

Algoritmos e Linguagens de Programação

**Introdução à Ciência da Computação
ICC0001**

Prof. Diego Buchinger

Conceitos Básicos

- Para dizer o que um computador deve fazer devemos programá-lo através de instruções específicas!!
- Ada Lovelace – a primeira programadora
- Computadores são construídos para executar tarefas bem determinadas a partir de um conjunto limitado de instruções.
 - E.g.: somar, subtrair, realizar decisões lógicas (x é menor que y?)

Conceitos Básicos

- Inicialmente cada máquina “entendia” um conjunto de instruções próprias
 - Instruções de uma máquina não eram exatamente iguais as instruções de outra máquina.
 - Dificuldade em programar diferentes máquinas

- Para um ser humano, a linguagem de máquina é de difícil compreensão.

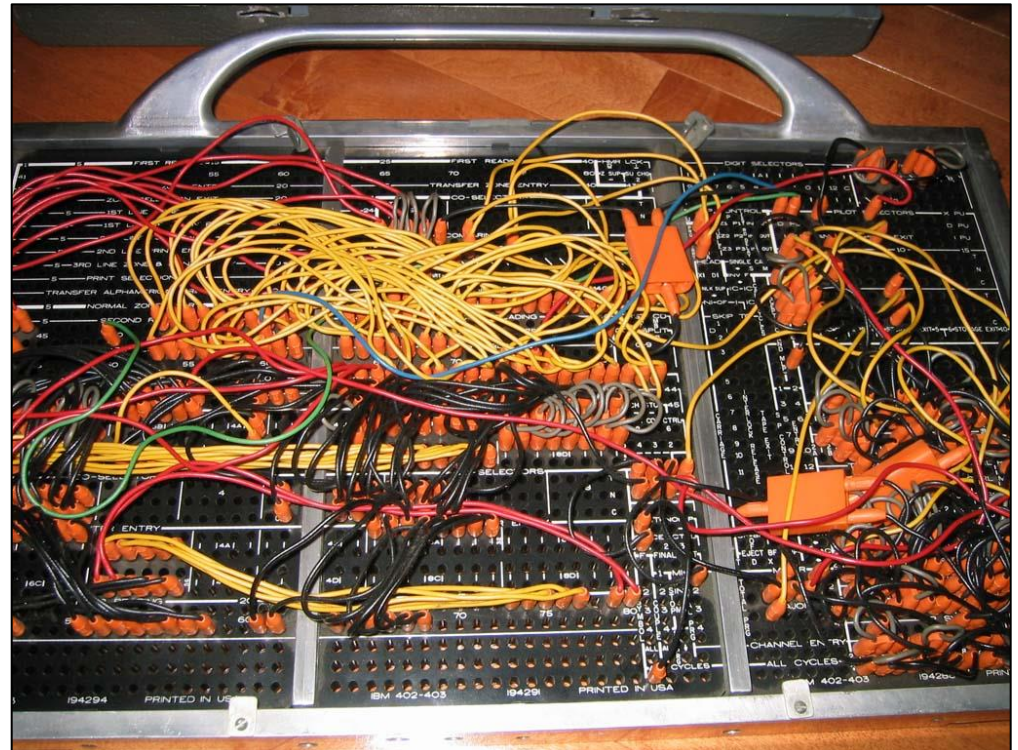
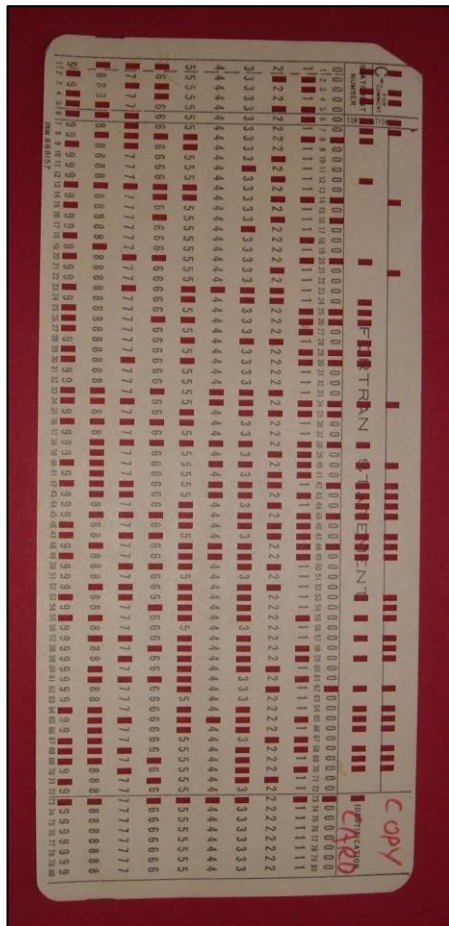
00000000 00101000 00001110

