

**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE**

Algoritmos e Técnicas de Programação

Estrutura, Visualg e Variáveis

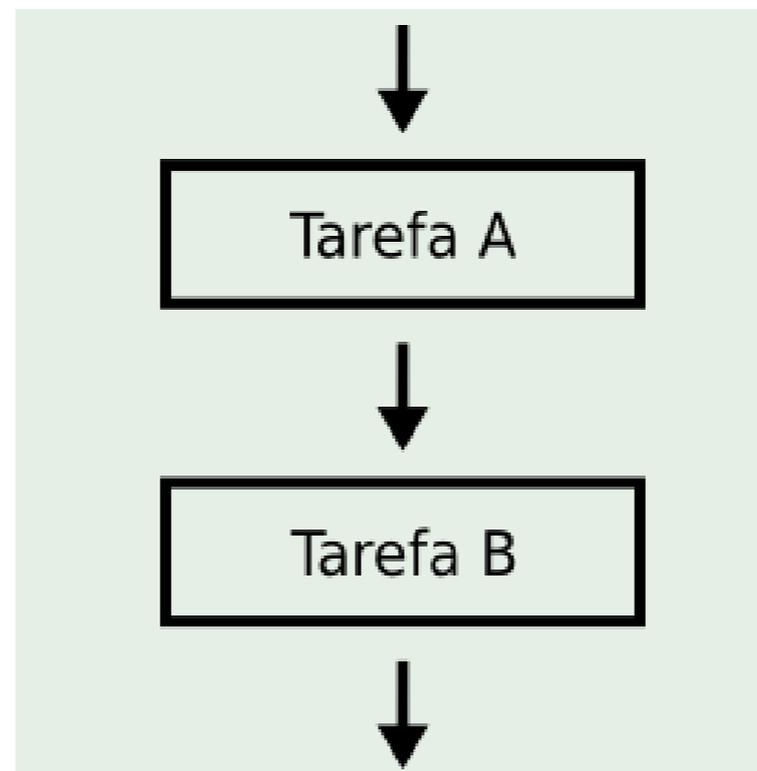
filipe.raulino@ifrn.edu.br

Programação Estruturada

- ▶ A programação estruturada (Top-Down) estabelece uma disciplina de desenvolvimento de algoritmos que facilita a compreensão de programas através do número restrito de mecanismos de controle da execução de programas;
- ▶ Qualquer algoritmo, independentemente da área de aplicação, de sua complexidade e da linguagem de programação na qual será codificado, pode ser descrito através destes mecanismos básicos;
- ▶ O princípio básico de programação estruturada é que um programa é composto por blocos elementares de código que se interligam através de três mecanismos básicos, que são:
 - ▶ Sequência;
 - ▶ Seleção; e
 - ▶ Iteração;
- ▶ Cada uma destas construções tem um ponto de início (o topo do bloco) e um ponto de término (o fim do bloco) de execução.

