens na lista de compras
Comprar próximo e item e excluí-lo da lista*

- ~~Arroz~~



- ~~Feijão~~



- ~~Banana~~



- ~~Maçã~~



- ~~Sabão em pó~~



- ~~Biscoito~~



Repetição com Teste no Início

enquanto <condição> **faça**

comando1

comando2

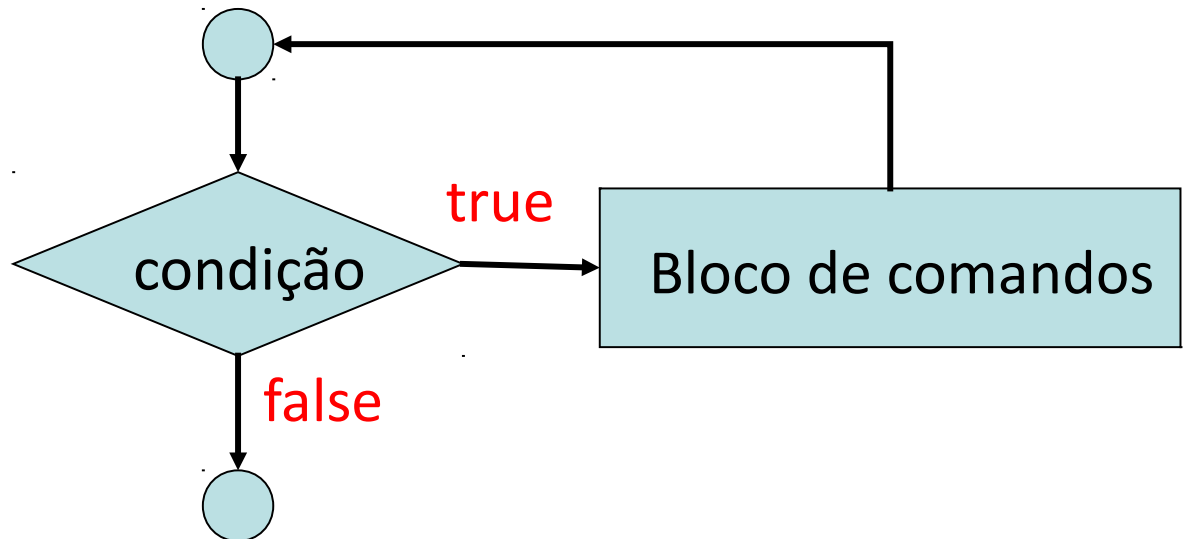
.

.

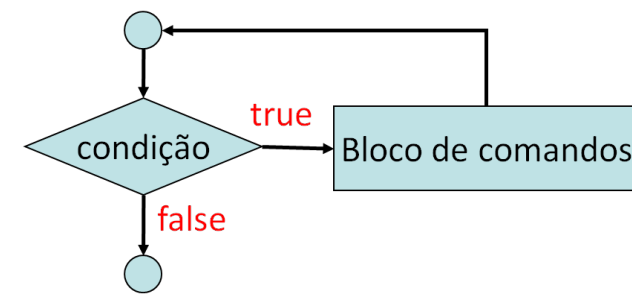
.

comandoN

fim-enquanto



Repetição com Teste no Início



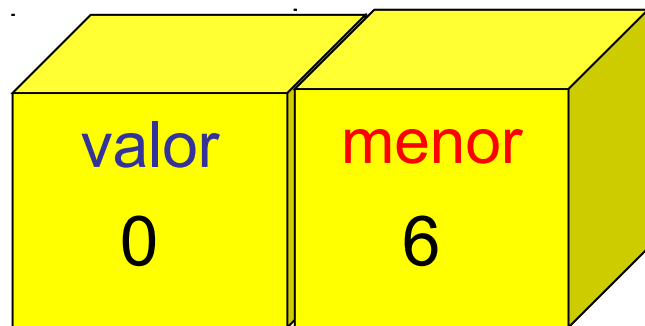
- Testa a condição antes da execução do bloco.
- Enquanto a condição for verdadeira, o bloco de comandos é executado. Assim, o bloco de comandos pode ser executado 0 ou mais vezes.
- Para a execução do bloco quando a condição se tornar falsa.
- Se já da primeira vez a condição for falsa, o bloco de comandos não é executado nenhuma vez.