01000100 00011110 11100111



Conceitos Básicos

- Primeira Geração de Linguagens



Conceitos Básicos

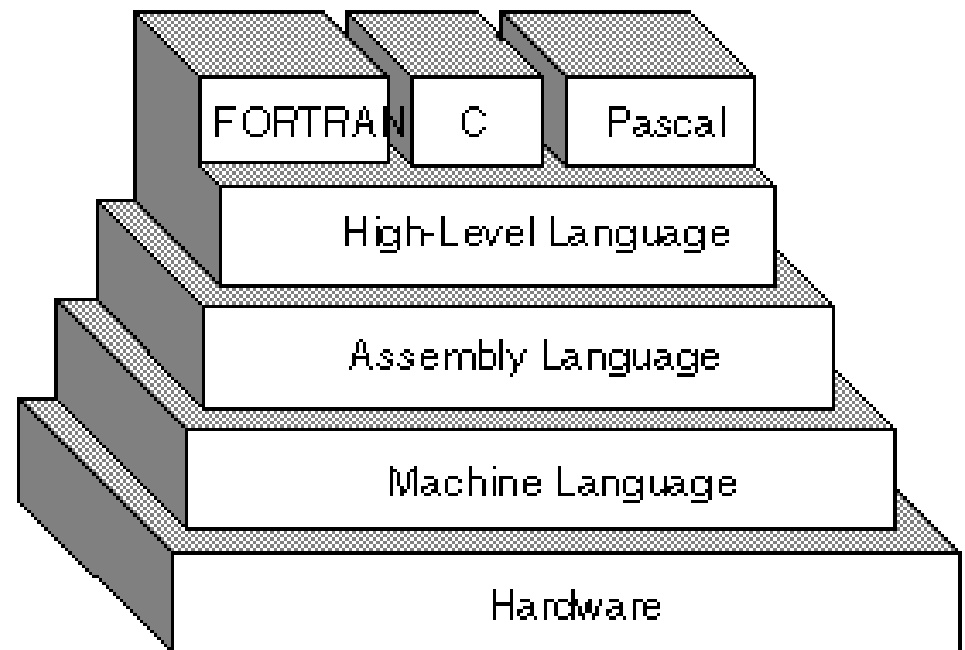
- Segunda Geração de Linguagens (baixo nível)
 - Uso de linguagem escrita
 - Uso de mnemônicos – mais fácil de memorizar e utilizar
 - Uso de um “montador” *assembler* para tradução

Linguagem Assembly	Linguagem de Máquina
ADD \$t1, t2, \$t3	04CB: 0000 0100 1100 1011
OR \$t3, \$t1, \$t2	029A: 0000 0010 1001 1010

