

Bacharelado em Sistemas de Informação

Algoritmos - Aula 06

Prof. Dr. Sérgio Luis Antonello

Prof. Me. Antero Sewaybricker Todesco

Bibliografia básica desta aula

- ✓ Medina & Fertig (2006). Capítulo 04.
- ✓ Forbellone & Eberspacher (2000). Capítulo 03.
- ✓ Ascencio & Campos (2002). Capítulo 04.

07/04/2020

Plano de Ensino

1. Unidade I – Introdução a algoritmos (objetivos a, b, c)
 - 1.1. Conceitos de abstração de dados
 - 1.2. Lógica de programação
 - 1.3. Algoritmos
 - 1.4. Formas de representação de algoritmos: pseudocódigo e fluxograma.
 - 1.5. Teste de mesa
 - 1.6. Tipos de dados
 - 1.7. Constantes e variáveis
 - 1.8. Atribuição
 - 1.9. Operadores e precedência
 - 1.10. Expressões aritméticas, relacionais e lógicas.
2. Unidade II – Estruturas básicas de controle (objetivos c, d, e)
 - 2.1. Blocos de comando
 - 2.2. Estruturas de decisão
 - 2.3. Estruturas de repetição
 - 2.4. Aninhamento
3. Unidade III – Modularização (objetivos c, d, e)
 - 3.1. Dividir para conquistar
 - 3.2. Procedimentos e funções
 - 3.3. Escopo de variáveis
 - 3.4. Parâmetros e argumentos
 - 3.5. Passagem de parâmetros por valor e por referência
4. Unidade IV – Estruturas de dados homogêneas (objetivos d, e)
 - 4.1. Vetor
 - 4.2. Matriz

Sumário

■ Primeiro momento

- ✓ Conceitos trabalhados até o momento

Segundo momento

- ✓ Estruturas básicas de controle: repetição
 - ✓ Comando Para

Terceiro momento: síntese

Primeiro momento: Revisão

- ✓ Vimos que os dados manipulados pelos algoritmos podem ser dos seguintes tipos primitivos: **inteiro**, **real**, **literal** ou **lógico**.
- ✓ Para guardar os dados precisamos de **identificadores** que servem de rótulo para dados variáveis e constantes, e que para usá-los é necessária a **declaração**, na qual associamos o identificador a um dos tipos primitivos válidos.

Primeiro momento: Revisão

- ✓ Vimos também os operadores aritméticos, lógicos e relacionais, que permitem escrever expressões aritméticas, lógicas e relacionais, sendo que as duas últimas devem resultar em um valor lógico, verdadeiro (v) ou falso (f).
 - ✓ Comandos de entrada e saída de dados.
-

Primeiro momento: Revisão

- As estruturas de **seleção ou decisão** são utilizadas quando existe a necessidade de verificar condições para a realização de uma instrução ou de uma sequência de instruções.
- Elas podem ser do tipo **simples, composta ou encadeada**.

Primeiro momento: Revisão

- Correção de exercícios

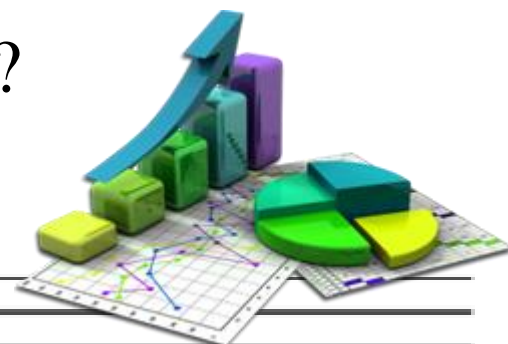
Segundo momento: motivação

- No autoatendimento de um banco, ao terminar uma determinada operação, o que acontece?
- Em um caixa de supermercado, o que acontece para que consigamos passar os produtos e saber quanto fica a compra a ser paga?



Segundo momento: motivação

■ Como é montado um relatório de vendas?



