

Introdução a Ciências da Computação

Capítulo 2

O que é um Algoritmo?



Tópicos

- Histórico
- Estruturas de Controle
- Comandos em seqüência
- Estruturas Condicionais
- Álgebra Booleana
- Estruturas de Repetição



O que é um algoritmo?

É um conjunto determinístico de instruções que cumprem alguma tarefa, na qual partindo de um estado inicial chegam a um determinado estado final.

Exemplo:

- Caipirinha
 - Cortar limões
 - Espremer limões
 - Adicionar açúcar
 - Misturar cachaça
 - Colocar gelo



2.1 Histórico



- O nome algoritmo vem do nome do matemático persa **Abu Abdullah Muhammad bin Musa al-Khwarizmi** (780 – 850).
- Entre outras coisas ele desenvolveu uma solução sistemática para funções lineares e quadráticas.
- A palavra 'algarismo' também é derivada de seu nome, devido a seu trabalho relacionado a notação posicional do sistema de numeração decimal.



Histórico

- No século XVIII na Europa, a tradução do nome de *al-Khwarizmi* deu origem à palavra algoritmo.
- O primeiro caso de um algoritmo escrito para um computador foi um programa de Ada Byron para a máquina analítica de Charles Babbage.
- No século XX, as pesquisas de Alan Turing e Alonzo Church formalizaram o que é um algoritmo. A partir disto foi possível definir quais tipo de problemas um algoritmo pode resolver.



Estruras de Controle

- Algoritmos são formas de codificar procedimentos estereotipados.
- Para descrever procedimentos estes têm de ser codificados de uma forma legível e organizada.
- Para organização de descrições de algoritmos usamos 3 **estruturas de controle**:
 - ◆ O **bloco** ou seqüência de comandos: descreve uma seqüência de ações que devem ser executadas uma após a outra.
 - ◆ A **estrutura condicional**: usada para descrever uma situação onde temos de decidir entre duas formas de agir em função de uma situação que poderá ocorrer.
 - ◆ A **estrutura de repetição**: usada para descrever uma situação onde temos de repetir um determinado número de vezes alguma ação até que uma situação seja atingida.
- Originadas do Teorema da Programação Estruturada.



Teorema da Programação Estruturada

- O teorema da programação estruturada (Corrado Böhm and Giuseppe Jacopini, 1966) é resultado da teoria das linguagens de programação. Ela define que cada rotina computável pode ser implementada em uma linguagem que combine os sub-programas em apenas três maneiras específicas:
 - ◆ 1. Executar um subprograma, depois outro sub-programa (seqüência)
 - ◆ 2. Executar um ou dois subprogramas de acordo com o valor de uma variável booleana (condição)
 - ◆ 3. Executar um subprograma até que uma variável booleana seja verdadeira (iteração)



Comandos em Seqüência: Exemplo

Exemplo:

- Caipirinha
 - Cortar limões
 - Espremer limões
 - Adicionar açúcar
 - Misturar cachaça
 - Colocar gelo

A ordem em que as instruções são executadas influi no estado final.

Exemplo: os limões devem ser cortados antes de espremidos.