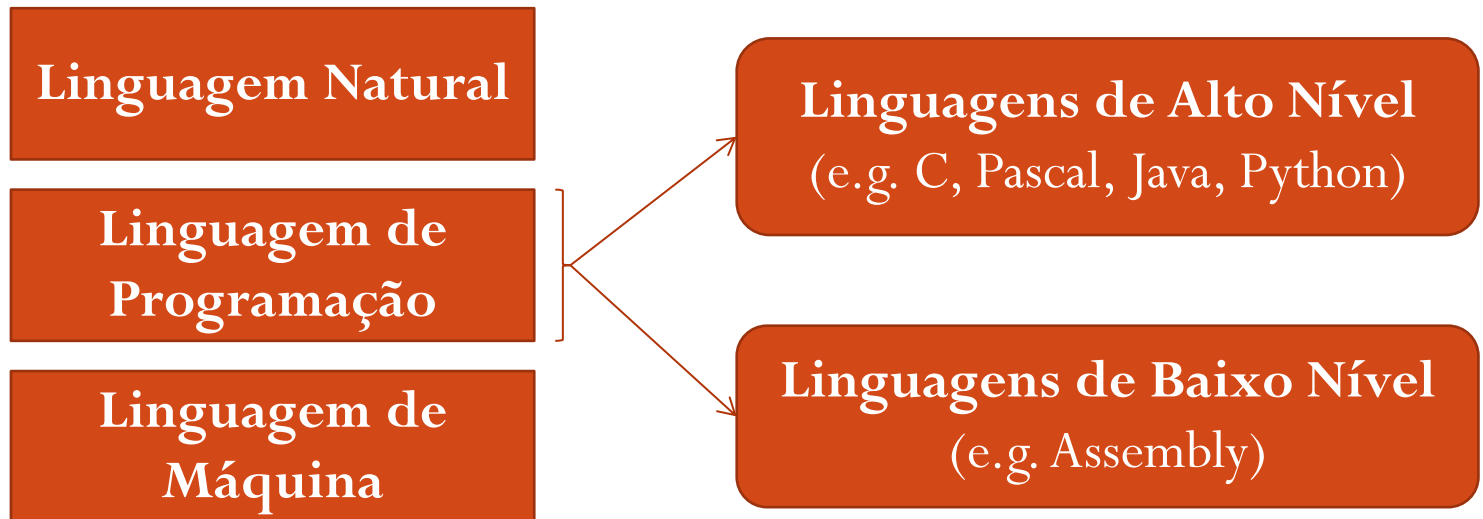
- Cada unidade de processamento utiliza sua própria linguagem (famílias: x86, x64, ARM / marcas: intel / AT&T)

Conceitos Básicos

- Terceira Geração de Linguagens (alto nível)
 - Linguagens projetadas para ser entendidas mais facilmente pelos humanos
 - Expressões similares a matemática e álgebra
 - Uso de nomes para variáveis
 - O “texto” pode ser convertido para uma linguagem de montagem

