

INE5231 Computação Científica I

Prof. A. G. Silva

14 de março de 2017

Conteúdo programático

- **O computador - [3 horas-aula]**
- **Representação de algoritmos - [3 horas-aula]:** ● Conceito de Algoritmo ● Pseudo-Código para Representar Algoritmos ● Tipos de Dados ● Variáveis ● Tipos de Variáveis ● Desenvolvimento Modularizado de Algoritmos
- **Linguagens de programação estruturadas [3 horas-aula]**
- **Introdução à programação em C [6 horas-aula]**
- **Programas envolvendo processos de repetição e seleção [6 horas-aula]**
- **Variáveis estruturadas unidimensionais homogêneas [9 horas-aula]**
- **Variáveis estruturadas multidimensionais homogêneas [6 horas-aula]**
- **Variáveis estruturadas heterogêneas [6 horas-aula]**
- **Subdivisão de problemas e subprogramação [6 horas-aula]**
- **Programação utilizando uma linguagem de computação técnica numérica [6 horas-aula]**

Lógica de programação

- Lógica de programação
 - ▶ Técnica de **encadear** pensamentos para atingir determinado objetivo
 - ▶ Necessária para desenvolver programas e sistemas, pois permite definir a **sequência lógica** para a solução de um problema
 - ★ ? → 1. → 2. → 3. → !
 - ★ Estes pensamentos podem ser descritos como uma **sequência de instruções**, que devem ser seguidas para se cumprir uma determinada tarefa
 - ★ **Passos** executados até se atingir um objetivo ou solução de um problema

Instrução

- Cada um dos **passos** representa uma ação a tomar (obedecendo a sequência lógica) para resolver um problema
- **Operações** elementares são executadas. Ex.: “somar”, “subtrair”, “comparar se é maior”, etc
- Uma só instrução não resolve problemas
 - ▶ Executar um **conjunto de instruções**
 - ▶ Executar em uma **sequência lógica**

Exemplo abstrato

- Exemplo de como “fazer omelete”
 - ▶ Instruções: “quebrar ovos”, “bater ovos”, “pôr sal”, “ligar fogão”, “pôr óleo na frigideira”, “pôr frigideira no fogo”, “fritar ovos batidos”, etc...
- Quanto às instruções isoladas:
 - ▶ Só “quebrar ovos”, ou só “pôr óleo na frigideira”, não é suficiente para cumprir a tarefa “fazer omelete”
- Quanto à sequência lógica:
 - ▶ Se executarmos “fritar ovos batidos” antes de “bater ovos”, ou pior, antes de “quebrar ovos”, não iremos cumprir a tarefa “fazer omelete”

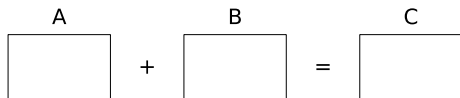
Algoritmo

- Conjunto finito de passos/instruções
- Usualmente, inicia com a primeira instrução
- Execução sequencial, uma instrução de cada vez, com possibilidade de saltos para outras instruções
- Instruções individuais suficientemente elementares ou primitivas
- Sempre deve alcançar uma instrução FIM ou PARE, para terminar a execução
- Utiliza dados (entrada) e gera um resultado (saída)
- Claro e preciso.

Exemplo de algoritmos

- Algoritmo para somar dois números:

- 1 Escrever primeiro número no retângulo A
- 2 Escrever segundo número no retângulo B
- 3 Somar o número do retângulo A com o número do retângulo B, e escrever o resultado no retângulo C



- Algoritmo para preparar um bolo:

- 1 Bata 4 claras em neve
- 2 Adicione 2 xícaras de açúcar
- 3 Adicione 2 colheres de farinha de trigo, 4 gemas, uma colher de fermento e duas colheres de chocolate
- 4 Bata por 3 minutos
- 5 Unte uma assadeira com margarina e farinha de trigo
- 6 Coloque o bolo para assar por 20 minutos

