



Universidade Federal do Vale do São Francisco

# Algoritmos e Programação

Professor: Marcelo Santos Linder

E-mail: [marcelo.linder@univasf.edu.br](mailto:marcelo.linder@univasf.edu.br)

Página: [www.univasf.edu.br/~marcelo.linder](http://www.univasf.edu.br/~marcelo.linder)

## Ementa

Conceito de algoritmo. Lógica de programação e programação estruturada. Linguagem de definição de algoritmos. Estrutura de um algoritmo. Constantes. Identificadores. Variáveis. Declaração de variáveis. Operações Básicas. Comandos de Entrada e Saída. Estruturas de Controle de Fluxo. Conceito e classificação de Linguagens de Programação. Introdução à uma Linguagem de Programação de alto nível estruturada. Ambiente de programação. Componentes da Linguagem de Programação selecionada: estrutura de um programa, identificadores, palavras reservadas, variáveis, constantes, declaração de variáveis, operações básicas, comandos de entrada e saída, estruturas de controle de fluxo, estruturas de dados homogêneas e modularização.

## Metodologia

- A disciplina será trabalhada com aulas expositivo-dialogadas, onde serão fornecidos os componentes teóricos e será feita a prática de exercícios.

## Recursos

- Quadro branco, marcador, notebook e projetor multimídia.

## Forma de Avaliação

- A avaliação será realizada mediante prova(s) escrita(s) e prova(s) prática(s). Duas provas possuirão datas especificadas previamente e a(s) outra(s) será(ão) realizada(s) em data(s) definida(s) no momento de sua(s) realização(ões). As avaliações com data definida (AVD's) têm, cada uma, o peso igual a dois, já cada avaliação surpresa (AVS) possui peso igual a um. Logo, a média final (MF) do aluno, considerando a realização de duas provas surpresas, resultará da seguinte equação:

$$MF = (AVD1*2 + AVD2*2 + AVS1*1 + AVS2*1)/6$$

- O aluno para obter aprovação deve ter no **mínimo 75% de presença**. As presenças nas aulas práticas são computadas separadamente das presenças na aulas teóricas.

# Bibliografia

## ➤ Bibliografia Básica:

- ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi de. **Fundamentos da Programação de Computadores**. 2ª edição. Editora Pearson Prentice Hall, 2003.
- OLIVEIRA, Alvaro Borges de; BORATTI, Isaias Camilo. **Introdução à Programação - Algoritmos**. Editora Visualbooks, 1999.
- CORMEN, Thomas H.; LEISERSON, Charles E.; RIVEST, Ronald L.; STEIN, Clifford. **Algoritmos, Teoria e Prática**. Editora Campus, 2002.
- SCHILDT, Herbert. **C Completo e Total**. Editora Pearson Makron Books, 2006.

## ➤ Bibliografia Complementar:

- MEDINA, Marco; FERTIG, Cristina. **Algoritmos e Programação - Teoria e Prática**. 2ª Edição. Editora Novatec, 2006.
- FORBELLONE, André; EBERSPÄCHER, Henri. **Lógica de Programação - A construção de algoritmos e estruturas de dados**. 3ª Edição. Editora Pearson Prentice Hall, 2005.
- CARBONI, Irenice de Fátima. **Lógica de Programação**. Editora Thomson, 2003.
- PUGA, Sandra; RISSETTI, Gerson. **Lógica de programação e estruturas de dados com aplicações em Java**. Editora Pearson Prentice Hall, 2003.