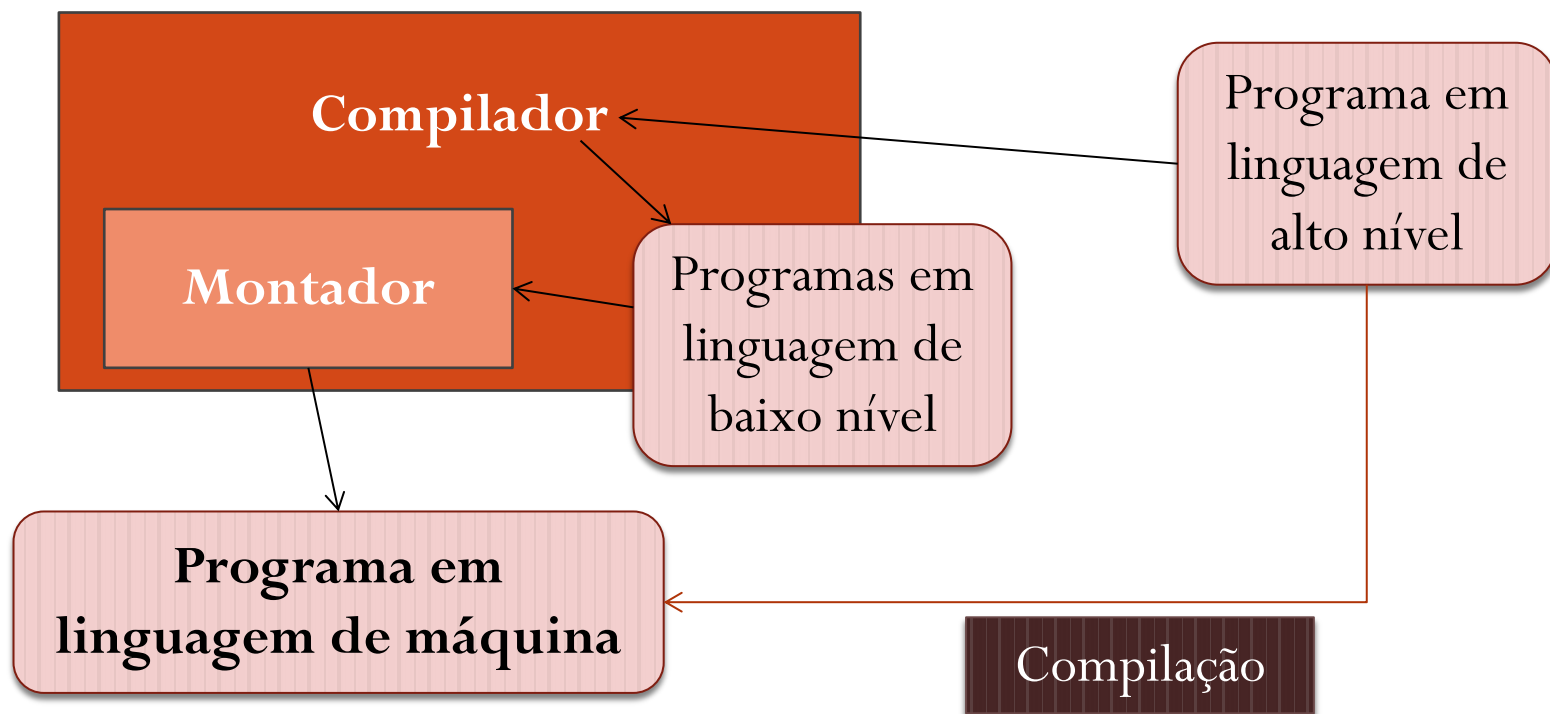


Conceitos Básicos



Conversão entre Linguagens

- **Compiladores**: tradutores que mapeiam programas escritos em linguagem de alto nível para programas equivalentes em linguagem de baixo nível.
- **Montadores**: tradutores de linguagem de baixo nível em linguagem de máquina para um dado computador em particular.



Conceitos Básicos

- **Software** ou **Programas** são escritos de forma que o *hardware* do computador possa entender o que deve ser feito
 - Descrever a ideia / lógica / passo-a-passo de como processar a entrada para se chegar a saída (**Algoritmo - Código**);

Algoritmo

- “Algoritmo é uma sequência de passos que visa atingir um objetivo bem definido” (FORBELLONE, 1999)
- “Algoritmo é a descrição de uma sequência de passos que deve ser seguida para a realização de uma tarefa” (ASCENCIO, 1999)
- “Algoritmo é uma sequência finita de instruções ou operações cuja execução (...) resolve um problema computacional, qualquer que seja sua instância” (SALVETTI, 1999)

Algoritmo

- Somar três números:
 1. Receber três números;
 2. Somar os três números;
 3. Mostrar o resultado obtido.

Algoritmo

- Fazer um sanduíche:
 1. Pegar pão;
 2. Cortar o pão ao meio;
 3. Pegar a maionese;
 4. Passar a Maionese no pão;
 5. Pegar e cortar alface e tomate;
 6. Colocar os pedaços de alface e tomate no pão;
 7. Pegar o hambúrguer;
 8. Fritar o hambúrguer;
 9. Colocar o hambúrguer no pão;

Algoritmo

- Trocar uma lâmpada
- Sacar dinheiro no banco
- Calcular a média de dois números

(Podem existir mais de um algoritmo para um mesmo problema)

- Para se elaborar um algoritmo é necessário **entender o problema** que se deseja resolver, definir os **dados de entrada**, o **processamento** e os **dados de saída**.

