

Algoritmos e Estruturas de Dados 1

Conceitos Iniciais

Prof. Ivre Marjorie (ivre.marjorie@gmail.com)

Desenvolvimento de um programa

- Etapas para o desenvolvimento de um programa:

1- Análise

- Nesta etapa estuda-se o enunciado do problema para definir a entrada de dados, o processamento e a saída de dados

2- Algoritmo

- Ferramentas do tipo descrição narrativa, fluxograma ou pseudocódigo são utilizadas para descrever o problema com suas soluções

3- Codificação

- O algoritmo é então transformado em códigos da **linguagem de programação** escolhida para se trabalhar

Conceito de Algoritmo

▶ Algoritmo é:

- ▶ “uma sequência de passos que visa atingir um **objetivo bem definido**”
- ▶ “a descrição de uma sequência de passos que deve ser seguida para a realização de **uma tarefa**”
- ▶ “uma sequência finita de instruções ou operações cuja execução, em tempo finito, resolve um problema computacional, qualquer que seja sua instância”



Exemplos de Algoritmo

▶ **Algoritmo 1 - Somar três números:**

Passo 1 – Receber três números

Passo 2 – Somar os três números

Passo 3 – Mostrar o resultado obtido



Exemplos de Algoritmo

► **Algoritmo 2 - Fazer um sanduíche:**

Passo 1 – Pegar o pão

Passo 2 – Cortar o pão no meio

Passo 3 – Pegar a maionese

Passo 4 – Passar a maionese no pão

Passo 5 – Pegar e cortar alface e tomate

Passo 6 – Colocar alface e tomate no pão

Passo 7 – Pegar o hambúrguer

Passo 8 – Fritar o hambúrguer

Passo 9 – Colocar o hambúrguer no pão



Exemplos de Algoritmo

► Algoritmo 3 – Ir para a Escola:

Passo 1 – Acordar cedo

Passo 2 – Ir ao banheiro

Passo 3 – Abrir o armário e escolher uma roupa

Passo 4 – **Se** o tempo estiver quente, pegar uma camiseta e uma calça jeans; **caso contrário**, pegar um agasalho e uma calça jeans

Passo 5 – Vestir a roupa escolhida

Passo 6 – Tomar café

Passo 7 – Pegar uma condução

Passo 8 – Descer próximo à escola





Exemplos de Algoritmo

► **Algoritmo 4 - Sacar dinheiro no Banco 24 horas:**

Passo 1 – Ir até um Banco 24 horas

Passo 2 – Colocar o cartão

Passo 3 – Digitar a senha

Passo 4 – Retirar o cartão

Passo 5 - Escolher a opção de saque

Passo 6 – Solicitar a quantia desejada

Passo 7 – **Se** o saldo for maior ou igual a quantia desejada, sacar;
caso contrário, mostrar mensagem de impossibilidade de saque

Passo 8 – Sair do Banco 24 horas



Construção de Algoritmos

► **Passos:**

- a) Compreender completamente o problema a ser resolvido, destacando os pontos mais importantes e os objetos que o compõem
- b) Definir os dados de entrada, ou seja, quais dados serão fornecidos e quais objetos fazem parte desse cenário-problema
- c) Definir o processamento, ou seja, quais cálculos serão efetuados e quais restrições para esses cálculos. O processamento é responsável pela transformação dos dados de entrada em dados de saída.