Estruturas Condicionais

Caipirinha

Cortar limões

Espremer limões

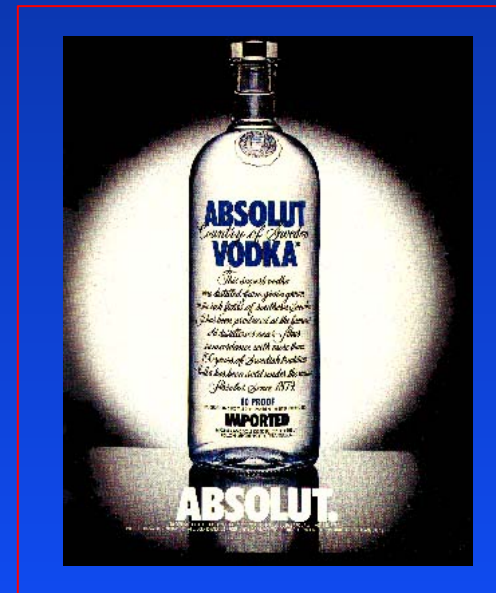
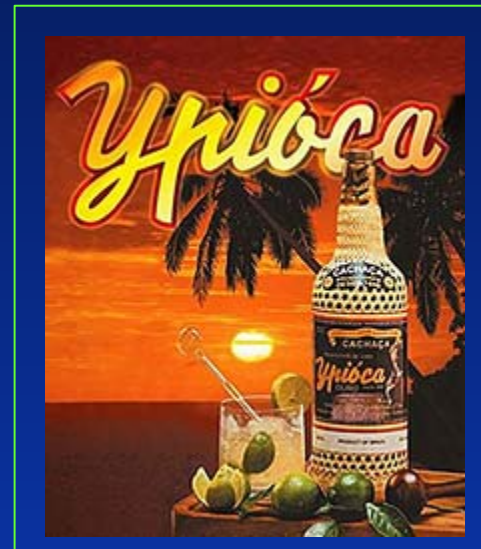
Adicionar açúcar

Se (houver Cachaça) então
misturar cachaça

senão

misturar vodka

Colocar gelo



1.3 Estruturas Condicionais

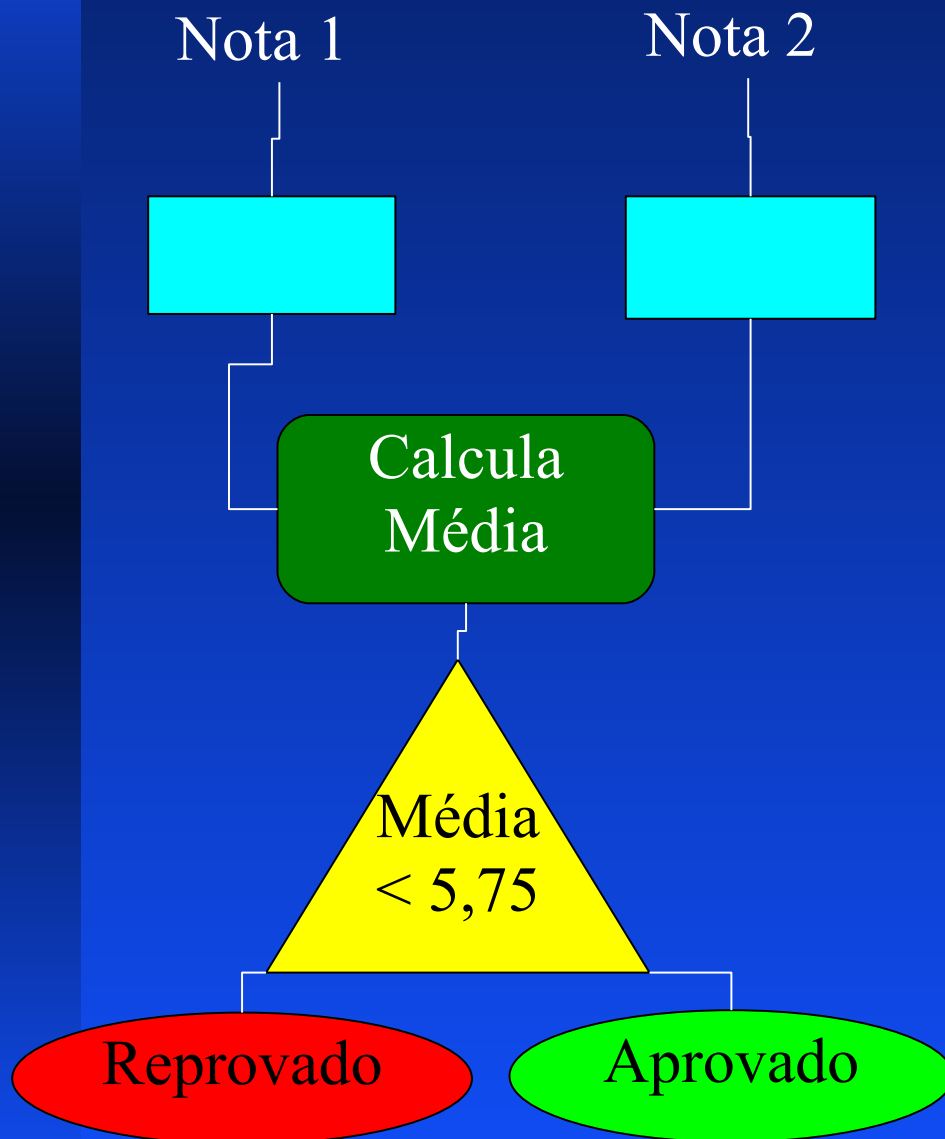
```
SE (condição satisfeita)
ENTÃO
    bloco de comandos 1
SENÃO
    bloco de comandos 2
FIM
```

