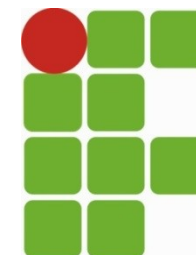


Abordagem Contextual

Formas de Representação de Algoritmos

Prof. Luis Claudio Gubert
luis.gubert@ibiruba.ifrs.edu.br





Abordagem Contextual

- Algoritmo não é a solução do problema, e sim, o caminho para a solução de um problema.
- Existem várias formas de representar um algoritmo.
- O aprendizado de algoritmos não se consegue a não ser através de muitos exercícios.
- Para a definição de um bom algoritmo é necessário desenvolver um raciocínio lógico.



Abordagem Contextual

- Algoritmos não se aprendem:
 - Copiando algoritmos
 - Estudando algoritmos
- Algoritmos só se aprendem:
 - Construindo algoritmos
 - Testando algoritmos

Abordagem Contextual – Exemplo 1



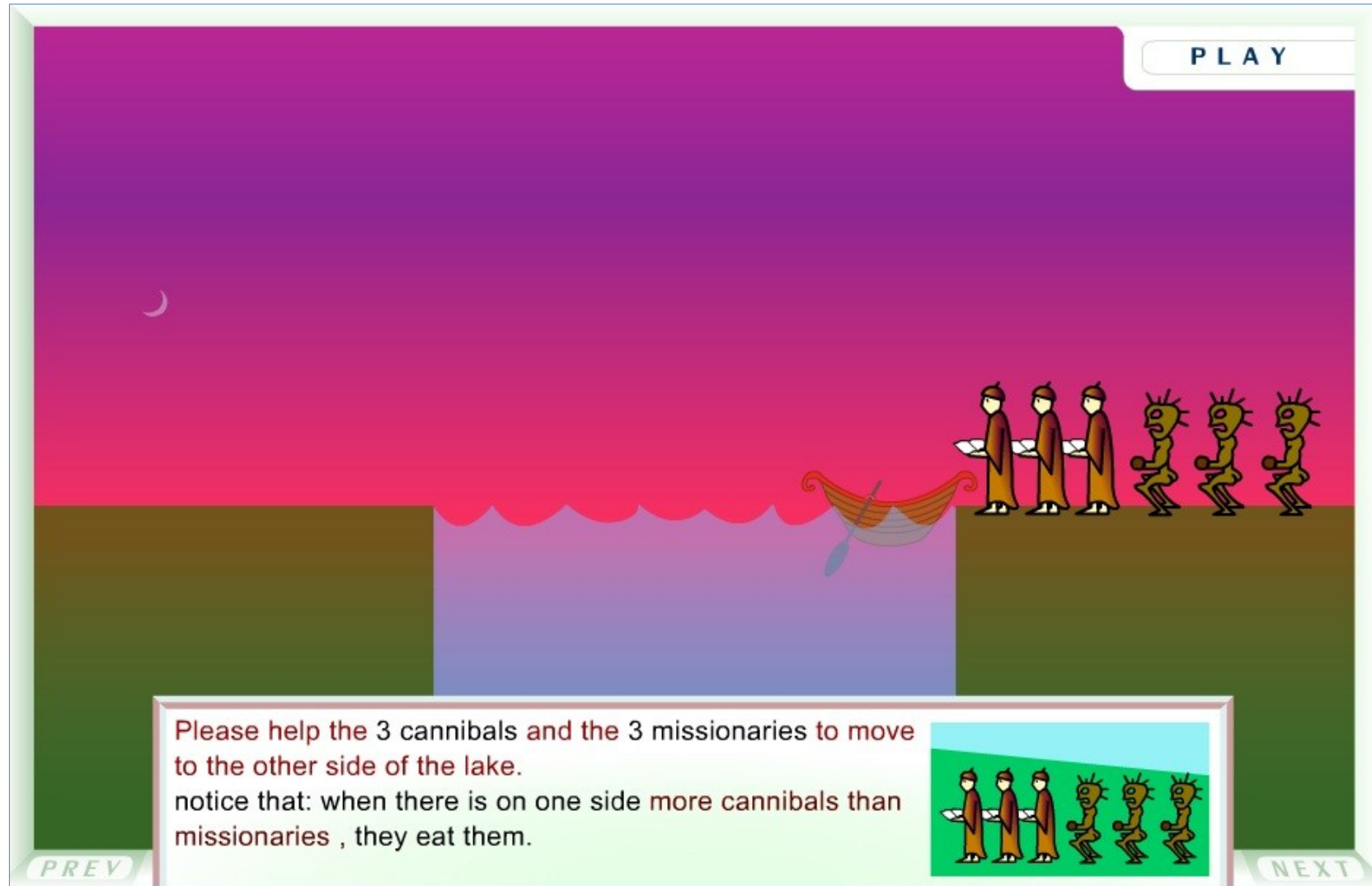
PLAY

please help the man in the boat to move - the wolf , the sheep and the box of cabbage to the other side of the lake.
notice that:
wolves eat sheep & sheep eat cabbage when no man around.



NEXT

Abordagem Contextual – Exemplo 2



PLAY

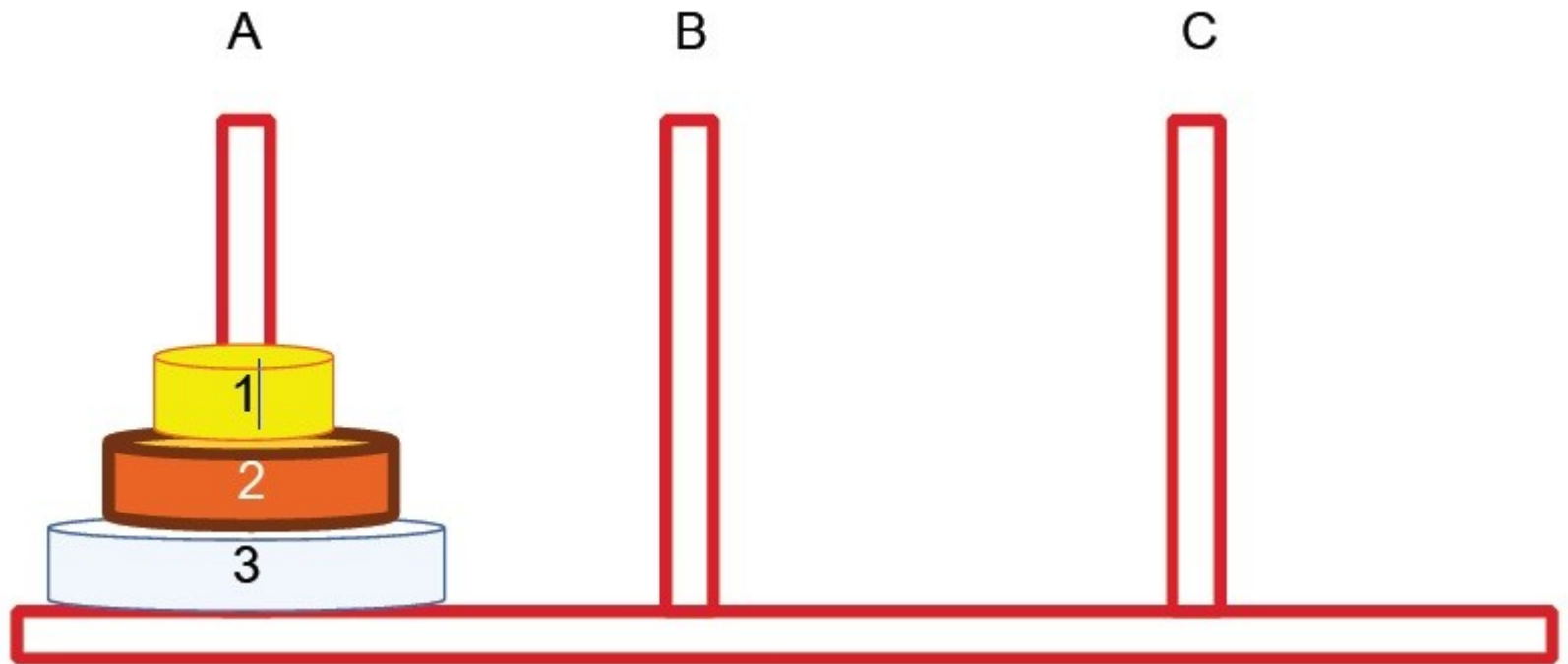
Please help the 3 cannibals and the 3 missionaries to move to the other side of the lake.
notice that: when there is on one side more cannibals than missionaries , they eat them.