Exemplo de algoritmo – troca de pneu

- Algoritmo para troca de um pneu furado:
 - 1 Verifica qual pneu está furado
 - 2 Posiciona o macaco para levantar o carro
 - 3 Pega o estepe
 - 4 Solta os parafusos
 - 5 Substitui o pneu furado
 - 6 Recoloca os parafusos
 - 7 Desce o carro
 - 8 Guarda o macaco e o pneu furado

Programa

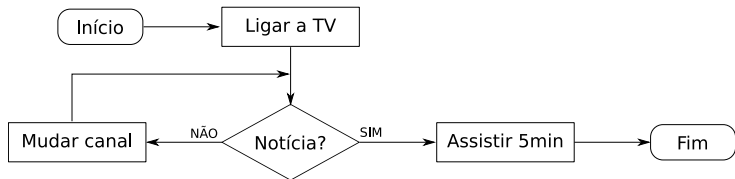
- Programa:

- 1 Algoritmo escrito em uma linguagem de computador (linguagem de programação - C, Pascal, COBOL, Fortran, Basic, Java, etc.)
- 2 Interpretado e executado por um computador
- 3 Interpretação rigorosa, exata, do computador \Rightarrow escrita do algoritmo na linguagem de programação tem que seguir regras mais rigorosas

Fluxograma

- Fluxograma:

- ▶ É uma forma universal de representação, pois se utiliza de figuras geométricas para ilustrar passos a serem seguidos para a resolução de problemas
- ▶ Elementos fundamentais:
 - ★ Início e fim: retângulos com cantos arredondados
 - ★ Processamento: retângulos com instruções
 - ★ Decisões: losangos com condições



Pseudocódigo

- Pseudocódigo:
 - ▶ Pseudocódigo = “código-falso”
 - ▶ Intermediária entre linguagem natural e linguagem de programação
 - ▶ Facilita descrever o algoritmo antes da escrita formal para uma linguagem de programação
 - ▶ Descrição do algoritmo menos rigorosa que na linguagem de programação (código-fonte)
 - ▶ Fácil de entender e fácil de codificar depois
 - ▶ Independente da linguagem de programação

Desenvolvimento de algoritmos

- Fases para o desenvolvimento de um algoritmo:

- ▶ Definir e delimitar o problema
- ▶ Dividir a solução em três partes



- Exemplo:

- ▶ **Problema:** cálculo da média de quatro números
 - ★ Dados de entrada: os números N_1 , N_2 , N_3 e N_4
 - ★ Processamento: somar os quatro números e dividir esta soma por quatro

$$\frac{N_1 + N_2 + N_3 + N_4}{4}$$

- ★ Dado de saída: a média final

Problema da média

- Algoritmo

- ▶ Receber o primeiro número
- ▶ Receber o segundo número
- ▶ Receber o terceiro número
- ▶ Receber o quarto número
- ▶ Somar os quatro números
- ▶ Dividir a soma por quatro
- ▶ Mostrar o resultado da divisão

Teste de mesa

- Os algoritmos devem ser testados
- Usar dados e resultados previamente calculados, seguir precisamente as instruções do algoritmo e verificar se o procedimento está correto ou não
- Exemplo: teste de mesa para o algoritmo da média entre quatro valores

N_1	N_2	N_3	N_4	Média

Variável

- Representa uma posição na memória, onde pode ser armazenado um dado
- Possui um nome (identificador) e um valor
- Durante a execução do algoritmo, pode ter seu valor alterado (seu valor pode variar)
- Mudanças no valor das variáveis:
 - ▶ Por entrada de dados. Exemplo: “Ler N1”
(lê-se: a variável N1 recebe um valor de entrada, por exemplo, digitado pelo teclado)
 - ▶ Por atribuição. Exemplo: “MEDIA \leftarrow <valor>”
(lê-se: a variável MEDIA recebe o valor explicitado na própria instrução)

Exemplo de sequência

- Problema: cálculo da média entre quatro números
- Pseudocódigo:
 - ▶ Ler N1
 - ▶ Ler N2
 - ▶ Ler N3
 - ▶ Ler N4
 - ▶ $MEDIA \leftarrow (N1 + N2 + N3 + N4) / 4$
 - ▶ Mostrar MEDIA
- N1, N2, N3, N4 e MEDIA são variáveis