Exemplo 1

- Elabore um algoritmo para determinar o menor número fornecido de um conjunto de valores inteiros positivos dados.
- Considere que o número **zero** indica o encerramento do conjunto de dados de entrada.

Elabore um algoritmo para determinar o menor número fornecido de um conjunto de valores inteiros positivos dados. Considere que o número **zero** indica o encerramento do conjunto de dados de entrada..

- O operador irá digitar os valores um a um;
- Cada valor lido será armazenado na variável **valor**.
- O menor valor será armazenado na variável **menor**
- Ao invés de tentar encontrar de uma vez o menor valor, vamos fazer isso passo a passo:
 - Quando tiver lido apenas o primeiro valor, este será o menor.
 - Cada próximo valor lido é comparado com o **menor**. Se for menor que o atual, este é substituído.



- Ao entrar o valor **0**, a escolha do menor é encerrada.

Elabore um algoritmo para determinar o menor número fornecido de um conjunto de valores inteiros positivos dados. Considere que o número **zero** indica o encerramento do conjunto de dados de entrada..

inteiro: valor, menor

início

Leia valor

menor \leftarrow valor *{ inicialização da variável que vai guardar o menor valor }*

enquanto (valor \neq 0) **faça**

se (valor < menor) **então**

 menor \leftarrow valor

fim-se

 Leia valor; *{ entrada do próximo }*

fim-enquanto

se (menor \neq 0) **então**

 Escreva 'O menor valor do conjunto é ', menor

senão

 Escreva 'Nenhum número foi digitado' *{ somente o zero do final }*

fim-se

fim

Observe que o valor 0 não faz parte do conjunto de valores da entrada.

Ele é usado como uma **sentinela**, para avisar que não há mais dados a entrar.

Observe que é preciso repetir o comando **leia** aqui.

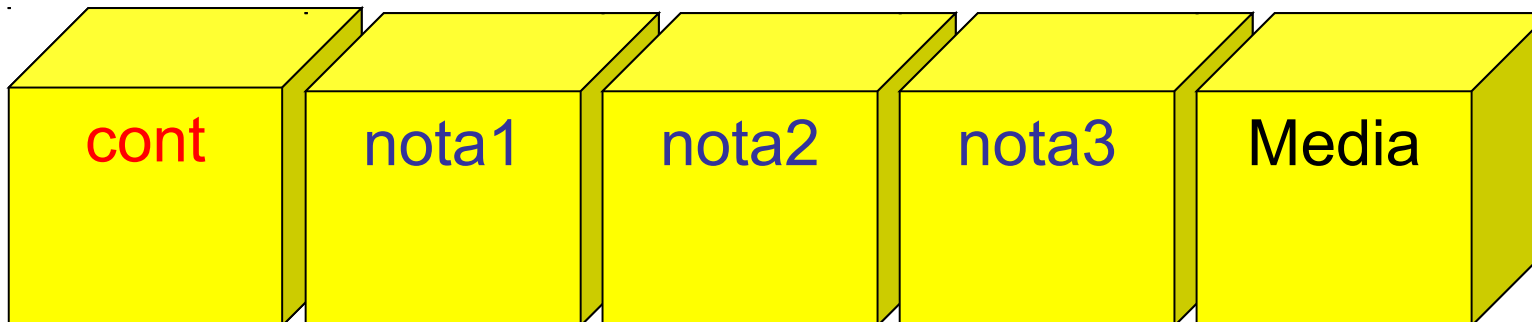
Exemplo 2

- Escrever um algoritmo que calcule as médias aritméticas de 50 alunos.

- Nesse caso, sabemos a quantidade de valores da entrada.
- Vamos usar um técnica diferente para controlar o laço (ou loop ou repetição).
- Vamos usar um **contador**
 - Uma variável que inicia com um valor definido e é incrementada cada vez que o evento que queremos contar ocorre.
 - No nosso exemplo, cada vez que uma média é processada/calculada.

Escrever um algoritmo que calcule as médias aritméticas de 50 alunos.

- Utilizaremos a variável **cont** para “contar” quantos alunos são processados
 - quantas médias são calculadas e impressas.
- Cada valor de nota lido será armazenado em uma variável **notaN**.
- Cada média calculada será armazenada na variável **Media** e em seguida, será entregue como resultado.



Escrever um algoritmo que calcule as médias aritméticas de 50 alunos.