# Informações Gerais

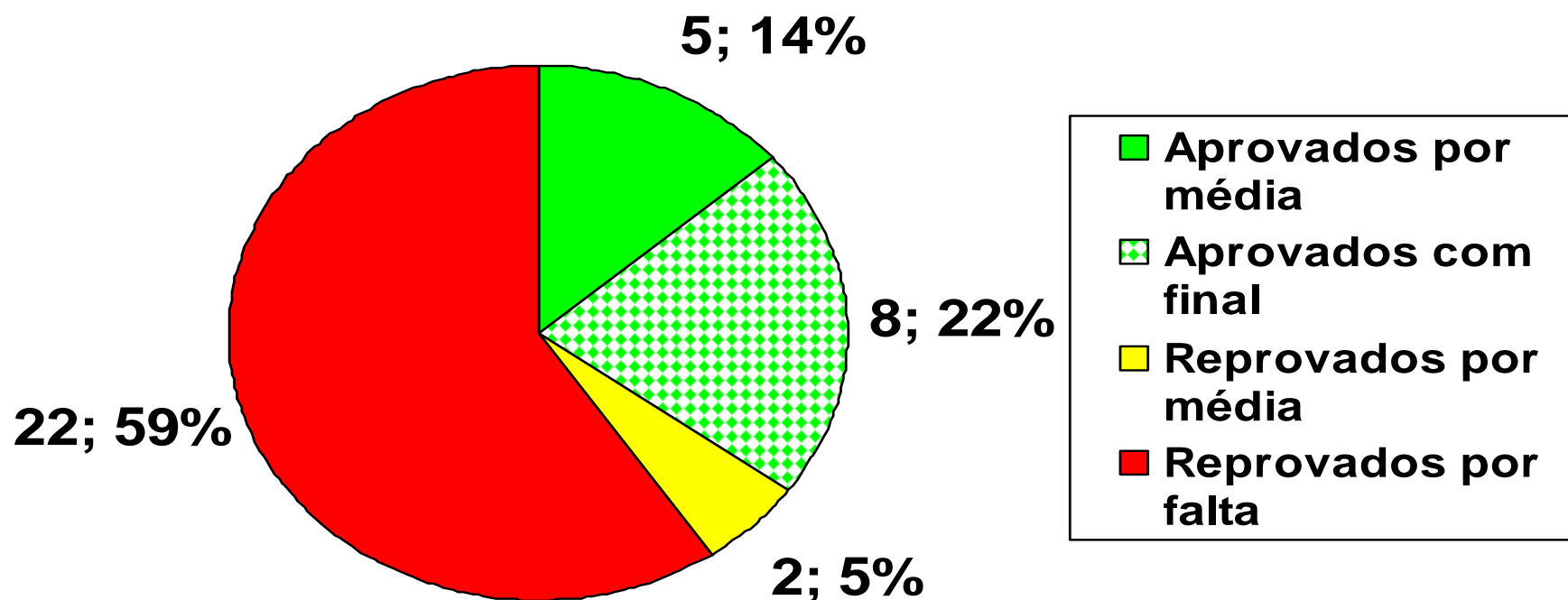
## ➤ Material de apoio

- Os slides utilizados em aula, listas de exercícios, datas de avaliações e demais informações referentes à disciplina serão encontradas na página

[www.univasf.edu.br/~marcelo.linder](http://www.univasf.edu.br/~marcelo.linder)

## Dados sobre a oferta anterior (2010.2)

- Total de discentes: 37 (eram 52)
- Percentual geral de aprovação: 35,14%



# Conceito de Computador

O que é um  
computador?



Não é uma...