PREV

NEXT

Abordagem Contextual - Torre de Hanói

- Transferir um disco para outra haste. Não pode colocar um disco maior em cima de um menor





Conceito de algoritmo

- A automação é o processo em que uma tarefa deixa de ser desempenhada pelo homem e passa a ser realizada por máquinas
 - Dispositivos mecânicos ➡ máquinas industriais
 - Eletrônicos ➡ computadores
 - Natureza mista ➡ robôs
- É necessário que seja especificado com clareza e exatidão o que deve ser realizado em cada uma das fases do processo a ser automatizado, bem como a sequência em que estas fases devem ser realizadas.
- À especificação da sequência ordenada de passos que deve ser seguida para a realização de uma tarefa, dá-se o nome de **algoritmo**.



Conceito de algoritmo

“Algoritmo é um conjunto finito de regras, bem definidas, para a solução de um problema em um tempo finito e com um número finito de passos.”



Conceito de algoritmo

- Para que um computador possa desempenhar uma tarefa é necessário que esta seja detalhada **passo a passo**, numa forma compreensível pela máquina, utilizando aquilo que se chama de programa.
- Neste sentido, um programa de computador nada mais é que um **algoritmo** escrito numa forma **compreensível** pelo computador.



Para se ter um algoritmo, é necessário:

1. Que se tenha um número finito de passos;
2. Que cada passo esteja precisamente definido, sem possíveis ambiguidades;
3. Que existam zero ou mais entradas tomadas de conjuntos bem definidos;
4. Que existam uma ou mais saídas;
5. Que exista uma condição de fim sempre atingida para quaisquer entradas e num tempo finito.

Importância

“Quanto mais tempo se leva na construção de um algoritmo do papel, menos tempo leva-se codificando-o”.





Formas de representação de um algoritmo

- Dentre as formas de representação de algoritmos mais conhecidas, sobressaltam:
 - a **Descrição Narrativa**
 - o **Fluxograma Convencional**
 - o **Diagrama de Chapin**
 - o **Pseudocódigo**, também conhecido como **Linguagem Estruturada** ou **Portugol**.



Descrição Narrativa

- Nesta forma de representação os algoritmos são expressos diretamente em linguagem natural.
Como por exemplo:
 - Troca de um pneu furado:
 - Afrouxar ligeiramente as porcas
 - Suspender o carro
 - Retirar as porcas e o pneu
 - Colocar o pneu reserva
 - Apertar as porcas
 - Abaixar o carro
 - Dar o aperto final nas porcas



Descrição Narrativa

- Cálculo da média de um aluno:
 - Obter as notas da primeira e da segunda prova
 - Calcular a média aritmética entre as duas
 - Se a média for maior ou igual a 7, o aluno foi aprovado, senão ele foi reprovado



Descrição Narrativa

➤ Sacar Dinheiro

- ✓ Ir até o caixa eletrônico
- ✓ Colocar o cartão
- ✓ Digitar a senha
- ✓ Solicitar o saldo
- ✓ Se o saldo for maior ou igual à quantia desejada, sacar a quantia desejada; caso contrário sacar o valor do saldo
- ✓ Retirar dinheiro e cartão
- ✓ Sair do caixa eletrônico

Fluxograma Convencional e Diagrama de Blocos

- É uma representação gráfica de algoritmos onde formas geométricas diferentes implicam ações (instruções, comandos) distintos. Tal propriedade **facilita o entendimento das idéias** contidas nos algoritmos.
- Símbolos utilizados no fluxograma:



= Início e final do fluxograma



= Operação de entrada de dados

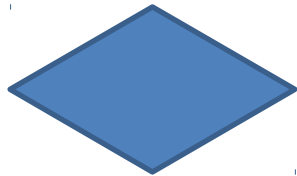
Fluxograma Convencional e Diagrama de Blocos