- A condição a ser satisfeita é o resultado de uma operação lógica.

- Ex:
SE (saldo_do_cliente <0)
ENTÃO “cliente está em débito”



Estruturas Condicionais:



Algoritmo MédiaAluno

Início do Algoritmo

Leia a primeira nota

Leia a segunda nota

Calcule a média

SE (média < 5,75) **ENTÃO**
 escreva “aluno reprovado”

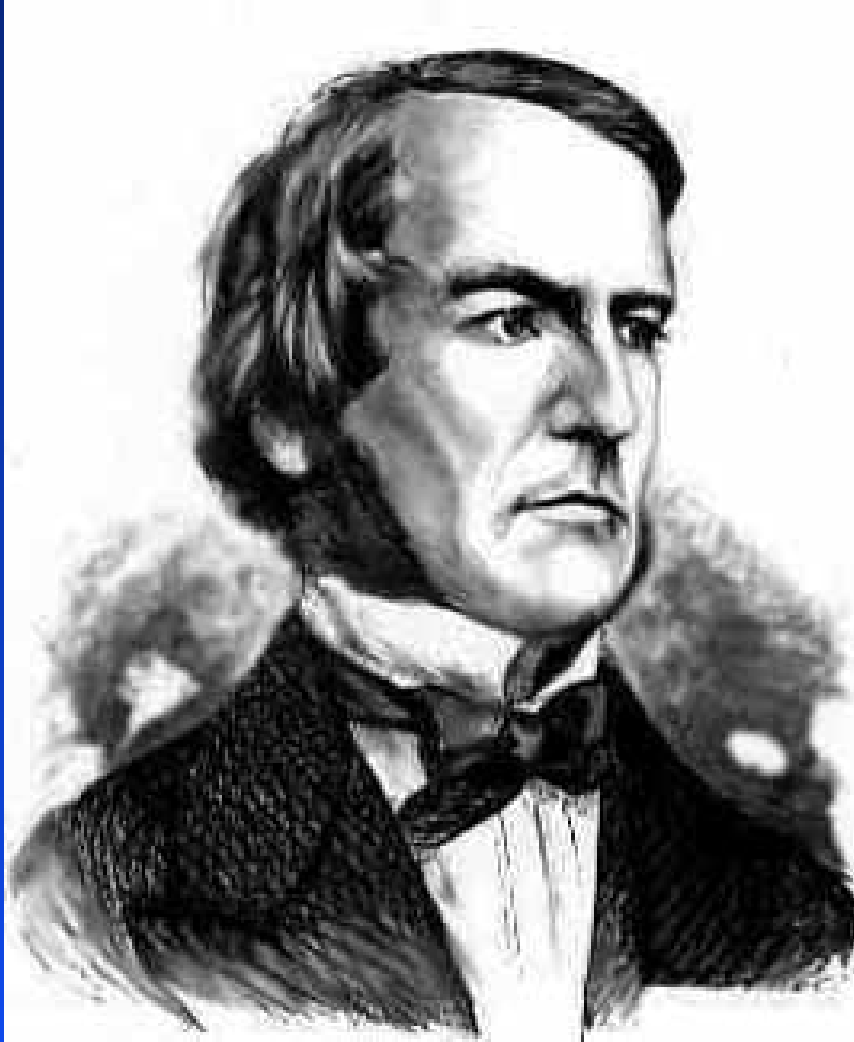
SENÃO
 escreva “aluno aprovado”