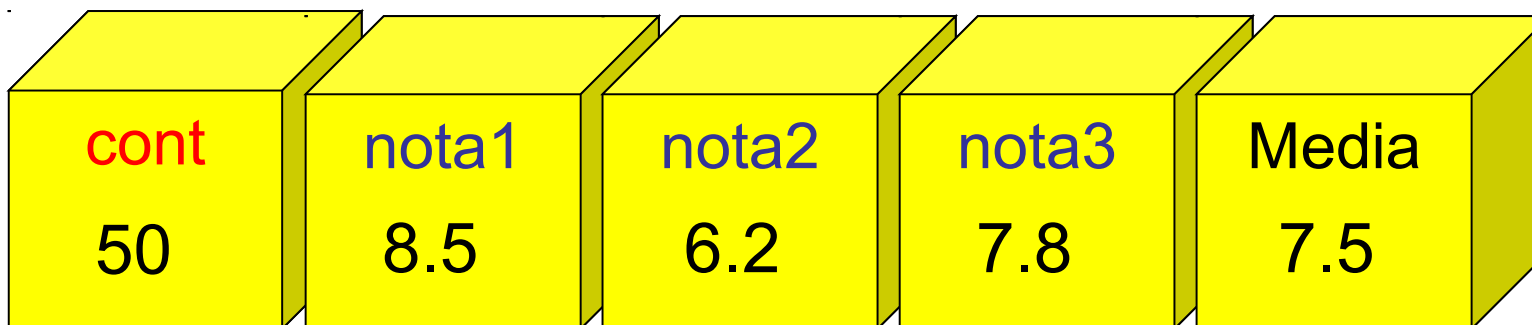
- Inicialmente, **cont** terá valor 0 ;
- Os valores das notas são lidos (suponha 8.5, 6.2 e 7.8) ;
- A média é calculada: $(\text{nota1} + \text{nota2} + \text{nota3}) / 3$;
- A média calculada é entregue: “escreva (Media)”;
- O contador é incrementado.

Isso se repetirá para cada aluno.

O contador com valor **1** indica que o trecho do programa que será repetido já ocorreu **uma** vez.

Cada vez que o trecho for executado, o contador é incrementado de 1.

Foram processados os dados de **1** aluno.



A repetição termina quando o contador atingir 50

Escrever um algoritmo que calcule as médias aritméticas de 50 alunos.

{ declaração de variáveis }

real: N1, N2, N3, *{notas parciais}*

Media *{média final}*

inteiro: cont *{contador}*

inicio

cont \leftarrow 0 *{inicialização do contador}*

enquanto (cont < 50) **faça** *{ teste da condição de parada }*

Leia N1, N2, N3 *{ entrada de dados }*

Media \leftarrow (N1 + N2 + N3)/3 *{ calculo da media }*

Escreva ' Media final = ', Media *{ saída de dados }*

cont \leftarrow cont + 1 *{ incrementa o contador em um }*

fim-enquanto

fim

repeti o laço 1 vezes
repeti o laço 2 vezes
repeti o laço 3 vezes
repeti o laço 4 vezes
repeti o laço 5 vezes
...
repeti o laço 47 vezes
repeti o laço 48 vezes
repeti o laço 49 vezes
repeti o laço 50 vezes

Qual o valor de **cont**
após sair do laço?

```
cont ← 1; {inicialização do contador}  
enquanto (cont <= 50) faça { teste da condição de parada }  
    Escreva “repeti o laço ”, cont, “vezes”  
    cont ← cont + 1  
fim-enquanto
```

Exemplo 3

- E se quisermos calcular a média da turma, ou seja, a média das médias de todos os 50 alunos?

Exemplo 3

- Nesse caso, usaremos um acumulador para acumular as médias obtidas pelos alunos;
- Após acumular todas as médias, calcularemos a média da turma.

Acumulador

- **Acumulador** é uma variável que ajuda a acumular valores a cada vez que um laço é executado.
- O acumulador deve ser **inicializado**.
- O valor adicionado ao acumulador em cada execução de um laço é variável
 - Diferentemente do contador, cujo valor adicionado é sempre constante.
- O acumulador funciona como uma caixa em que vamos colocando todos os valores que queremos acumular.
- À medida que colocamos um novo valor na caixa, ele é adicionado ao total já existente ali.