RELATÓRIO DE VENDAS (RECARGA DE CARTÕES)

Período: 01/04/2015 a 30/04/2015

VENDAS POR DATA E ESTAÇÃO DE VENDA

Data	Refeição	Est. de Venda	Usuário	Valor	Qtd.	Sub. Total
30/04/2015	Almoço	Restaurante Universitário - Estação 1	Login do Usuário	R\$ 3,00	482	1.446,00
30/04/2015	Almoço	Restaurante Universitário - Estação 1	Login do Usuário	R\$ 4,50	6	27,00
30/04/2015	Almoço	Restaurante Universitário - Estação 2 (C. CONVIVENCIA)	Login do Usuário	R\$ 3,00	291	873,00
30/04/2015	Almoço	Restaurante Universitário - Estação 2 (C. CONVIVENCIA)	Login do Usuário	R\$ 3,00	47	141,00
30/04/2015	Almoço	Restaurante Universitário - Estação 2 (C. CONVIVENCIA)	Login do Usuário	R\$ 4,50	4	18,00
01/04/2015	Almoço	Restaurante Universitário - Estação 2 (C. CONVIVENCIA)	Login do Usuário	R\$ 3,00	352	1.056,00
01/04/2015	Almoço	Restaurante Universitário - Estação 2 (C. CONVIVENCIA)	Login do Usuário	R\$ 4,50	4	18,00
01/04/2015	Jantar	Restaurante Universitário - Estação 1	Login do Usuário	R\$ 3,00	91	273,00
01/04/2015	Jantar	Restaurante Universitário - Estação 1	Login do Usuário	R\$ 4,00	2	8,00
01/04/2015	Jantar	Restaurante Universitário - Estação 2 (C. CONVIVENCIA)	Login do Usuário	R\$ 3,00	106	318,00
					29200	88.267,50

RESUMO DE VENDAS POR ESTAÇÃO

Restaurante	Quantidade
Restaurante Universitário - Estação 1	16678
Restaurante Universitário - Estação 2 (C. CONVIVENCIA)	12522
	29200

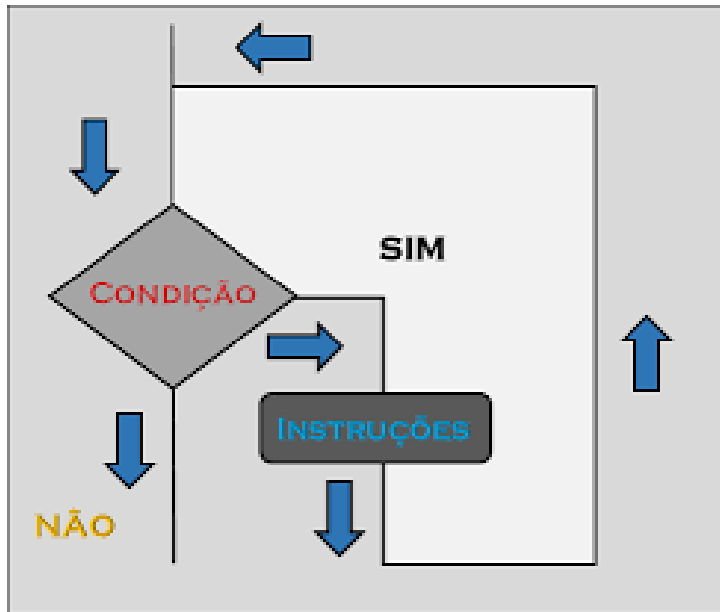
Segundo momento: motivação

- O que é fundamental na operações de equipamentos na indústria automatizada?



Segundo momento

- Estrutura de repetição: Para



Estruturas de Repetição

- Em determinadas situações, **é necessário repetir o programa ou parte dele inúmeras vezes**, como no cálculo das médias das notas de um grupo de alunos.
- Reiniciar o programa para cada cálculo não é uma solução muito prática e muitas vezes inviável.
- Uma solução comum é a utilização de estruturas de repetição.