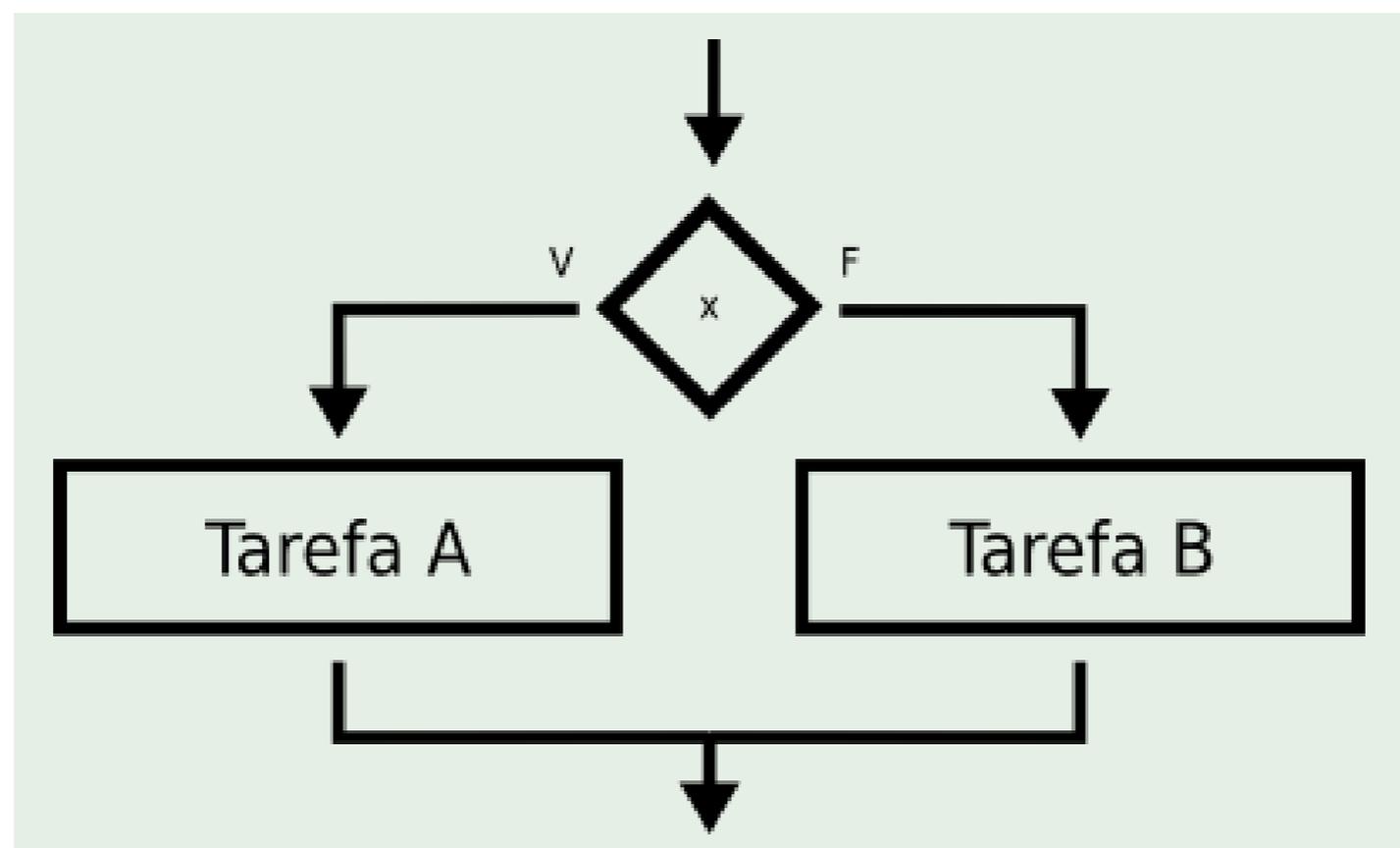
Mecanismo de Sequência

Implementa os passos de processamento necessários para descrever qualquer programa. Por exemplo, um segmento de programa da forma “faça primeiro a Tarefa A e depois a Tarefa B”.