## Conceito de Computador

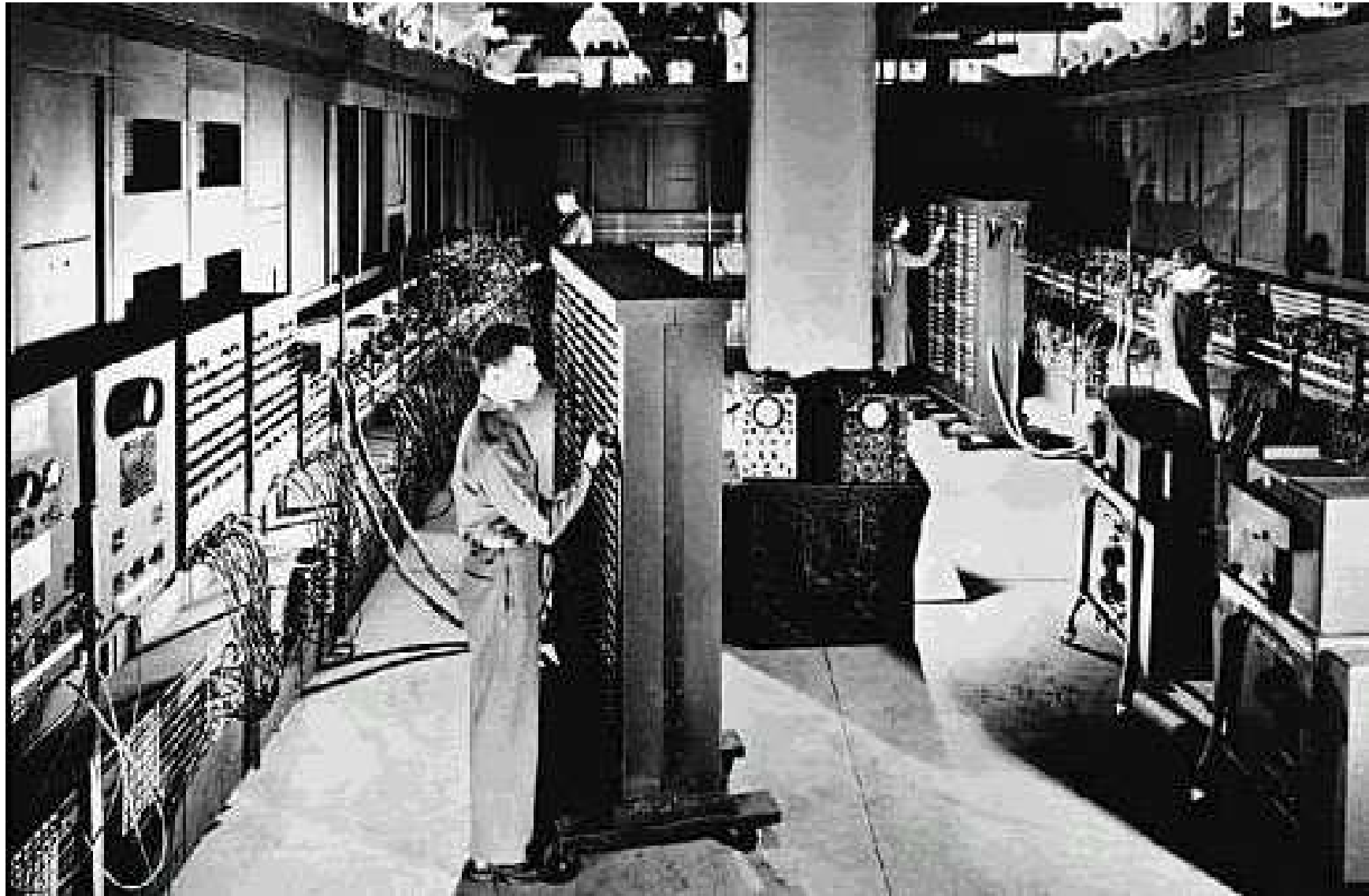
Denomina-se computador uma máquina capaz de variados tipos de tratamento automático de informações ou processamento de dados.

Contudo, o que deve ser feito para que um determinado tratamento automático de informações ou processamento de dados ocorra?

Deve-se instruir o computador para que o mesmo utilizando-se de sua estrutura execute determinada tarefa.