= Operação de saída de dados



= Operações de atribuição e chamada ou retorno de subalgoritmo



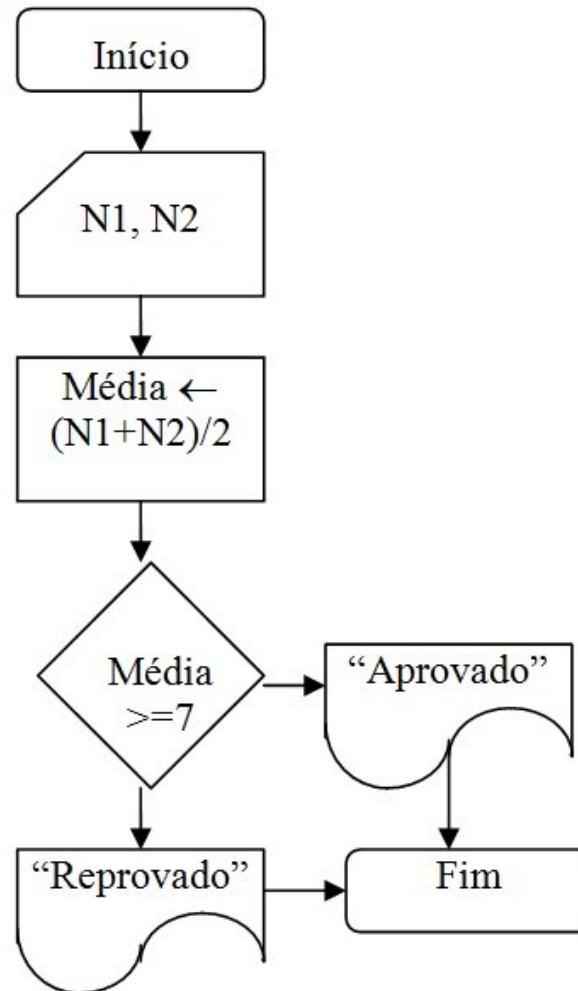
= Decisão



= Seta do Fluxo de Dados

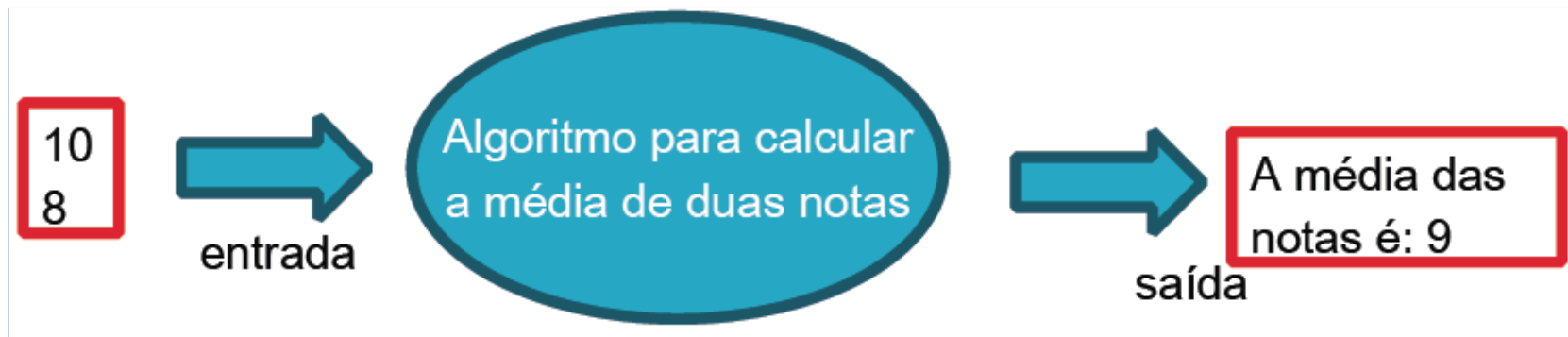
Fluxograma Convencional e Diagrama de Blocos

Exemplo



Saída

- Um algoritmo é desenvolvido para que uma determinada tarefa seja realizada
- Pode ser necessário apresentar ao usuário um resultado