Atribuição

- Atribui o valor da direita à variável da esquerda
- O valor pode ser uma constante, uma variável ou uma expressão

▶ $MEDIA \leftarrow (N1 + N2 + N3 + N4) / 4$
(lê-se: a variável “MEDIA” recebe o resultado da conta/fórmula)

- Outros exemplos

▶ $a \leftarrow 3$
(lê-se: a variável “a” recebe o valor 3)

▶ $a \leftarrow x$
(lê-se: a variável “a” recebe o valor da variável “x”)

Operadores aritméticos

- Operadores aritméticos básicos em pseudolinguagem:

<code>+</code>	\Rightarrow	adição
<code>-</code>	\Rightarrow	subtração
<code>*</code>	\Rightarrow	multiplicação
<code>/</code>	\Rightarrow	divisão
<code>^</code> ou <code>**</code>	\Rightarrow	exponenciação

Precedência de operadores básicos

- Regras de precedência de operações em pseudolinguagem:

1º	()	parênteses
2º	^ ou **	exponenciação
3º	*, /	multiplicação, divisão (esquerda para direita)
4º	+, -	adição, subtração (esquerda para direita)

- Exemplos:

▶ $1 + 7 * 2 ^ 2 - 1$ (resultado = 28)

▶ $3 * (1 - 2) + 4 * 2$ (resultado = 5)

Operadores relacionais e lógicos

- Operadores que resultam um valor lógico (*boolean*), ou seja, verdadeiro (V) ou falso (F). Exemplo:

```
aprovado ← nota >= 6.0;
```

- Operadores relacionais em pseudolinguagem:

>	⇒	maior que
>=	⇒	maior ou igual a
<	⇒	menor que
<=	⇒	menor ou igual a
=	⇒	igual a
<>	⇒	diferente de

- Conectores e negação lógicos em pseudolinguagem:

e	⇒	“E” lógico
ou	⇒	“OU” lógico
nao	⇒	“NÃO” lógico

Expressões lógicas

- Exemplo: (`nota >= 6.0`) e (`frequencia >= 75`)
V = verdadeiro, F = falso
- Tabela-verdade do operador `e`:

A	B	A e B
V	V	V
V	F	F
F	V	F
F	F	F

- Tabela-verdade do operador `ou`:

A	B	A ou B
V	V	V
V	F	V
F	V	V
F	F	F

Exemplo de operadores relacionais

- Exemplo de operações relacionais em pseudolinguagem:

A=5 e B=3

A = B	falso
A <> B	verdadeiro
A > B	verdadeiro
A < B	falso
A >= B	verdadeiro
A <= B	falso

Exemplo de operadores relacionais e lógicos

- Exemplo de operações relacionais e lógicas em pseudolinguagem:

A=5, B=8 e C=1

$(A = B) \text{ E } (B > C)$	falso (F e V)
$(A <> B) \text{ OU } (B < C)$	verdadeiro (V ou F)
$\text{NÃO}(A > B)$	verdadeiro (não F)
$(A < B) \text{ E } (B > C)$	verdadeiro (V e V)
$(A \geq B) \text{ OU } (B = C)$	falso (F ou F)
$\text{NÃO}(A \leq B)$	falso (não V)

Precedência de operações (versão ampliada)

- Regras de precedência de operações em pseudolinguagem:

1º	()	parênteses
2º	+, -, nao	positivo, negativo, negação (unários)
3º	^ ou **	exponenciação
4º	*, /	multiplicativos
5º	<, <=, >, >=	relacionais
6º	=, <>	igualdade, diferença
7º	+, -	aditivos
8º	e	conjunção
9º	ou	disjunção

Exercícios



- Lista 1 de exercícios no Moodle: [lista1.pdf](#) (*não é necessária entrega*)

Referências

- Notas de aula das professoras Vania Bogorny, Patrícia Jaques e Mônica Py
- Ferramentas para escrever algoritmos:
 - ▶ Portugol Online
<https://vinyanalista.github.io/portugol/>
 - ▶ VISALG
<https://sourceforge.net/projects/visualg30/>
 - ▶ Portugol Studio
<https://sourceforge.net/projects/portugolstudio/>