Construção de Algoritmos

► **Passos:**

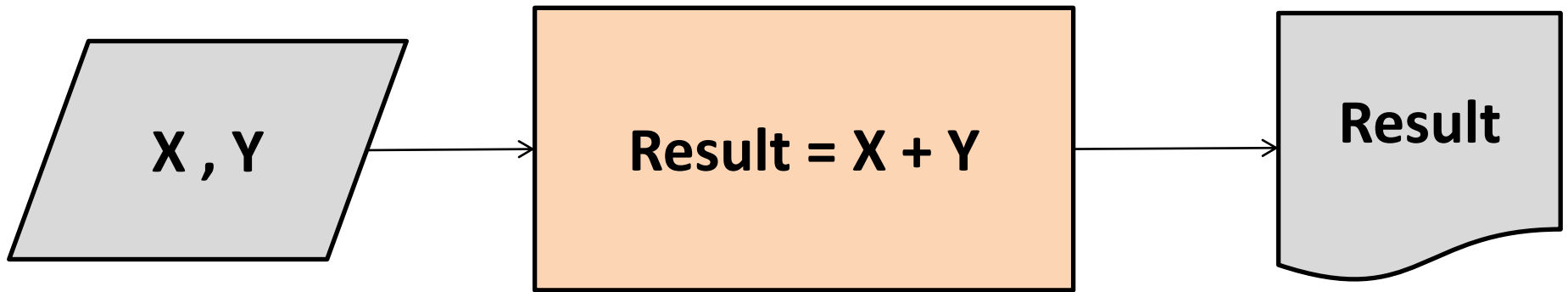
- d) Definir os dados de saída, ou seja, quais dados serão gerados depois do processamento
- e) Construir o algoritmo utilizando um dos tipos descritos na próxima seção
- f) Testar o algoritmo realizando simulações (chinesinho)



Construção de Algoritmos



Construção de Algoritmos



Tipos de Algoritmos

- ▶ Os tipos mais utilizados são:
 1. Descrição Narrativa
 2. Fluxograma
 3. Pseudocódigo (ou Portugol)



Descrição Narrativa

- ▶ Consiste em analisar o enunciado do problema e escrever, utilizando uma **linguagem natural** (por exemplo, português), os passos a serem seguidos para sua resolução
- ▶ **Vantagem:** não é necessário aprender nenhum conceito novo, pois uma linguagem natural, neste ponto, já é bem conhecida
- ▶ **Desvantagem:** a língua natural abre espaço para várias interpretações, o que posteriormente dificultará a transcrição desse algoritmo para programa



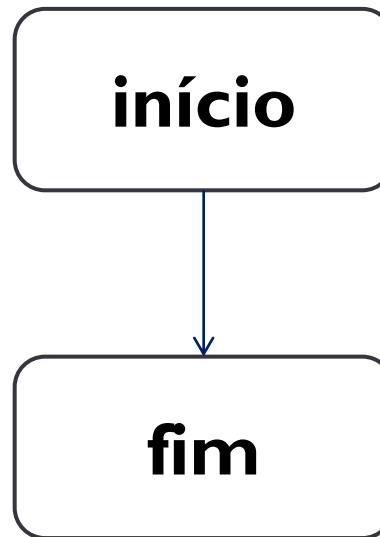
Fluxograma

- ▶ Consiste em analisar o enunciado do problema e escrever, utilizando **símbolos gráficos predefinidos**, os passos a serem seguidos para sua resolução
- ▶ **Vantagem:** o entendimento de elementos gráficos é mais simples que o entendimento de textos
- ▶ **Desvantagem:** é necessários aprender a simbologia dos fluxogramas e , além disso, o algoritmo resultante não apresenta muitos detalhes, dificultando sua transcrição para um programa






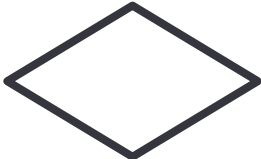


Fluxograma

- ▶ Fluxograma mínimo: não executa absolutamente nada, mas possui um **início** e um **fim**.



Fluxograma - Símbolos

	Símbolo utilizado para indicar o início e o fim do algoritmo
	Permite indicar o sentido do fluxo de dados. Serve exclusivamente para conectar os símbolos ou blocos existentes.
	Símbolo utilizado para indicar cálculos e atribuições de valores
	Símbolo utilizado para representar a entrada de dados
	Símbolo utilizado para representar a saída de dados
	Símbolo utilizado para indicar que deve ser tomada uma decisão, apontando a possibilidade de desvios



Pseudocódigo ou Portugol

- ▶ Consiste em analisar o enunciado do problema e escrever por **meio de regras predefinidas**, os passos a serem seguidos para sua resolução
- ▶ **Vantagem:** a passagem do algoritmo para qualquer linguagem de programação é quase imediata, bastando conhecer as palavras reservadas dessa linguagem que serão utilizadas
- ▶ **Desvantagem:** é necessário aprender as regras do pseudocódigo, que serão apresentadas nas próximas seções