Estruturas de Repetição

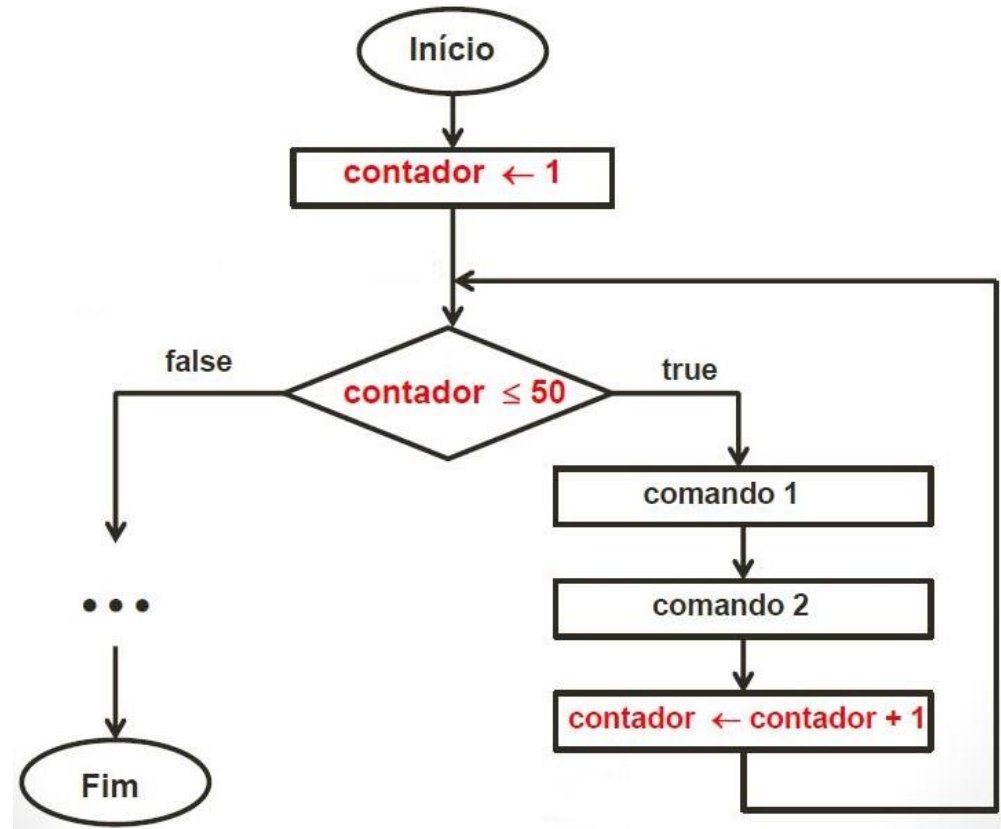
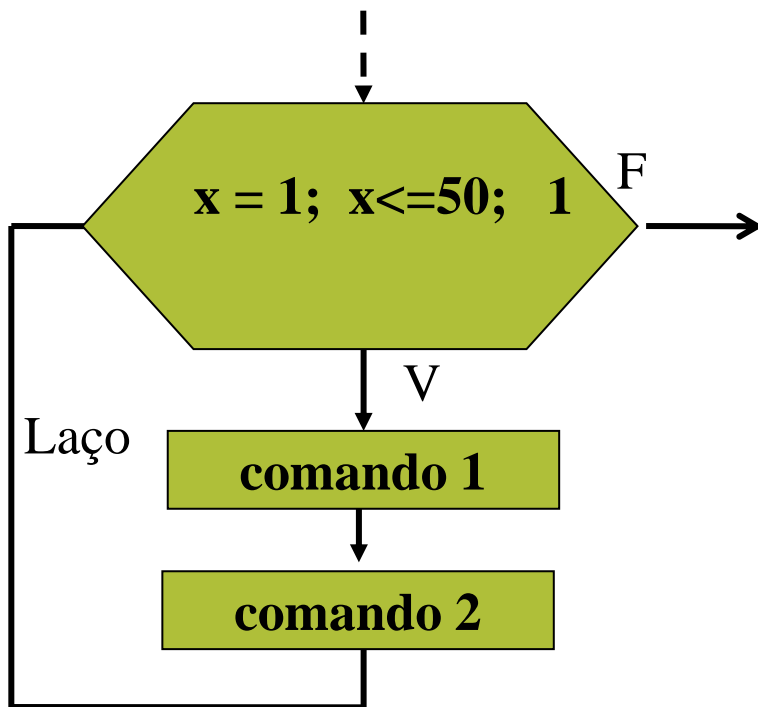
- O conceito de repetição (**loop**) é utilizado quando se deseja repetir um certo trecho de instruções por um número de vezes.
- O número de repetições pode ser conhecido anteriormente ou não, mas necessariamente **precisa ser finito**.
- Neste conceito, **laço** é o bloco de instruções que serão realizadas repetidas vezes e que estão contidas em uma estrutura de repetição.