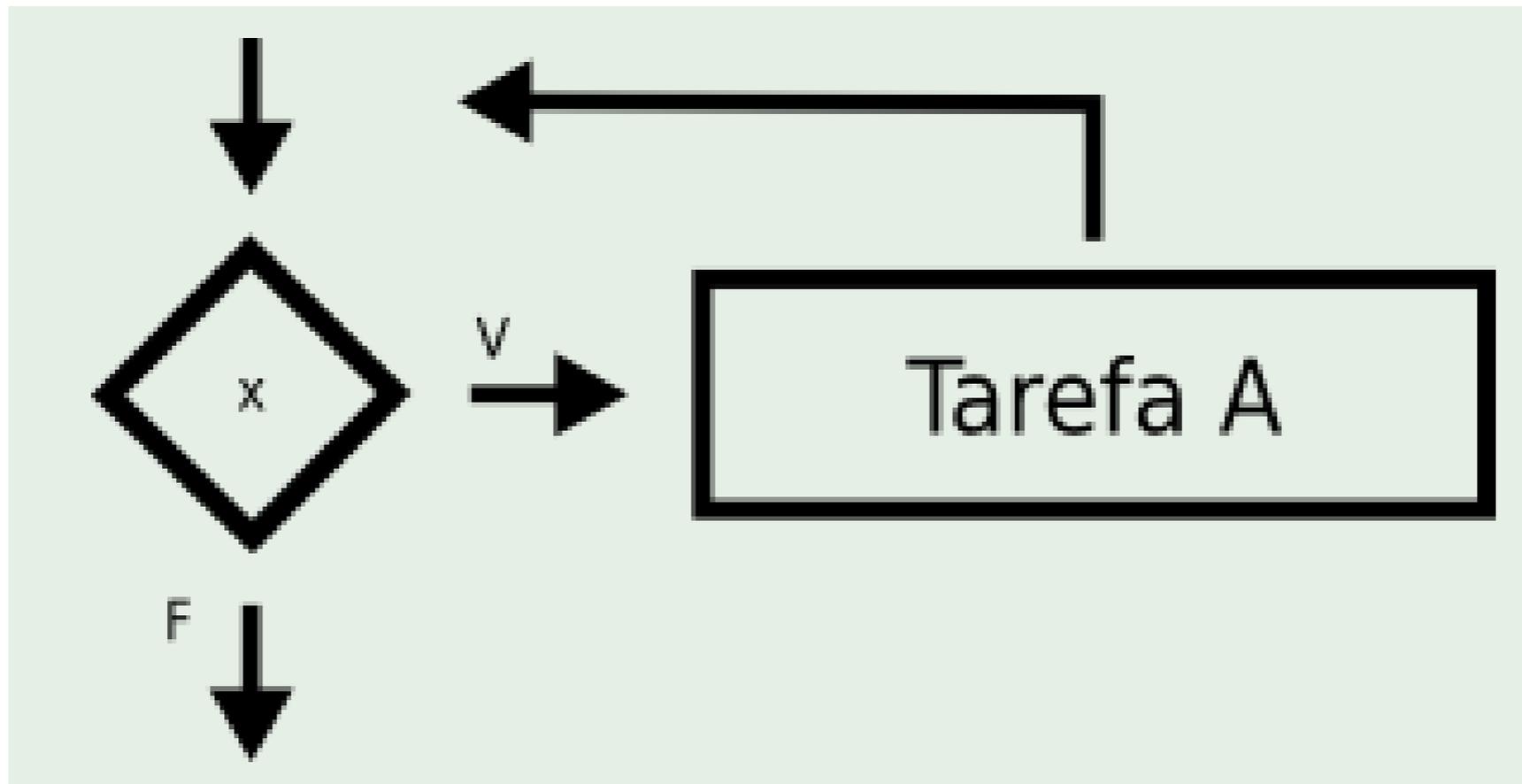
Mecanismo de Seleção

Especifica a possibilidade de selecionar o fluxo de execução do processamento baseado em ocorrências lógicas. Por exemplo, um segmento de seleção permite representar fluxos da forma “se a condição lógica x for verdadeira, faça a Tarefa A; senão (isto é, se a condição x for falsa), faça a Tarefa B.”



Mecanismo de Iteração

Permite a execução repetitiva de segmentos do programa. Na forma básica de repetição, uma condição lógica é verificada. Caso seja verdadeira, o bloco de tarefas associado ao comando é executado. A condição é então reavaliada; enquanto for verdadeira, a tarefa é repetidamente executada.



Estrutura de Um algoritmo

Nome do Programa

Inclusão de Bibliotecas

Declaração de Variáveis Globais

Criação de Procedimentos e Funções

Indicador de Início do Programa (Início)

Códigos do Programa

Indicador de Término do Programa (Fim)

Visualg

- ▶ O Visualg é uma ferramenta (software) que permite aos alunos iniciantes em programação o exercício dos seus conhecimentos num ambiente próximo da realidade;
- ▶ Linguagem parecida com o "Portugol", de grande popularidade nos meios acadêmicos e presente nos livros mais utilizados;
- ▶ É capaz de simular o que acontece na tela do computador com o uso dos famosos comandos "leia" e "escreva".

O VISUALG

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 16/10/2012  
// Seção de Declarações
```

var

Declaração de variáveis;
As variáveis devem ser declaradas aqui.

```
inicio  
// Seção de Comandos  
fimalgoritmo
```

Fim do código;

Início do código;

“semnome” é o nome do algoritmo, não tem influência sobre o código.

Comentários;

Não são obrigatórios, mas ajudam no entendimento do código. No visualg utiliza-se // para iniciar um comentário.

Início do código;

Nessa área deve vir o código;

O VISUALG

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 16/10/2012  
// Seção de Declarações  
var  
  
inicio  
// Seção de Comandos  
fimalgoritmo
```

- ▶ As palavras em **azul** são próprias da linguagem utilizada no Visualg, ou seja, são palavras reservadas e **não** podem ser utilizadas como nome de variáveis ou funções, por exemplo;

HELLO, WORLD!