# Conceito de Computador



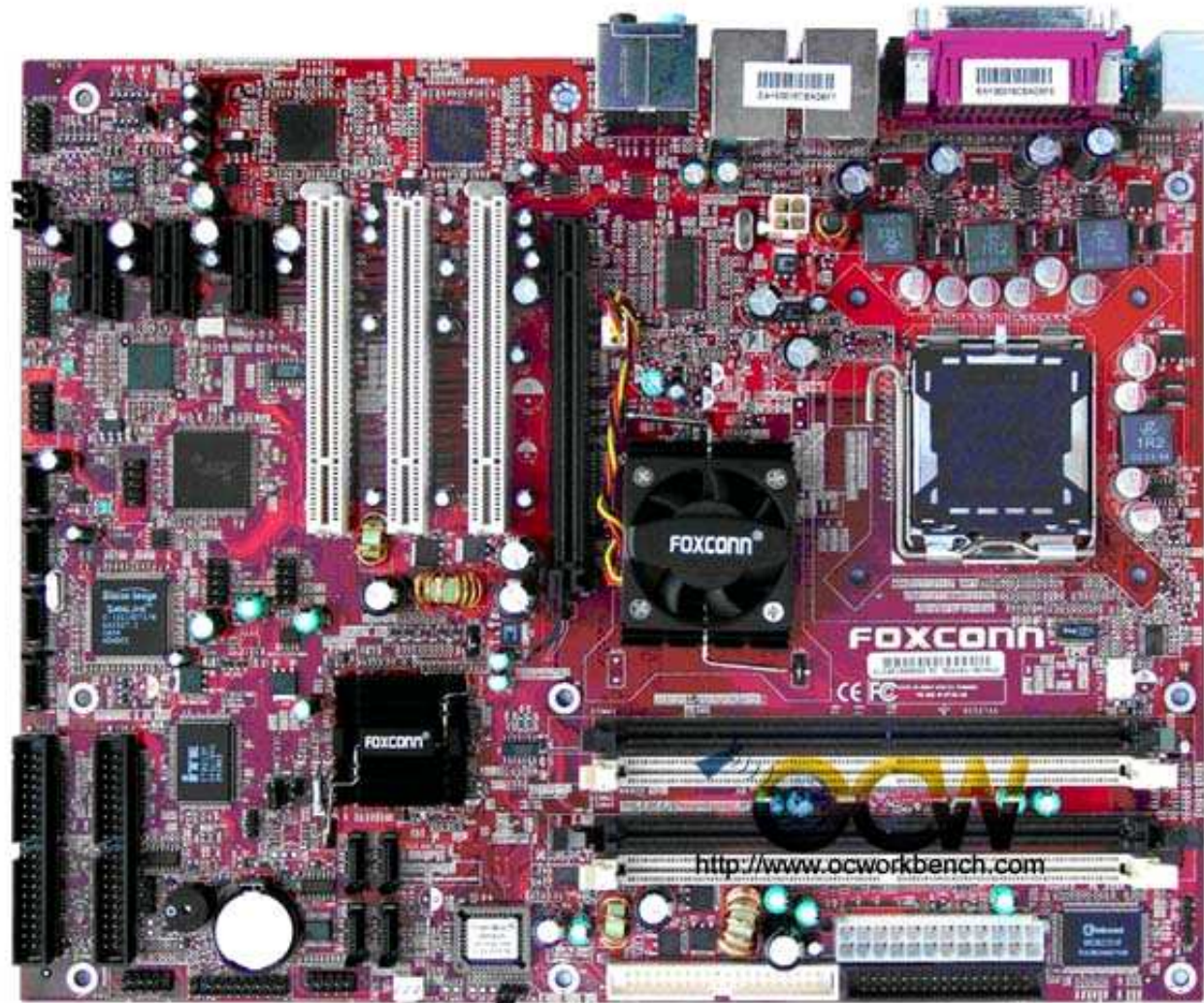
**ENIAC (1946)**

# Conceito de Computador



**Motorola V3**

# Conceito de Computador



Placa Mãe

# Conceito de Computador

**Estrutura de um computador:**



**Unidade de  
Processamento**



**Periféricos de Saída**



**Periféricos de entrada**

# Conceito de Computador

## Estrutura de um computador:



## Conceito de Algoritmo

Como instruir o computador a fazer determinado processamento?

Podemos utilizar uma descrição narrativa em linguagem natural?

Qual a vantagem?

Não há a necessidade de aprender nenhum novo conceito.

Qual a desvantagem?

Em virtude da ambigüidade da linguagem natural a descrição narrativa é passível de mais de uma interpretação.

## Conceito de Algoritmo

Um exemplo de ambigüidade presente em uma sentença na linguagem natural é:

O policial escutou o barulho da porta.

Esta frase pode ter pelo menos três interpretações:

- 1 - O policial escutou o barulho produzido pela porta.
- 2 - O policial estava junto à porta e escutou o barulho.
- 3 - O policial escutou o barulho que veio através da porta.

## Conceito de Algoritmo

Com objetivo de se instruir o computador a fazer determinado processamento impossibilitando mais de uma forma de interpretação, foi criado o conceito computacional de algoritmo, onde:

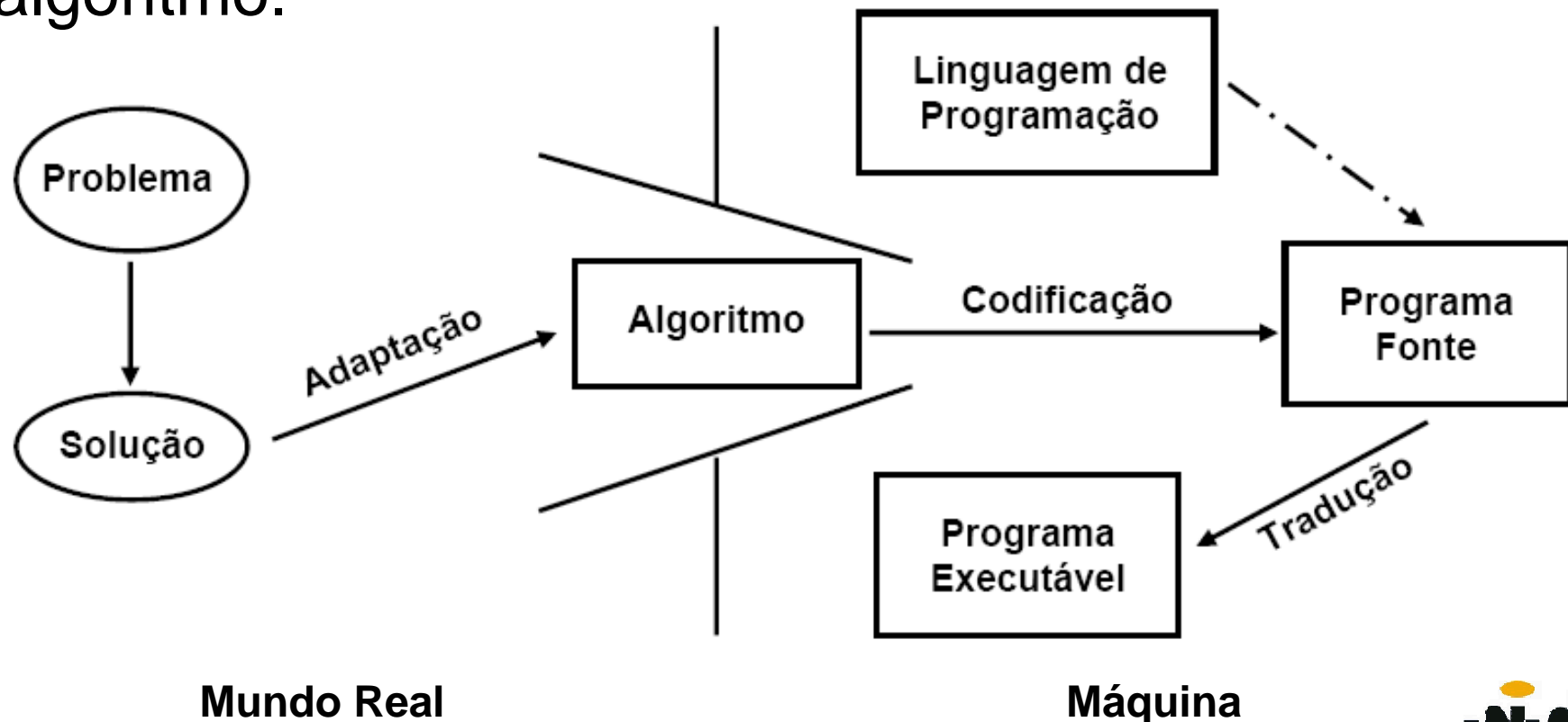
Um algoritmo é uma seqüência, que não permite ambigüidade, de paços finitos, passível de ser executada com um esforço finito em tempo finito e que acaba para qualquer entrada (inclusive erro).

Devemos ter consciência que um computador **não** é dotado da capacidade de tomar decisões com base em premissas. Portanto, não podemos instruir um computador com sentenças dúbias.



# Funcionalidade do Algoritmo

O algoritmo, do ponto de vista computacional, tem um papel fundamental por ser o elo de ligação entre dois mundos (real e computacional). A atividade de programação tem início com a construção do algoritmo.



## Exemplos de algoritmos

Como vimos o conceito de algoritmo é bem amplo, sendo importante salientar que qualquer tarefa que siga determinado padrão pode ser descrita por um algoritmo, como por exemplo:

### **ALGORITMO: TROCAR UMA LÂMPADA**

**PASSO 1: Pegar a lâmpada nova**

**PASSO 2: Pegar a escada**

**PASSO 3: Posicionar a escada embaixo da lâmpada queimada**

**PASSO 4: Subir na escada com a lâmpada nova**

**PASSO 5: Retirar a lâmpada queimada**

**PASSO 6: Colocar a lâmpada nova**

**PASSO 7: Descer da escada**

**PASSO 8: Ligar o interruptor**

**PASSO 9: Guardar a escada**

**PASSO 10: Jogar a lâmpada velha no lixo**

### **ALGORITMO: SACAR DINHEIRO**

**PASSO 1: Ir até o caixa eletrônico**

**PASSO 2: Colocar o cartão**

**PASSO 3: Digitar a senha**

**PASSO 4: Solicitar o saldo**

**PASSO 5: Se o saldo for maior ou igual à quantia desejada, sacar a quantia desejada; caso contrário sacar o valor do saldo**

**PASSO 6: Retirar dinheiro e cartão**

**PASSO 7: Sair do caixa eletrônico**

Este método de representação de algoritmos é adequado?