Exemplo 1

Faça um algoritmo para mostrar o resultado da multiplicação de dois números



Exemplo 1

Descrição narrativa:

Passo 1 - Receber os dois números que serão multiplicados

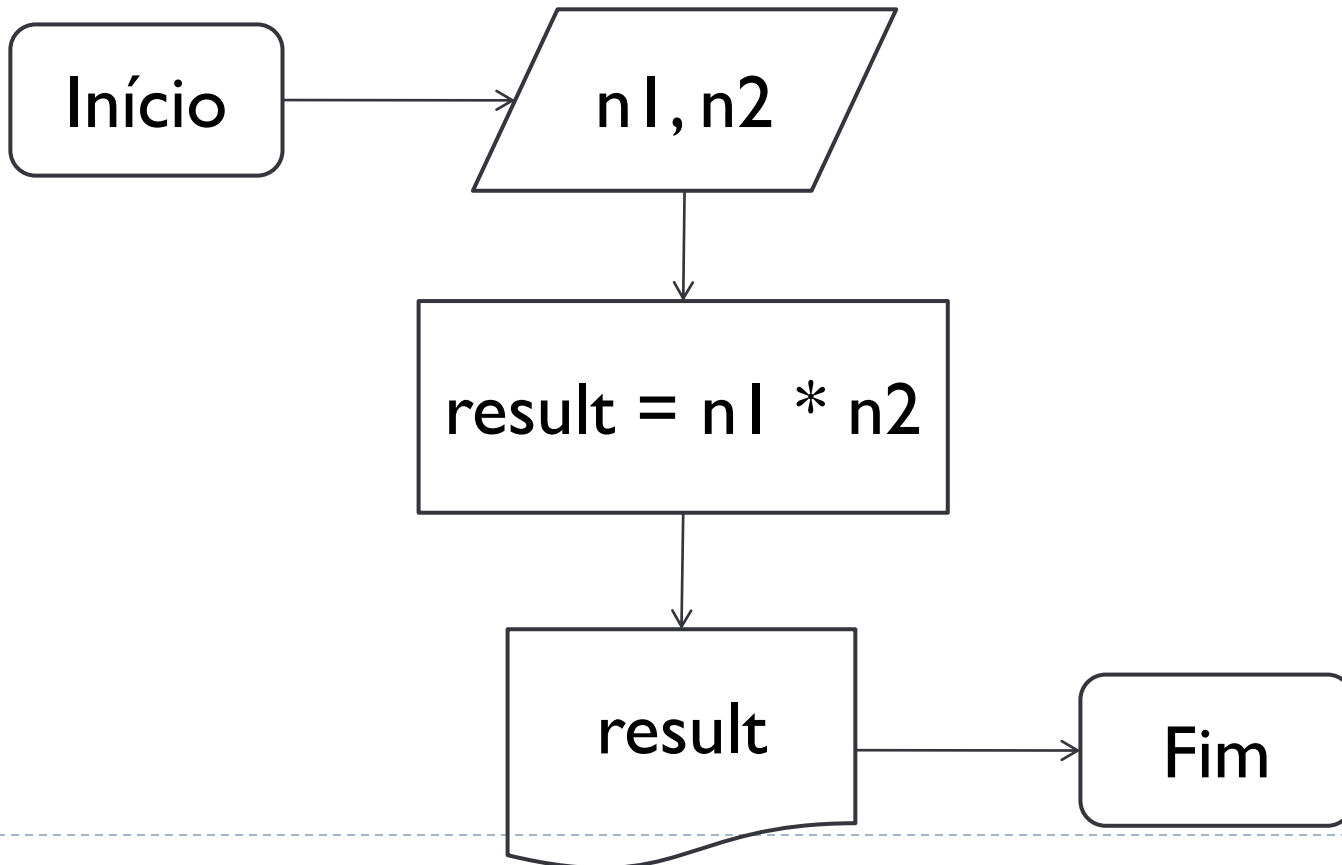
Passo 2 - Multiplicar os números

Passo 3 - Mostrar o resultado obtido na multiplicação



Exemplo 1

Fluxograma:



Exemplo 1

Pseudocódigo:

ALGORITMO

DECLARE n1, n2, result NUMÉRICO

ESCREVA “Digite um número”

LEIA n1

ESCREVA “Digite outro número”

LEIA n2

result \leftarrow n1 * n2

ESCREVA “Multiplicação = ”, result

FIM_ALGORITMO



Exemplo 2

Faça um algoritmo para mostrar o resultado da divisão de dois números



Exemplo 2

Descrição narrativa:

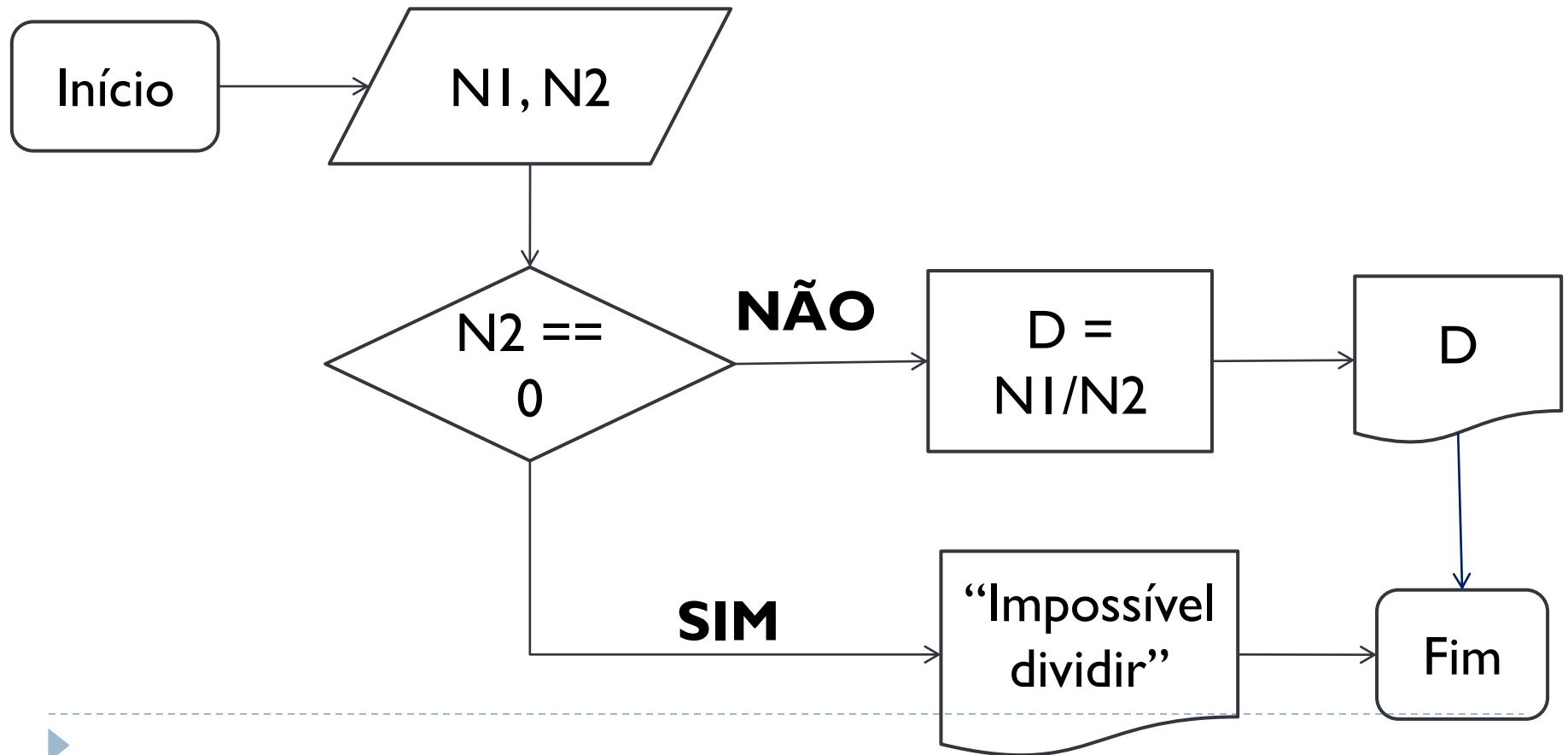
Passo 1 - Receber os dois números que serão divididos