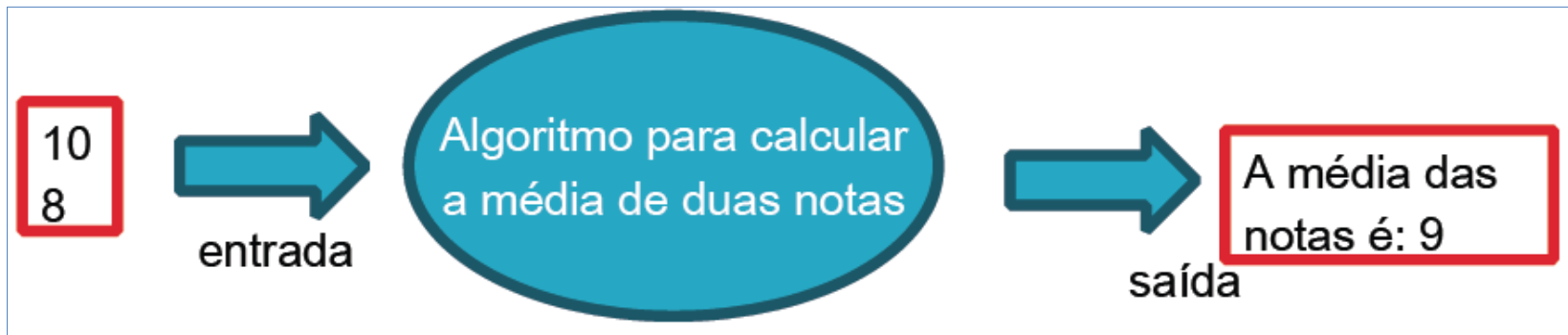
Saída

- Qual o dispositivo de saída padrão?
 - Monitor



Entrada

- O algoritmo pode precisar de **dados** para sua execução!



Entrada

- Qual o dispositivo padrão de entrada?
 - Teclado!