18 Por que?

# Métodos de Representação de Algoritmos

Veremos dois métodos para representação de algoritmos:

- ✦ fluxograma – representação gráfica;
- ✦ pseudocódigo (português estruturado) – representação textual.

## Métodos de Representação de Algoritmos

Os métodos em questão impõem regras e disponibilizam um conjunto reduzido de palavras/símbolos passíveis de serem utilizadas(os) (rigidez sintática). O objetivo é obter uma consistência semântica para a eliminação da ambigüidade intrínseca à linguagem natural.

Destacaremos agora uma vantagem e uma desvantagem de cara uma das formas, mencionadas, de representação de algoritmos.

# Métodos de Representação de Algoritmos

## Fluxograma:

Vantagem – a representação gráfica é mais concisa que a representação textual.

Desvantagem – é necessário aprender a simbologia dos fluxogramas.

## Pseudocódigo:

Vantagem – sua transcrição para qualquer linguagem de programação é quase que direta.

Desvantagem – é necessário aprender as regras do pseudocódigo.

# Métodos de Representação de Algoritmos

Analisaremos a solução do problema de trocar a resistência de um chuveiro resolvido com um algoritmo representado em pseudocódigo.

Conforme foi mencionado são impostas regras e é definido um número restrito de ações. Neste caso as ações disponíveis são: pegar, largar, abrir, fechar, retirar e colocar.

## Descrição Narrativa

Adquira uma resistência nova e localize o chuveiro a ser manipulado. Em seguida abra o chuveiro retirando a resistência defeituosa, coloque a resistência nova e feche o chuveiro. Após descarte a resistência defeituosa.

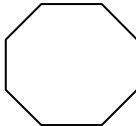
## Pseudocódigo


1. Pegar (resistência nova);
2. Pegar (chuveiro);
3. Abrir (chuveiro);
4. Retirar (resistência defeituosa);
5. Colocar (resistência nova);
6. Fechar (chuveiro);
7. Largar (resistência defeituosa).

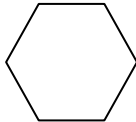
## Métodos de Representação de Algoritmos


Solução do problema de trocar a resistência de um chuveiro resolvido com um algoritmo representado em fluxograma.

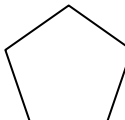
Conforme foi mencionado são impostas regras e é definido um número restrito de ações. Neste caso as ações disponíveis e os símbolos a elas associados são:


pegar  $\langle \Rightarrow \rangle$  

largar  $\langle \Rightarrow \rangle$  

fechar  $\langle \Rightarrow \rangle$  

abrir  $\langle \Rightarrow \rangle$  

colocar  $\langle \Rightarrow \rangle$  

retirar  $\langle \Rightarrow \rangle$  

# Métodos de Representação de Algoritmos

## Fluxograma

### Descrição Narrativa

Adquira uma resistência nova e localize o chuveiro a ser manipulado. Em seguida abra o chuveiro retirando a resistência defeituosa, coloque a resistência nova e feche o chuveiro. Após, descarte a resistência defeituosa.