Passo 2 – **Se** o segundo número for igual a zero, não poderá ser feita a divisão, pois não existe divisão por zero; **caso contrário**, dividir os números e mostrar o resultado da divisão



Exemplo 2

Fluxograma:



Exemplo 2

Pseudocódigo:

ALGORITMO

DECLARE N1, N2, D NUMÉRICO

ESCREVA “Digite dois números”

LEIA N1, N2

SE N2 == 0

ENTÃO ESCREVA “Impossível
dividir”

SENÃO INÍCIO

$D \leftarrow N1 / N2$

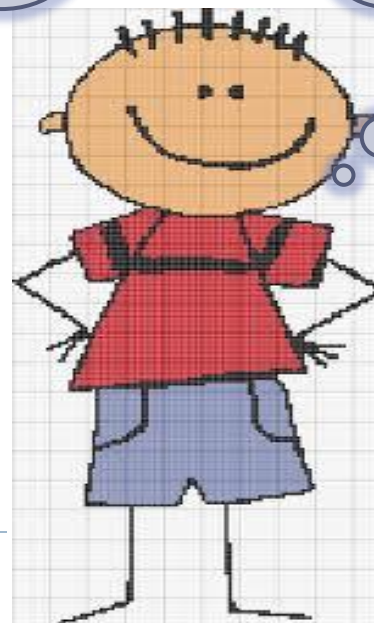
ESCREVA “Divisão = ”, D

FIM_ALGORITMO



Variável

- ▶ Duas pessoas estão conversando e desejam realizar uma conta



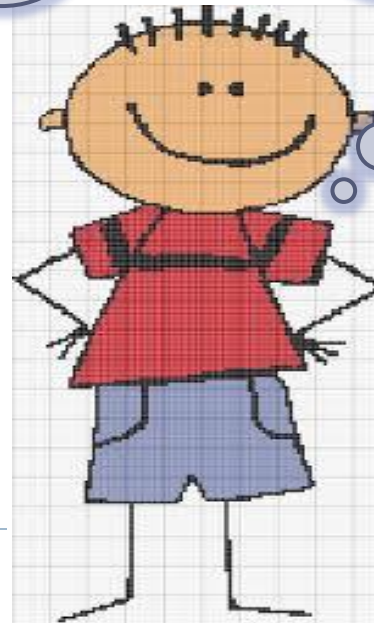
Variável

- ▶ Duas pessoas estão conversando e desejam realizar uma conta



Variável

- ▶ Duas pessoas estão conversando e desejam realizar uma conta



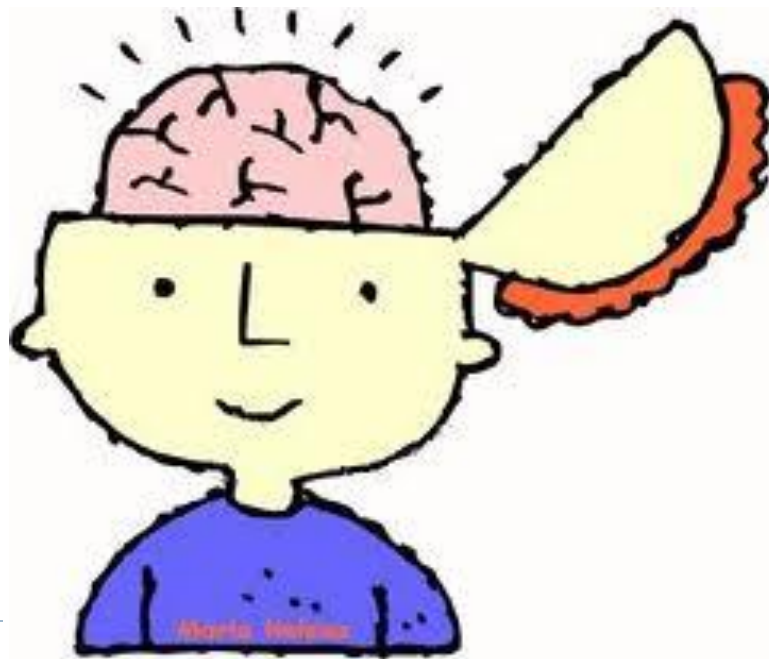
Variável

- ▶ Duas pessoas estão conversando e desejam realizar uma conta



Variável

- ▶ A segunda pessoa resgata os valores armazenados na **memória**, realiza a conta e responde dizendo que o resultado é 8