Estrutura PARA

- A estrutura de repetição **Para** possibilita **repetir, uma quantidade predefinida de vezes**, uma instrução ou um conjunto de instruções, que serão executadas dentro do loop (laço).
- Ela utiliza variáveis de controle que indicam exatamente o número de vezes que a sequência de instruções será executada.

Estrutura PARA

Fluxograma



Estrutura PARA

Pseudocódigo

*Para (var de valorInicial Ate valorLimite Passo incremento) Faca
Conjunto de instruções*

FimPara

var	É a variável contadora que controla o número de repetições do laço.
valorInicial	Especifica o valor inicial da variável contadora antes da primeira repetição do laço.
valorLimite	Especifica o valor máximo que a variável contadora pode alcançar.
incremento	Especifica o incremento que será acrescentado à variável contadora em cada repetição do laço. Também pode ocorrer valores negativos (decremento).
FimPara	Indica o fim da sequência de comandos a serem repetidos. Cada vez que o programa chega neste ponto, é acrescentado à variável contadora o valor de incremento, e comparado a valorLimite. Se for menor ou igual (ou maior ou igual, quando incremento for negativo), a sequência de comandos será executada mais uma vez; caso contrário, a execução do programa prosseguirá a partir do primeiro comando existente após o FimPara.

Estrutura PARA

Analise este caso

O pseudocódigo ao lado calcula a média das notas de um aluno.

```
algoritmo "Media"  
// Função : calcular a média de um aluno  
var  
    prova, trab, apres, media: real  
inicio  
    Leia (prova)  
    Leia (trab)  
    Leia (apres)  
  
    media <- (prova*2 + trab + apres) / 4  
  
    Escreva (media)  
fimalgoritmo
```

Como fazer esse algoritmo rodar para uma turma toda?

Estrutura PARA

Agora o algoritmo calcula a média de todos os alunos de uma turma

```
algoritmo "Media"  
// Função : calcular a média de um aluno  
var  
    prova, trab, apres, media: real  
    ctd, nroalunos: inteiro  
inicio  
    Leia (nroalunos)  
    Para ctd de 1 Ate nroalunos Passo 1 Faca  
        Leia (prova)  
        Leia (trab)  
        Leia (apres)  
  
        media <- (prova*2 + trab + apres) / 4  
        Escreval(media)  
    FimPara  
fimalgoritmo
```

Melhorando a comunicação com o usuário

```
algoritmo "Media"
// Função :  calcular a média de um aluno
var
    prova, trab, apres, media: real
    ctd, nroalunos: inteiro
inicio
    Escreva("Quantidade de alunos da turma: ")
    Leia (nroalunos)

    Para ctd de 1 Ate nroalunos Passo 1 Faca
        Escreva("Nota da prova: ")
        Leia (prova)
        Escreva("Nota do trabalho: ")
        Leia (trab)
        Escreva("Nota da apresentacao: ")
        Leia (apres)

        media <- (prova*2 + trab + apres) / 4

        Escreval("Nota final: ", media:3:1)
        Escreval()
    FimPara
fimalgoritmo
```

Vamos usar o Programar!



Exercícios

- 1) Durante as aulas no laboratório Joãozinho vivia bagunçando e não prestava atenção. A professora em um dado momento pediu que ele escrevesse 100 vezes a frase "Não vou mais fazer bagunça durante a aula". Acontece que Joãozinho é muito "esperto" e convidou você para fazer um programa que exiba as frases solicitadas na tela do computador. Codifique o programa solicitado.
 - 2) Você foi convidado a fazer um programa que apresente o produto da multiplicação de dois números informados pelo usuário durante a execução do programa. Acontece que o operador de multiplicação (*) não está funcionando. De qualquer forma você deve fazer o programa.
-

Exercícios

3) URI 1070 - Seis Números Ímpares (tarefa)

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1070>

4) Obter a soma dos números múltiplos de 4 de um intervalo definido por dois números fornecidos pelo usuário. Cada múltiplo deve ser exibido na tela. (tarefa)