### Conjunto de símbolos disponíveis

pegar <=>



largar <=>



fechar <=>



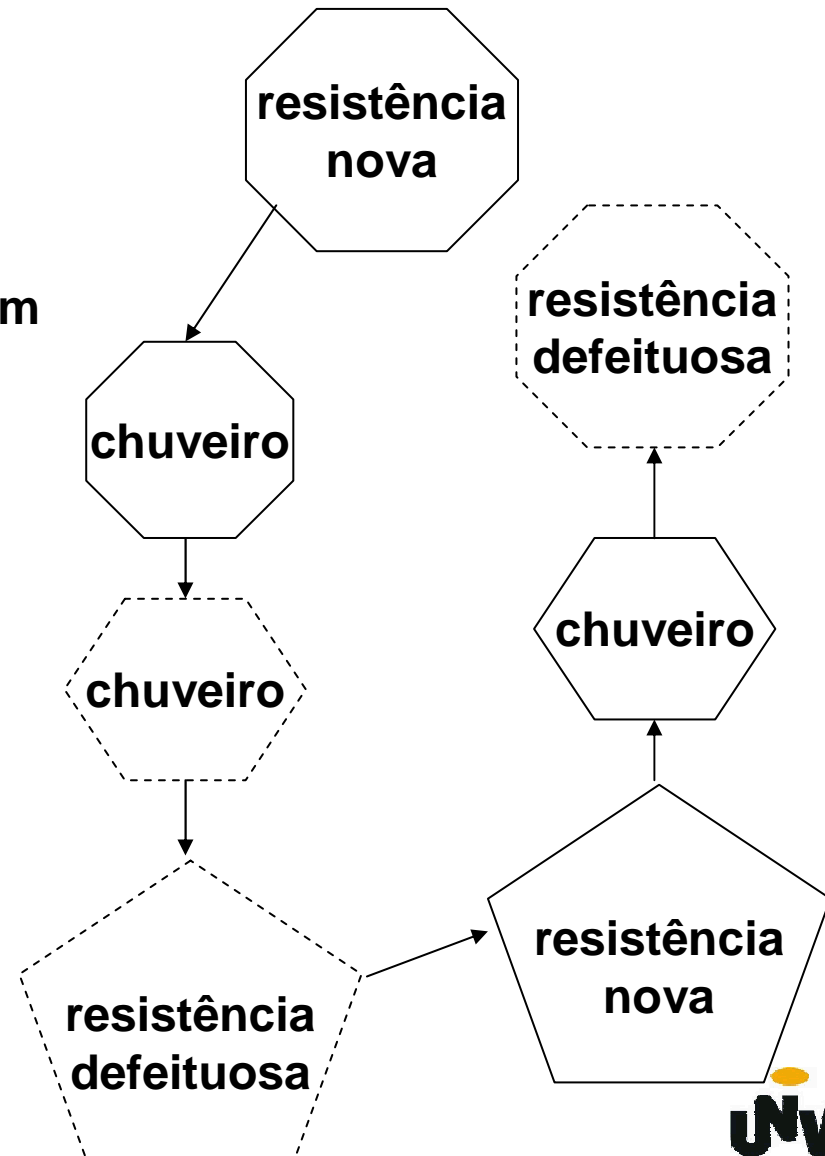
abrir <=>



colocar <=>



retirar <=>





# Conceitos Básicos de Algoritmos

Antes de nos aprofundarmos nos métodos de representação de algoritmos, devemos ter de forma clara a compreensão de alguns conceitos como:

- Constante;
- Variável;
- Identificador;
- Palavra-reservada;
- Entrada;
- Saída;
- Operadores.

# Conceitos Básicos de Algoritmos

Para uma melhor compreensão dos conceitos citados, analisaremos o problema de preparar uma omelete.

Partiremos da lista de ingredientes:

- 5 ovos
- 1/3 de uma xícara de chá de leite
- 2 colheres de sopa de cebolinha verde picada
- 3 pitadas de sal
- 1 colher de sopa de manteiga

## Conceitos Básicos de Algoritmos

O procedimento de preparo, em linguagem natural, é o seguinte:

Colocar em uma tigela os ovos, o leite, a cebolinha e o sal. Com a ajuda de uma espátula bater bem os ingredientes contidos na tigela. Colocar a manteiga em uma frigideira e a derreter. Colocar o conteúdo da tigela na frigideira e fritar em fogo baixo até a omelete dourar suavemente. Quando estiver quase seca, dobrar a omelete ao meio, colocar a omelete em um prato. A omelete está pronta para servir.

## Conceitos Básicos de Algoritmos

Ao analisarmos o procedimento de preparo percebemos que são necessários três recipientes: uma tigela, uma frigideira e um prato, além de uma espátula.

Podemos visualizar os recipientes como **variáveis**, pois estes são regiões do espaço onde pode-se armazenar inúmeras substâncias. A espátula como uma **constante** devido a esta representar um elemento que participa do processo de preparo da omelete e que ao final se mantém inalterado.

## Conceitos Básicos de Algoritmos

Percebemos também que são executadas algumas operações: colocar, bater, derreter, fritar, dobrar e servir.

Os nomes dos recipientes e os verbos correspondentes às ações podem ser visualizados como **identificadores**, uma vez que estes especificam o que será manipulado ou como será manipulado. Uma observação a ser feita é o fato de não podermos nomear um recipiente com um verbo correspondente a uma ação utilizada no processo, logo os verbos podem ser considerados como **palavras-reservadas**.

Os ingredientes constituirão as **entradas** para o processo e a omelete será a **saída**.