Tipos de Algoritmos

- **Descrição narrativa**: escrever os passos em linguagem natural (e.g. português)
 - [+] não é necessário aprender uma nova simbologia;
 - [-] abre espaço para várias interpretações;
- **Fluxograma**: utilização de símbolos e escrita
 - [+] menos escrita, e menor margem de múltiplas interpretações
 - [-] aprender a simbologia e apresenta poucos detalhes

ENTRADA DE DADOS

$N1, N2$

CÁLCULOS E ATRIBUIÇÕES

$M = (N1 + N2)$

TOMADA DE DECISÃO

$M > 0$

SAÍDA DE DADOS

M

Tipos de Algoritmos

- Fluxograma – Principais blocos de construção:

INDICA INÍCIO DO
ALGORITMO

INÍCIO

INDICA FIM DO
ALGORITMO

FIM

ENTRADA DE DADOS

N1, N2

CÁLCULOS E ATRIBUIÇÕES

$M = (N1 + N2)$

TOMADA DE DECISÃO

$M > 0$

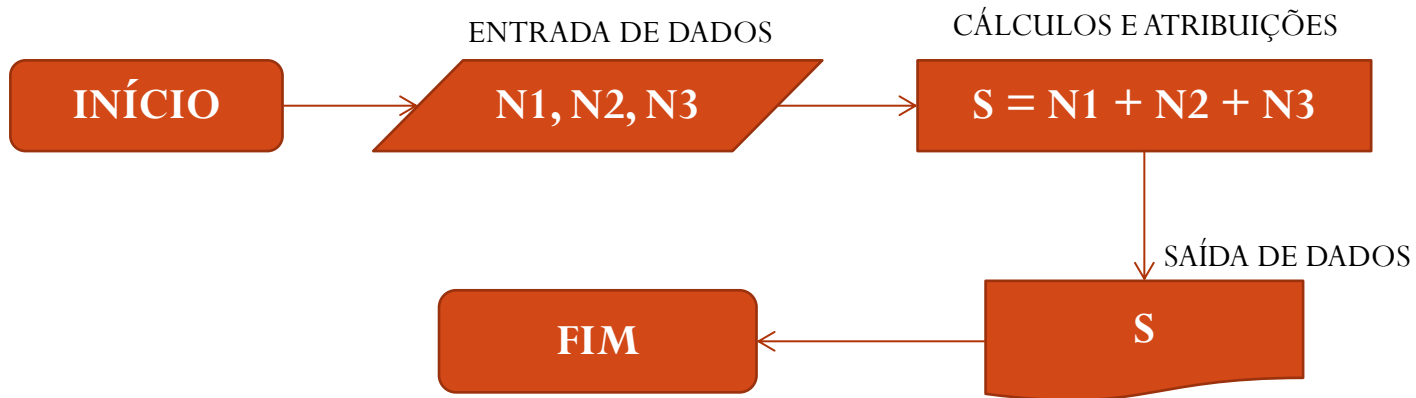
SAÍDA DE DADOS

M

Tipos de Algoritmos

- **Fluxograma:**

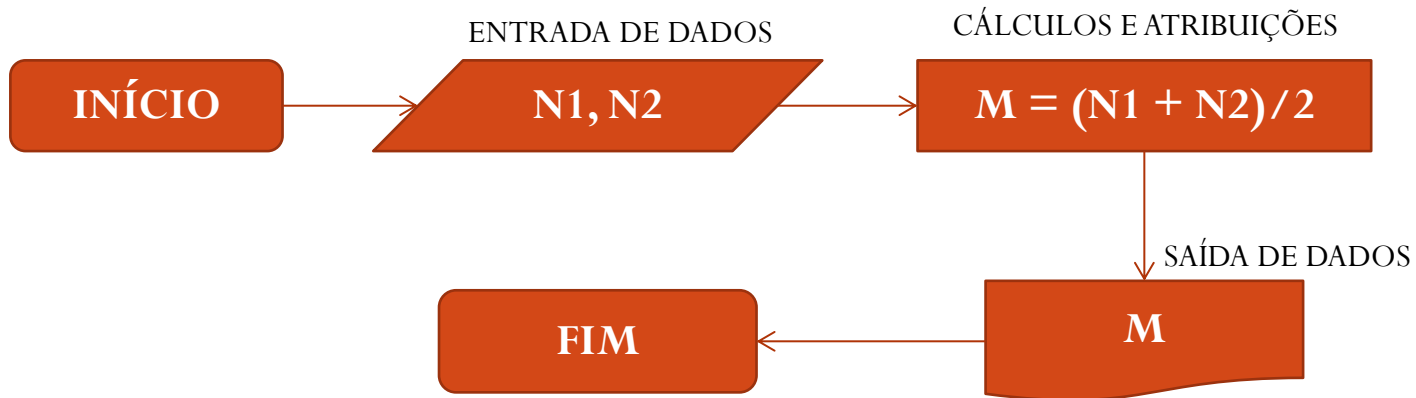
- E.g.: Somar três números e mostrar o resultado



Tipos de Algoritmos

- Fluxograma:

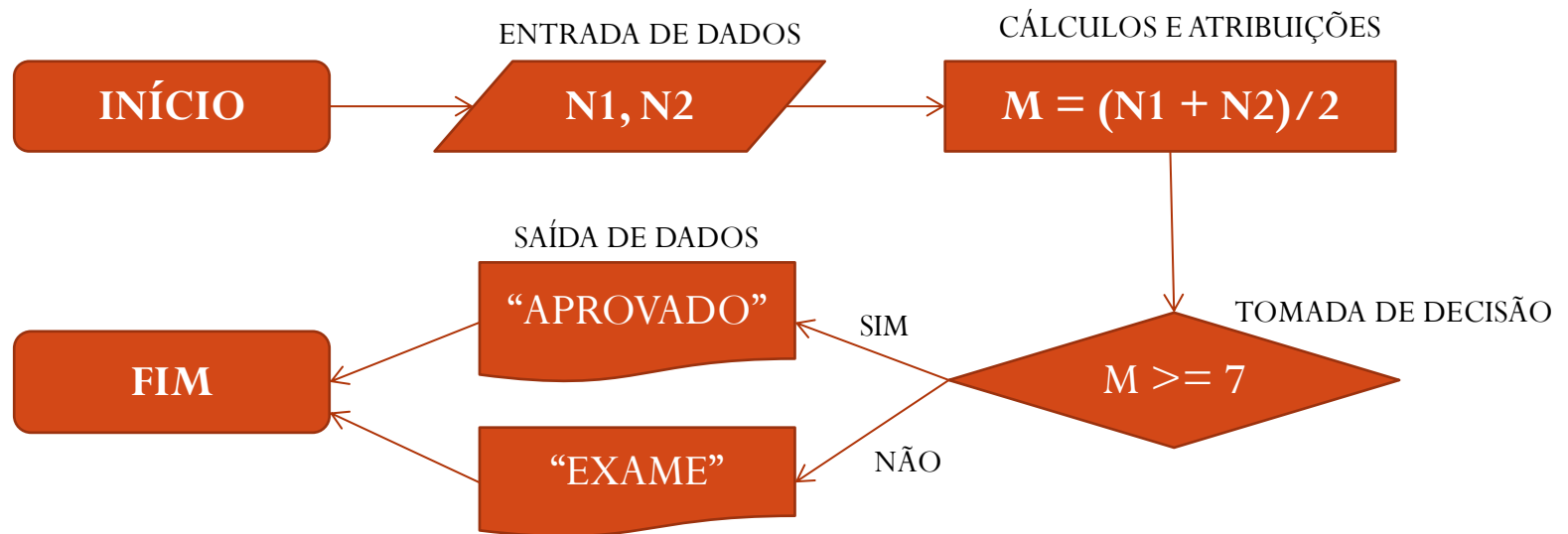
- E.g.: Calcular a média entre dois valores