FIM

FIM do Algoritmo



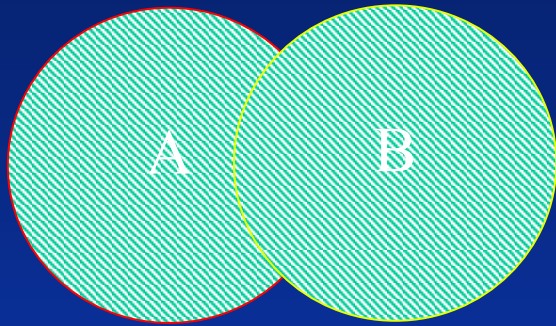
1.2 Álgebra Booleana



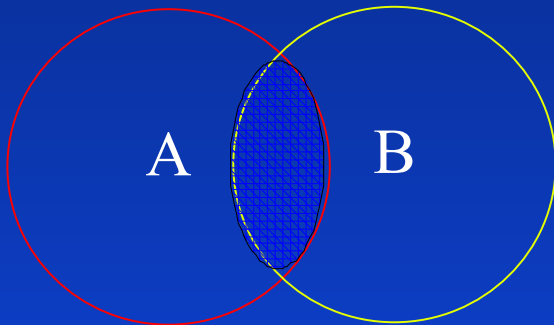
- Desenvolvida pelo inglês George Boole no século XIX.
- É considerada a base de toda a matemática computacional.
- A álgebra de boole envolve operadores essenciais da teoria de conjuntos e lógica.



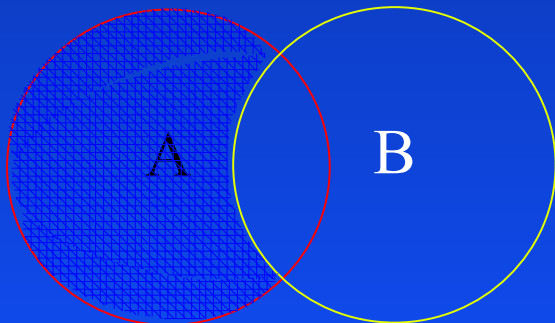
Álgebra Booleana: Operadores de Conjunto



■ União:



■ Intersecção:

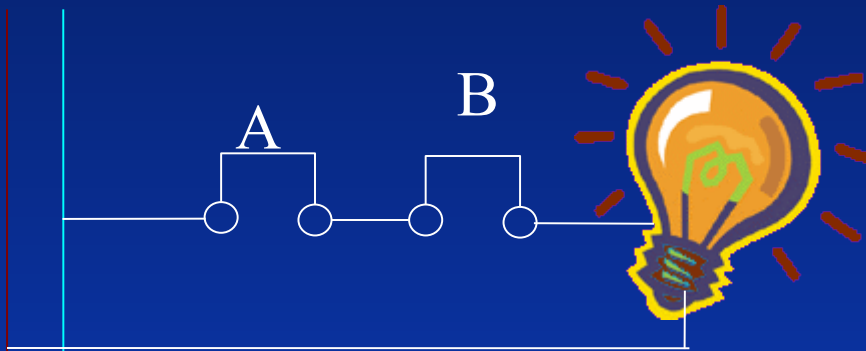


■ Complemento:



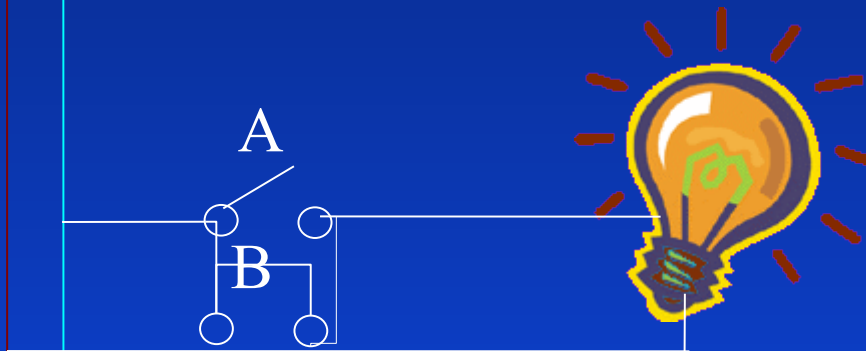
Álgebra Booleana: Operadores Lógicos

$A \text{ and } B = L$



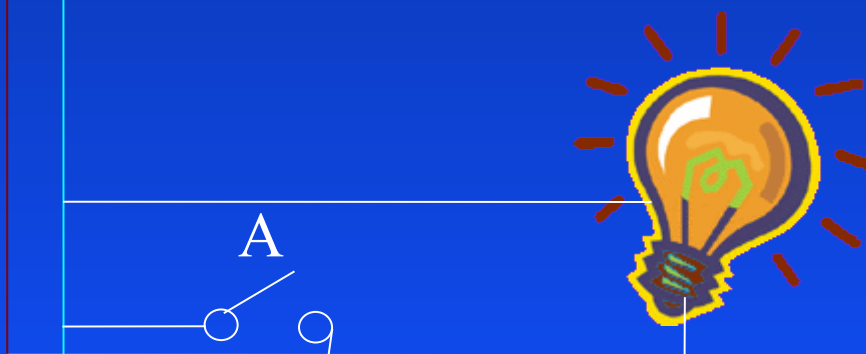
A	B	Lâmpada acessa
0	0	0
0	1	0
1	0	0
1	1	1

$A \text{ or } B = L$



A	B	Lâmpada acessa
0	0	0
0	1	1
1	0	1
1	1	1

$\text{not } A = L$



A	Lâmpada acessa
0	1
1	0



Álgebra Booleana: Operadores Lógicos

S = Caipirinha Razoável

A = Limão azedo

B = Açúcar

C = Gelo

$$S = (((\text{not } A) \text{ and } B) \text{ or } C)$$

A	B	C	Not A	(not A) and B	S
0	0	0	1	0	0
0	0	1	1	0	1
0	1	0	1	1	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	1	0	0	1
1	1	0	0	0	0
1	1	1	0	0	1



Estruturas de Repetição

```
Para i:=1 até 10 faça  
  início  
    bloco_de_comandos  
  fim
```

```
Enquanto( condição_satisfeita) faça  
  início  
    bloco_de_comandos  
  fim
```

```
Repita  
  início  
    bloco_de_comandos  
  fim  
até que( condição_satisfeita )
```



Estruturas de Repetição

Caipirinha

Cortar limões



Espremer limões

Enquanto (muito azedo)