Acumulador

$\text{cont} \leftarrow 0$

$\text{acum} \leftarrow 0$ *{inicialização do acumulador}*

enquanto ($\text{cont} < 5$) **faça** *{teste da condição de parada}*

Leia Num

$\text{acum} \leftarrow \text{acum} + \text{Num}$

$\text{cont} \leftarrow \text{cont} + 1$

fim-enquanto

5

8

4

6

2

cont
5

acum
25

E se quisermos calcular a média da turma, ou seja, a média das médias de todos os 50 alunos?

{ declaração de variáveis }

real: N1, N2, N3, *{notas parciais}*

Media, MediaTurma, *{média de cada aluno e média da turma}*

acum *{acumulador das médias}*

inteiro: cont *{contador}*

inicio

cont \leftarrow 1 *{inicialização do contador}*

acum \leftarrow 0 *{inicialização do acumulador}*

enquanto (cont \leq 50) **faça** *{ teste da condição de parada }*

Leia N1, N2, N3 *{ entrada de dados }*

 Media \leftarrow (N1 + N2 + N3)/3 *{ calculo da media }*

Escreva ' **Media final =** ', Media *{ saída de dados }*

 acum \leftarrow acum + Media *{acumula a última média calculada}*

 cont \leftarrow cont + 1 *{ incrementa o contador em um }*

fim-enquanto

MediaTurma \leftarrow acum/50; *{calcula a média da turma}*

Escreva ' **A média da turma é** ', MediaTurma

fim

Exercício 1

- Nos exemplos anteriores, sabíamos previamente quantos alunos são.
- Mas... e se não soubéssemos?



Pra casa ...

Exemplo

- Construa um algoritmo que encontre a primeira potência de 2 maior do que 1000.

Construa um algoritmo que encontre a primeira potência de 2 maior do que 1000.

$2^0 = 1$	$1 > 1000 ?$	F
-----------	--------------	----------

$2^1 = 2$	$2 > 1000 ?$	F
-----------	--------------	----------

$2^2 = 4$	$4 > 1000 ?$	F
-----------	--------------	----------

...

$2^{10} = 1024$	$1024 > 1000 ?$	V
-----------------	-----------------	----------

Fim do processo !!

Construa um algoritmo que encontre a primeira potência de 2 maior do que 1000.

$$2^0 = 1 \qquad 1 > 1000 ? \qquad \text{F}$$

$$2^1 = 2 \qquad 2 > 1000 ? \qquad \text{F}$$

$$2^2 = 4 \qquad 4 > 1000 ? \qquad \text{F}$$

...

$$2^{10} = 1024 \qquad 1024 > 1000 ? \qquad \text{V}$$

Calcule a primeira potência de 2 (2^0)

Enquanto (potência < 1000)

calcule a próxima potencia de 2;

Construa um algoritmo que encontre a primeira potência de 2 maior do que 1000.

Coloque 0 em n
Coloque 2^n em $potencia$
Enquanto ($potencia < 1000$)
Adicione 1 a n
Coloque 2^n em $potencia$

n	potencia
0	$2^0 = 1$
1	$2^1 = 2$
2	$2^2 = 4$
...	
9	$2^9 = 512$
10	$2^{10} = 1024$

Construa um algoritmo que encontre a primeira potência de 2 maior do que 1000.



Por que precisa calcular **potencia** dentro e fora do laço?

{ declaração de variáveis }

inteiro: n, potencia; *{expoente, potencia}*

inicio

$n \leftarrow 0$ *{inicialização do expoente}*

$\text{potencia} \leftarrow \text{pot}(2, n)$ *{inicialização de potencia}*

enquanto ($\text{potencia} < 1000$) **faça** *{ teste da condição de parada }*

$n \leftarrow n+1$

$\text{potencia} \leftarrow \text{pot}(2, n)$ *{cálculo da nova potencia}*

fim-enquanto

Escreva 'A primeira potencia de 2 maior do que 1000 e', potencia

fim

Exercício 2

- Construa um algoritmo para calcular a soma dos 100 primeiros números naturais. Considere $N=\{1,2,\dots\}$.

Exercício 3

- Construa um algoritmo para calcular a média aritmética dos 100 primeiros n° naturais.

Exercício 4

- Construa um algoritmo para imprimir o maior inteiro positivo n cujo quadrado é inferior a um valor L dado.

Estruturas de Repetição 1

MATA37: Introdução à Lógica de Programação

Prof.: Rafael A. Melo (melo@dcc.ufba.br)

Departamento de Ciência da Computação

Instituto de Matemática

Universidade Federal da Bahia