Tipos de Algoritmos

- Fluxograma:

- E.g.: Calcular a média de um aluno com duas notas e dizer se ele está aprovado ou não

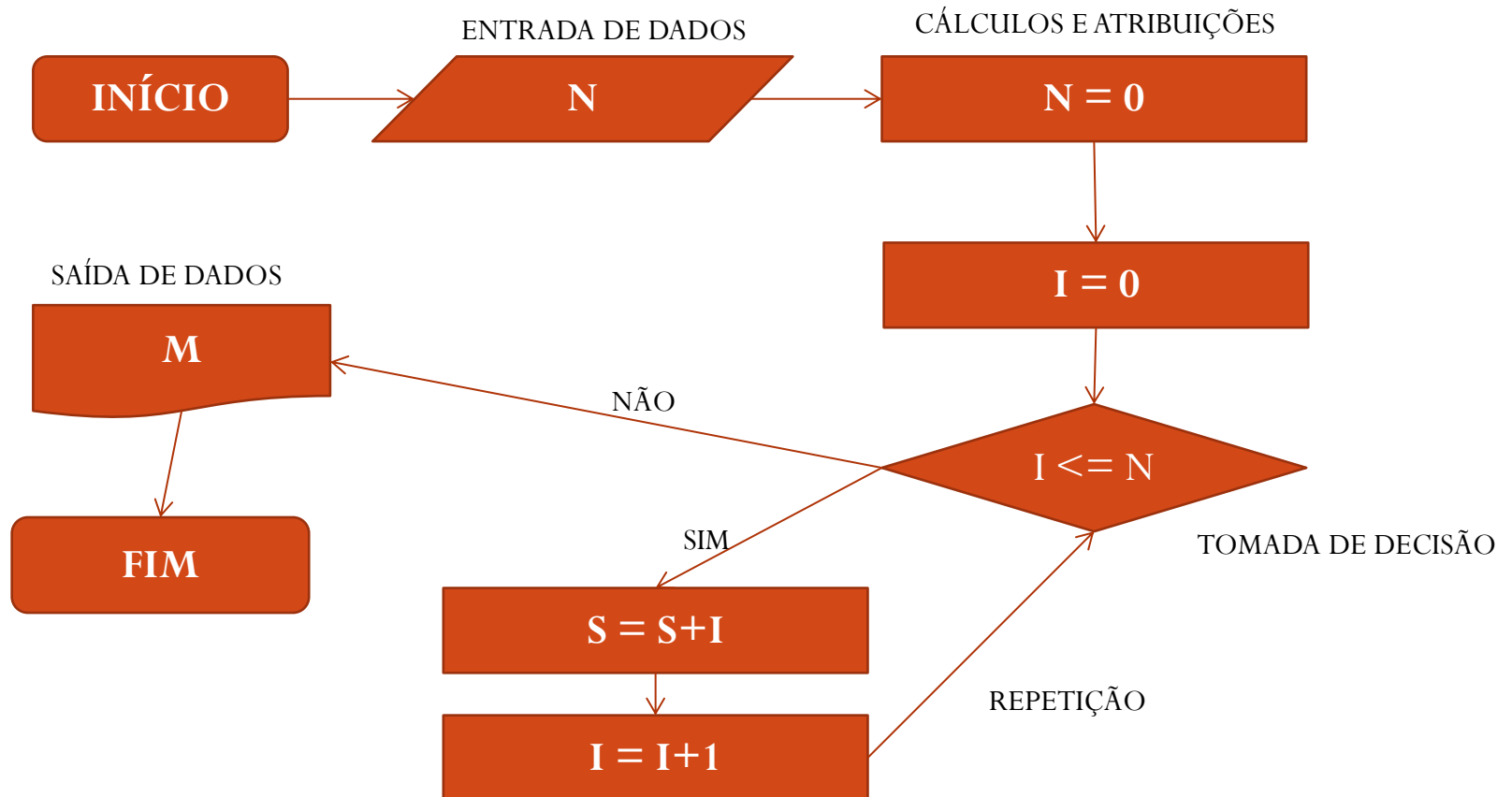


OBS: Quando queremos mostrar uma palavra a colocamos entre aspas!