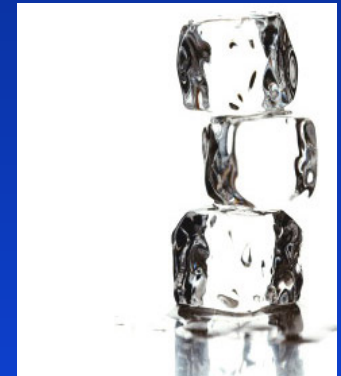
Adicionar açúcar



Misturar cachaça

Repita

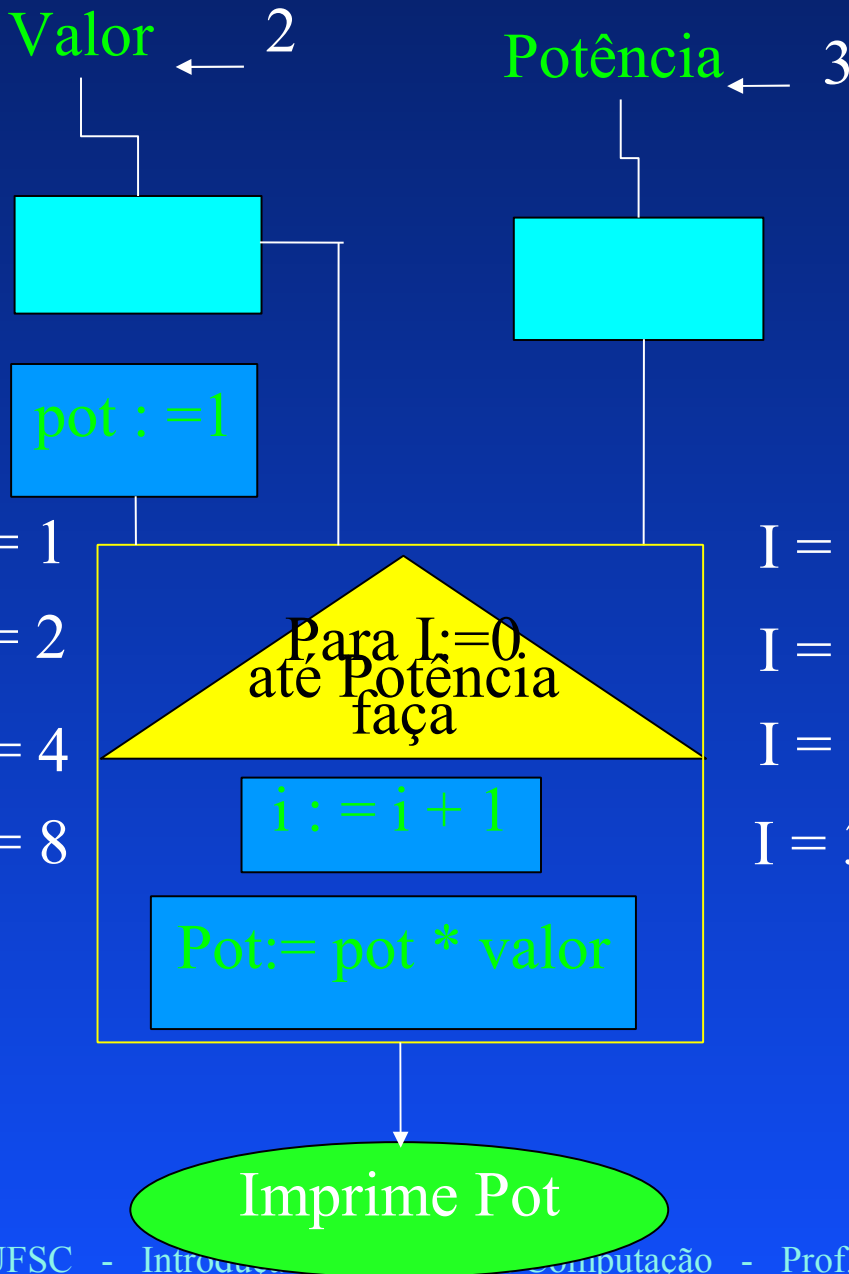
Colocar gelo



até que ((not copo cheio) or (caipirinha muito forte))



Estruturas de Repetição



Algoritmo Potência

```

var
    pot : inteiro
    i : inteiro
    valor : inteiro
    potencia : inteiro
Início
    Leia valor
    Leia potência
    pot := 1;
    Para i:=0 até potência faça
        início
            pot := pot * valor
        fim
    imprime 'valor', valor, 'elevado a
potência', potência, '=', pot
Fim
    
```



EXERCÍCIOS