Exemplo Entrada-Saída

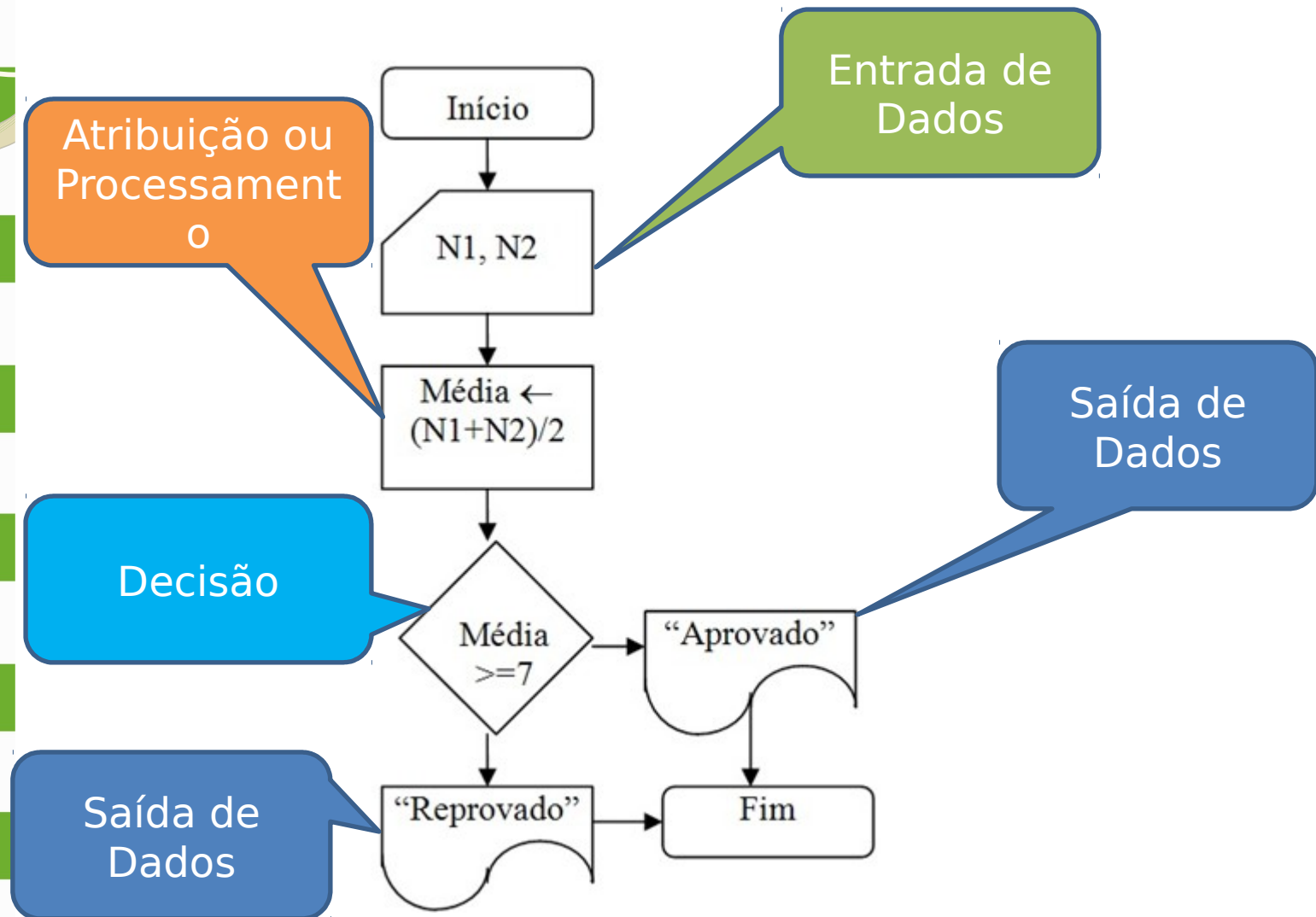
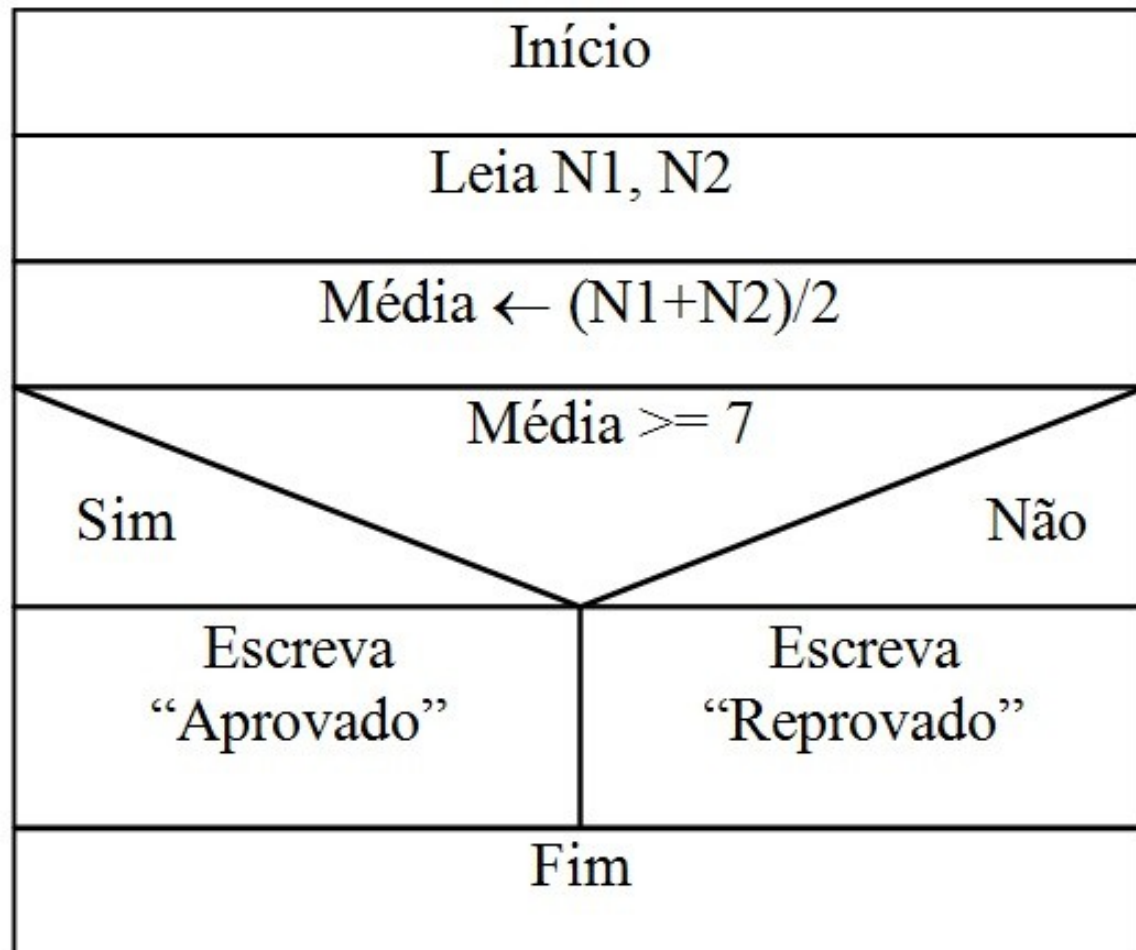