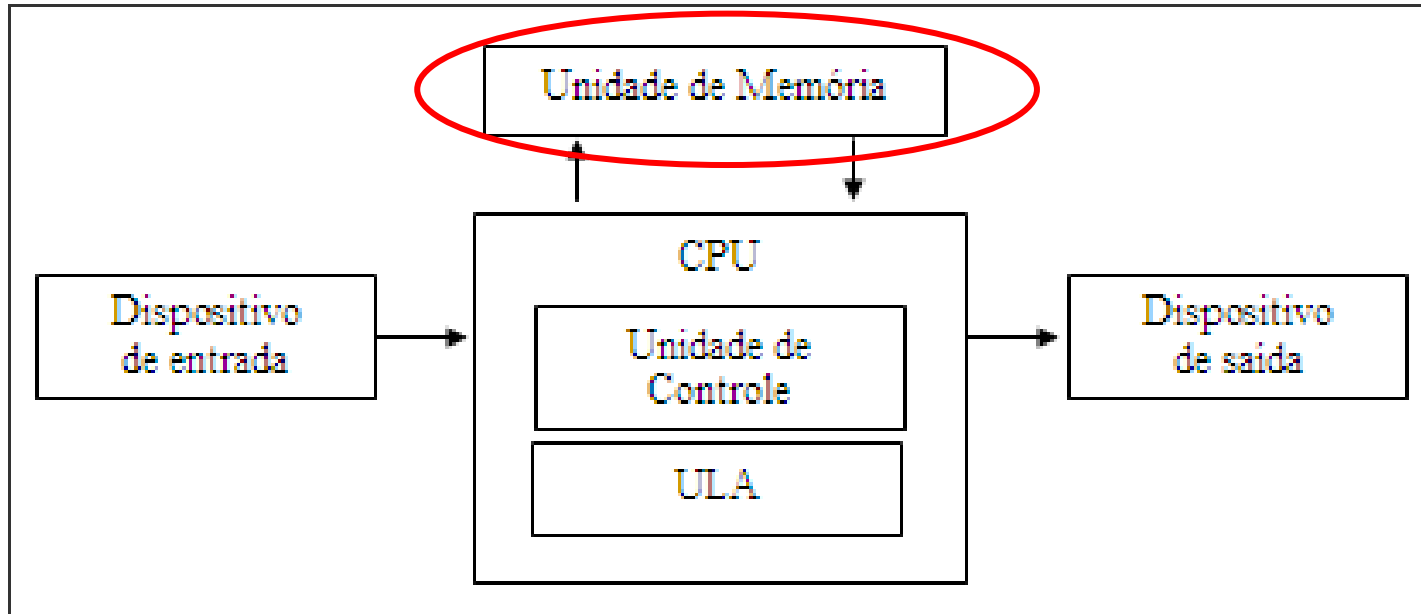
## Conceitos Básicos de Algoritmos

Nosso objetivo final com o estudo de algoritmos é a aplicação computacional dos mesmos.

Desta forma devemos contextualizar os conceitos vistos. Definiremos quais serão as entradas possíveis para os procedimentos que virão a constituir soluções de futuros problemas e especificaremos quais as formas de manipulação das mesmas.

# Conceitos básicos de algoritmos

## Arquitetura de John Von Neumann



# Conceitos Básicos de Algoritmos

Constante →

São Valores fixos, tais como números. Estes valores não podem ser alterados pelas instruções do algoritmo, ou seja, é um espaço de memória cujo valor não deve ser alterado durante a execução de um algoritmo.

Exemplos:

inteiro (10, -23768)

real (-2.34, 0.149)

caractere (“a”, “professor”)



# Conceitos Básicos de Algoritmos

Variável →

é um espaço de memória que recebeu um nome (identificador) e armazena um valor que pode ser modificado durante a execução do algoritmo.

Identificadores →

são os nomes utilizados para referenciar variáveis, funções ou vários outros objetos definidos pelo construtor do algoritmo.

Regras para a construção de identificadores:

- podem conter letras, dígitos e sublinhado(\_);
- **não** podem começar com dígito;
- **não** podem ser iguais a uma palavra-reservada e nem iguais a um nome de uma função declarada pelo construtor do algoritmo ou disponibilizada pelo método utilizado para construção de algoritmos.

# Conceitos Básicos de Algoritmos

Palavras-reservadas (palavras-chave) →

são identificadores predefinidos que possuem significados especiais para o interpretador do algoritmo.

|        |         |          |        |
|--------|---------|----------|--------|
| inicio | senao   | para     | repita |
| var    | logico  | se       | ate    |
| faca   | inteiro | enquanto | real   |