Variável

- ▶ Um programa recebe dados, que precisam ser armazenados no computador para serem utilizados no processamento
- ▶ Todos os computadores trabalham com sistema numérico binário (0 e 1)
- ▶ Cada dígito binário (0 e 1) ocupa porções de memória chamadas bytes (8 bits), e cada byte é identificado e acessado



Variável

- ▶ Desta maneira, uma **variável** representa uma posição de memória
- ▶ Possui nome e tipo e 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



Variável

Uma variável
representa uma
área na memória
onde se pode
*armazenar um
valor qualquer*

num1



Memória

4



Tipos de dados

► Os tipos mais utilizados são:

1. Numéricos
2. Lógicos
3. Literais ou caracteres



Tipos de dados - Numéricos

- ▶ Dividem-se em: *inteiros* e *reais*
- ▶ Os números **inteiros** podem ser positivos ou negativos e **não** possuem parte fracionária

Ex.:

-23

98

0

-357

237

-2



Tipos de dados - Numéricos

- ▶ Os números **reais** podem ser positivos ou negativos e **possuem** parte fracionária

Ex.:

23,45

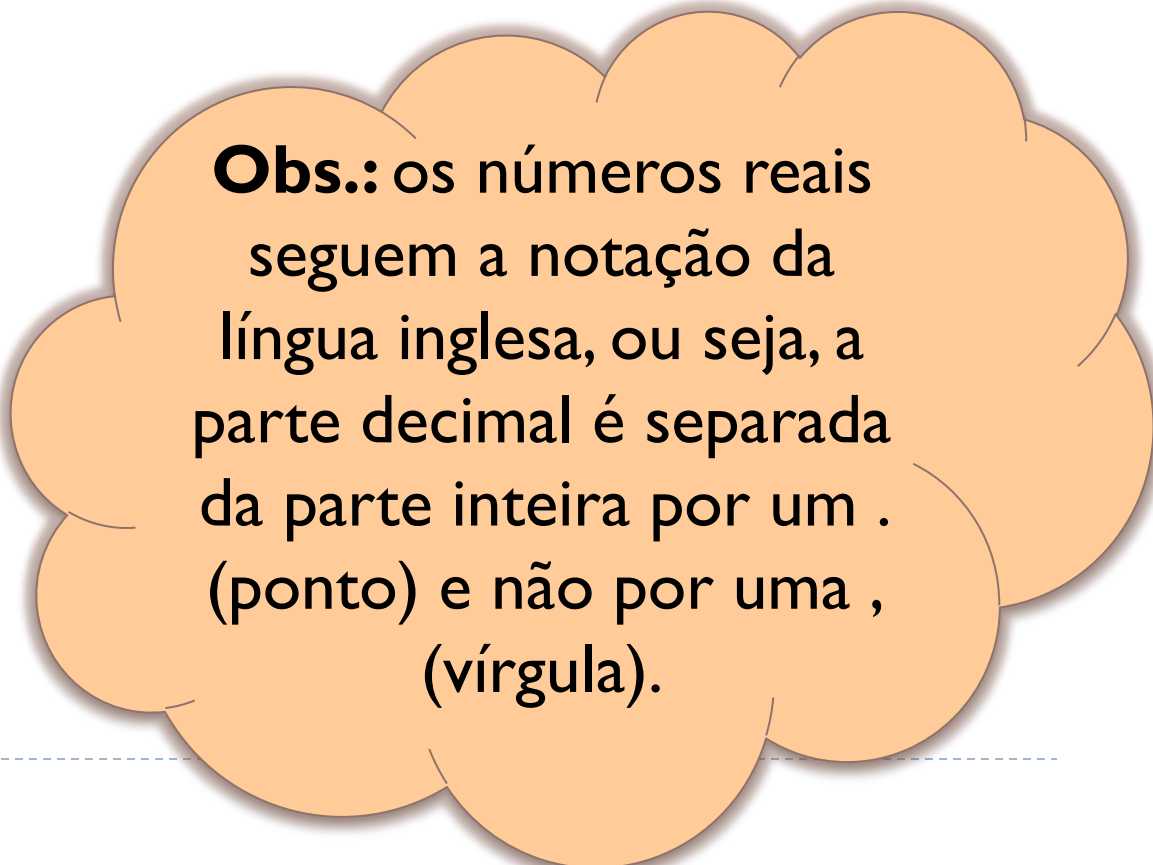
346,89

0,0

-357,0

237,88

-2,7



Obs.: os números reais seguem a notação da língua inglesa, ou seja, a parte decimal é separada da parte inteira por um . (ponto) e não por uma , (vírgula).

Tipos de dados - Lógicos

- ▶ São também chamados dados booleanos (por causa da álgebra de Boole)
- ▶ Podem assumir os valores *verdadeiro* ou *falso*



Tipos de dados – Caracteres

- ▶ Ou Literais, são dados formados por um único caractere ou por uma cadeia de caracteres
- ▶ Podem ser letras maiúsculas, minúsculas, números (**não** podem ser usados para cálculos) e os caracteres especiais (&, #, @, ?, +)
- ▶ Ex

‘aluno’

‘1234’

‘@ internet’



Formação de identificadores

- ▶ Identificadores são os **nomes** das **variáveis**, dos programas, das constantes, das rotinas, das unidades etc