Diagrama de Chapin



- Criador: Ned Chapin
- Substituição do fluxograma tradicional por um diagrama que apresenta uma visão **hierárquica e estruturada** da lógica do programa.
- **Vantagem:** representação das estruturas que tem um ponto de entrada e um ponto de saída e são compostas pelas estruturas básicas de controle de sequência, seleção e repartição.

Diagrama de Chapin – cálculo da média de um aluno





Pseudocódigo

- Conhecido como português estruturado ou português
- Rico em detalhes
- Esta representação é suficientemente geral para permitir que a tradução de um algoritmo nela representado para uma linguagem de programação específica seja praticamente direta



Representação de um algoritmo na forma de pseudocódigo

Algoritmo *<nome_do_algoritmo>*
 <declaração_de_variáveis>
 <subalgoritmos>

Início
 <corpo_do_algoritmo>

Fim.

Representação do algoritmo de cálculo da média

Algoritmo Media

Var N1, N2, Media

Inicio

Leia N1, N2

 Media := (N1+N2)/2

Se Media >= 7 **Entao**

Escreva "Aprovado"

Senao

Escreva "Reprovado"

Fim.



Operadores de Atribuição

- Um operador de atribuição serve para **atribuir um valor** a uma variável.
- Em algoritmo usamos o operador de atribuição:
 - =
 - <-
- A sintaxe de um comando de atribuição é:
 - ***nome_da_variável = expressão***
- A expressão localizada no lado **direito** do sinal de igual é avaliada e armazenado o valor resultante na variável à **esquerda**.
 - O nome da variável aparece sempre sozinho, no lado esquerdo do sinal de igual deste comando.



Operadores de Atribuição

- O **comando de atribuição** ou simplesmente **atribuição**, é a principal maneira de armazenar uma informação numa variável.

Ex:

```
nome = "Jenoveva";
```

```
preco = 15.85;
```

```
quantidade = 5;
```

```
total = preco * quantidade;
```

```
imposto = total * 17 / 100;
```



Operadores Aritméticos

Operador	Operação	Prioridade
+	Adição	2
-	Subtração	2
*	Multiplicação	1
/	Divisão	1
%	Resto da Divisão	1

Operadores Relacionais

- Os operadores relacionais são operadores binários que devolvem os valores lógicos verdadeiro e falso.

Operador	Comparação
>	maior que
<	menor que
>=	maior ou igual que
<=	menor ou igual que
==	igual
<>	diferente

- Estes valores são somente usados quando se deseja efetuar **comparações**.
 - Comparações só podem ser feitas entre objetos de mesma natureza, isto é, variáveis do mesmo tipo de dados.

Operadores Lógicos

- Os operadores lógicos ou booleanos são usados para combinar expressões relacionais.
- Também devolvem como resultado valores lógicos verdadeiro ou falso.

Operador	Tipo	Operação	Prioridade
OU	Binário	Disjunção	3
E	Binário	Conjunção	2
NAO	Unário	Negação	1

Operadores Lógicos

- Tabela Verdade

A	B	A E B	A OU B	NÃO A	NÃO B
V	V	V	V	F	F
V	F	F	V	F	V
F	V	F	V	V	F
F	F	F	F	V	V



Operadores Lógicos

- Forma uma nova proposição lógica a partir de outras proposições lógicas.

EX:

- Se chover **e** relampejar, eu fico em casa.
Quando eu fico em casa?
- Se chover **ou** relampejar, eu fico em casa.
Quando eu fico em casa?