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 16/10/2012  
// Seção de Declarações  
var  
  
inicio  
// Seção de Comandos  
escreva ("Hello, World!")  
fimalgoritmo
```

- ▶ O propósito do nosso primeiro algoritmo é escrever na tela a frase Hello, World!
- ▶ Para isso, usamos o comando *escreva()*;
- ▶ Dentro dos () deve vir a nossa frase entre aspas, haja vista que estamos querendo escrever na tela uma string (cadeia de caracteres).

A instrução de saída de dados é a instrução através da qual o computador se comunica com usuário durante a execução do algoritmo. Isso é feito, geralmente, através da exibição de alguma informação na tela.

- ▶ O visualg possui dois comandos para escrever na tela:
 - ▶ `Escreva()`
 - ▶ `Escreval()`
- ▶ A única diferença é que o segundo, `escreval()`, pula uma linha em seguida.

COMANDOS DE SAÍDA DE DADOS

- ▶ Execute os exemplos a seguir e verifique a diferença.

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 16/10/2012  
// Seção de Declarações  
var  
  
inicio  
// Seção de Comandos  
escreva ("Hello, World!")  
escreva ("Meu nome é Visualg.")  
fimalgoritmo
```

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 16/10/2012  
// Seção de Declarações  
var  
  
inicio  
// Seção de Comandos  
escreval ("Hello, World!")  
escreval ("Meu nome é Visualg.")  
fimalgoritmo
```

Exercícios

Variável

- ▶ Uma variável representa uma posição de memória;
- ▶ Possui um nome e tipo;
- ▶ Seu conteúdo pode variar ao longo do tempo, durante a execução do programa;
- ▶ Embora uma variável possa assumir diferentes valores, ela só pode armazenar um valor a cada instante; e
- ▶ Não existe limite para o número de variáveis em um programa, porém cada variável criada ocupa um espaço de memória de acordo com seu tipo e seu tamanho. Em outras palavras, quanto maior o número de variáveis utilizadas, maior o gasto de memória pelo programa desenvolvido.

Tipos de Dados

O tipo de dados indica qual valor pode ser armazenado em uma posição de memória.

Os tipos de dados mais utilizados são:

- ▶ numérico;
- ▶ lógico; e
- ▶ literal ou caracteres.

Tipo Numérico

Os dados numéricos dividem-se em dois grupos:

▶ **Inteiros**

- ▶ Os números inteiros podem ser positivos ou negativos e não possuem parte fracionária.

-2 -1 0 1 2

▶ **Reais**

- ▶ Os dados reais podem ser positivos ou negativos e possuem parte fracionária.

-8.5 -0.5 0.0 1.234 10.32

Tipo Lógico

São também chamados de dados booleanos (por causa da álgebra de Boole).

Verdadeiro ou Falso

0 ou 1

0 ou 1

Tipo Literal ou Caracteres

São formados por um único caractere ou por uma cadeia de caracteres. Esses caracteres podem ser as letras maiúsculas, as letras minúsculas, os números e os caracteres especiais.

“D” “Aluno” “sala1” “@ifrn.edu.br” “(84) 9999-9999”

Formação de Identificadores

Os identificadores são os nomes das variáveis, dos programas, dos procedimentos e das funções.

As regras básicas para a formação dos identificadores são:

- ▶ Os caracteres que você pode utilizar são: os números, as letras maiúsculas, as letras minúsculas e o caractere sublinhado;
- ▶ O primeiro caractere deve ser sempre uma letra ou um caractere sublinhado;
- ▶ Não são permitidos espaços em branco e caracteres especiais; e
- ▶ Não podemos usar as palavras reservadas nos identificadores, ou seja, palavras que pertençam a uma linguagem de programação.

A a cont var nota data_nascimento

Variáveis

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 16/10/2012  
// Seção de Declarações
```

```
var  
alg1, alg2 : inteiro  
num1, num2 : real  
nome : caractere  
flag : logico
```

```
inicio  
// Seção de Comandos
```

```
fimalgoritmo
```

- ▶ O visualg prevê quatro tipos de dados:
 - ▶ inteiro
 - ▶ real
 - ▶ caractere
 - ▶ logico
- ▶ Observe que elas não têm acentuação.
- ▶ As palavras que definem os tipos de dados também são reservadas.

Comandos de Entrada de Dados

```
algoritmo "semnome"  
// Função :  
// Autor :  
// Data : 16/10/2012  
// Seção de Declarações  
var  
x : inteiro  
inicio  
// Seção de Comandos  
leia (x)  
escreva (x)  
fimalgoritmo
```

- ▶ É o meio pelo qual as informações (mais especificamente os dados) são transferidas pelo usuário ou pelos níveis secundários de memória ao computador.
- ▶ Utiliza-se o `leia()` para receber os valores digitados pelo usuário, atribuindo-os às variáveis correspondentes.

Exercícios