- ▶ Os identificadores possuem regras básicas para sua formação



Formação de identificadores

1. Os caracteres que podem ser utilizados são: os números, as letras (maiúsculas e minúsculas) e o caractere sublinhado (underline)
2. O primeiro caractere deve ser sempre uma letra ou caractere sublinhado (underline)
3. Não são permitidos espaços em branco e caracteres especiais (@, \$, %, &, !, ?)
4. Não podem ser usadas palavras reservadas nos identificadores, ou seja palavras que pertençam a uma linguagem de programação



Exemplos de identificadores

► Identificadores válidos:

A

A2

Nota

NOTA

nota

XYZ

NOTA I

Nota_I

_Dia

Idade

Linguagem de programação C

- ▶ Antecedentes: UNIX, BCPL e B (Bell Labs – AT&T)
- ▶ Inventada e implementada por Dennis Ritchie
- ▶ Padrão **ANSI C** (*American Standard for Information System* - 1990) - criado para guiar implementações da linguagem C
- ▶ Todo programa em C é composto de uma ou mais funções. A Biblioteca C padrão contém funções mais comuns.
- ▶ A linguagem C++ é uma extensão da linguagem C (C++ = (C + orientação a objetos: POO))

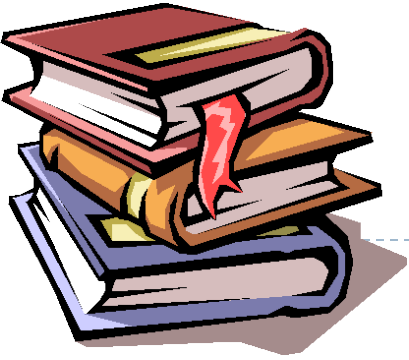




Exercícios

- 1- Faça um algoritmo para somar **dois números** e multiplicar o resultado pelo primeiro número.
- 2- Faça um algoritmo para calcular o novo salário do funcionário. Considere que o funcionário terá **10% de aumento**.





Referência Bibliográfica

- ▶ ASCENCIO, Ana Fernanda Gomes e CAMPOS, Edilene A. Veneruchi. **Fundamentos da Programação de Computadores – Algoritmos, Pascal e C/C++**. São Paulo: Pearson Prentice Hall, 2007. 2ª Edição. Capítulo I.
- ▶ SOUZA, A. Furlan; GOMES, Marcelo Marques; SOARES, Marcio Vieira e CONCILIO, Ricardo. **Algoritmos e Lógica de Programação**. 2ª ed. Ver. e ampl. São Paulo: Cengage Learning, 2011.
- ▶ MSDN Microsoft: <https://msdn.microsoft.com/pt-br/library/kx37x362.aspx>