Tipos de Algoritmos

- Fluxograma:

- E.g.: Calcular um somatório



Exercício

Escreva os seguintes algoritmos em fluxograma:

- Receber um número, aumentá-lo em dois, e mostrar o valor resultante.
- Receber quatro números que representam notas de um aluno durante o ano letivo e calcular a sua média final.
- Como seria um algoritmo que recebe o número de faltas de um aluno durante o ano letivo e determina se ele reprovou por frequência (dez ou mais faltas) ou não.
- E se fossem informados quatro valores, cada um indicando o número de faltas por bimestre?

Tipos de Algoritmos

- **Pseudocódigo ou portugol**: escrita por regras pré-definidas
 - [+] parecido com o fluxograma
 - [+] similar a uma linguagem que o computador entende
 - [-] aprender as regras de pseudocódigo
- E.g.: Somar três números e mostrar o resultado

ALGORITMO	<i>(informa o início)</i>
DECLARE N1,N2,N3,S	<i>(ingredientes)</i>
LEIA N1,N2,N3	<i>(entrada de dados)</i>
S = N1 + N2 + N3	<i>(cálculo e atribuição)</i>
ESCREVA S	<i>(saída de dados)</i>
FIM_ALGORITMO	<i>(informa o fim)</i>

OBS: Note que os desenhos do fluxograma foram trocados por palavras-chaves

Tipos de Algoritmos

- Pseudocódigo ou portugol:
 - E.g.: Calcular a média entre dois valores

```
ALGORITMO
  DECLARE N1, N2, M
  LEIA N1, N2
  M = (N1 + N2) / 2
  ESCREVA M
FIM_ALGORITMO
```

Tipos de Algoritmos

- Pseudocódigo ou portugol:

- E.g.: Calcular a média de um aluno com duas notas e dizer se ele está aprovado ou não

ALGORITMO

DECLARE N1, N2, M

LEIA N1, N2

$M = (N1 + N2) / 2$

SE $M \geq 7$

(tomada de decisão)

 ESCREVA “APROVADO”

(se $M \geq 7$)

SENÃO

 ESCREVA “EXAME”

(se $M < 7$)

FIM_ALGORITMO

Tipos de Algoritmos

- Pseudocódigo ou portugol:

- E.g.: Calcular um somatório

ALGORITMO

DECLARE S, N, I

LEIA N

S = 0

I = 0

ENQUANTO I <= N (REPETIÇÃO)

 S = S + I

 I = I+1

ESCREVA S

FIM_ALGORITMO

Exercício

Escreva os seguintes algoritmos em pseudocódigo:

- Receber um número, aumentá-lo em dois, e mostrar o valor resultante.
- Receber quatro números que representam notas de um aluno durante o ano letivo e calcular a sua média final.
- Como seria um algoritmo que recebe o número de faltas de um aluno durante o ano letivo e determina se ele reprovou por frequência (dez ou mais faltas) ou não.
- E se fossem informados quatro valores, cada um indicando o número de faltas por bimestre?

Interessante

Ferramenta de auxílio no estudo de programação:

VisualAlg

Parecido com uma linguagem em alto nível mas
escrito em português e com auxílio visual de execução

Linguagem de Programação Alto Nível

Vamos conhecer duas linguagens de programação de alto nível e escrever alguns exemplos...

Linguagem C

Linguagem JAVA

Game Based Learning

Existem diversas páginas na internet que oferecem ferramentas de programação, tutoria e ensino

<https://studio.code.org/s/mc>

<https://www.robomindacademy.com/go/navigator/courses>

<https://blockly-games.appspot.com/?lang=en>

http://br.codecombat.com/play?hour_of_code=true

Algoritmos e Linguagens de Programação

**Introdução à Ciência da Computação
ICC0001**

Prof. Diego Buchinger