

Introdução a Ciências da Computação

Capítulo 1

Como funciona um computador



Tópicos

1. Histórico
2. Componentes
3. Execução de um programa
4. Sistema Operacional
5. Linguagens de Programação
6. Compilador / Interpretador
7. Diferentes tipos de linguagem



O que é um computador?

- Originalmente o termo “computador” se referia a uma pessoa que realizava cálculos matemáticos, freqüentemente com o auxílio de dispositivos mecânicos.
- Atualmente pode ser definido como: Máquina desenvolvida para manipular dados de entrada de acordo com uma lista de instruções conhecida como programa.

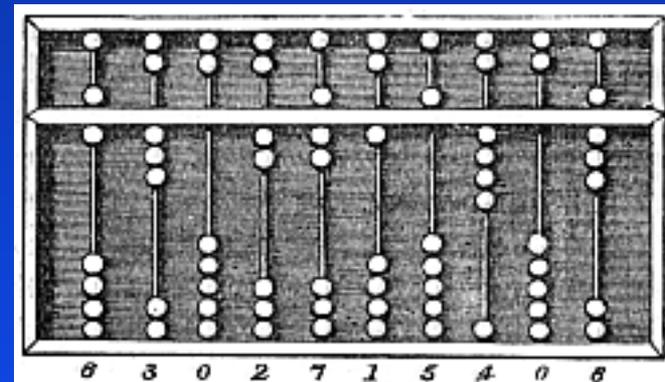


Histórico - Ábaco

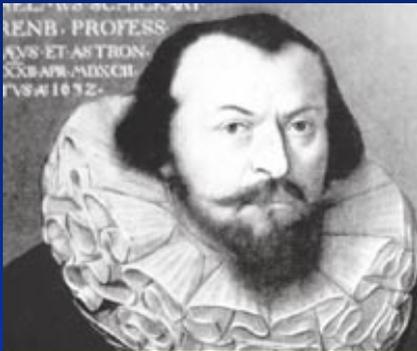
- Babilônia 2400 - 300 A.C.

- Ábaco romano

- Chinês – Permite realizar as quatro operações básicas, raiz quadrada e cúbica



Histórico – Calculadoras Mecânicas



- Primeira calculadora mecânica - William Schickard (1623) Alemanha.



- Pascaline – Blaise Pascal (1642) França.



- Arithmometer – Charles Xavier Thomas de Colmar (1820) França. Realizava as quatro operações básicas.



Histórico – Charles Babbage



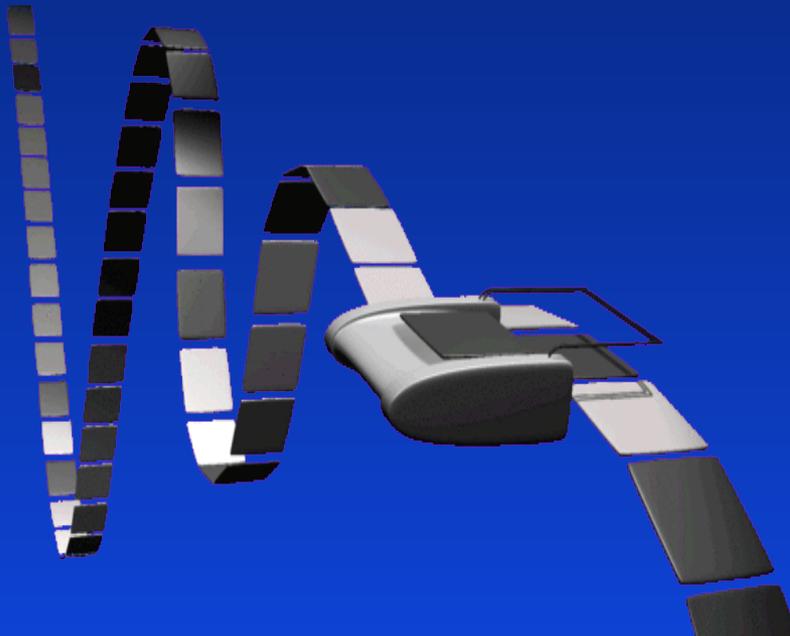
- Máquina diferencial (1822): Babbage buscava criar uma máquina totalmente automatizada, minimizando os erros gerados por humanos na entrada dos dados. (Foto – máquina construída em 1989-1991)



- Máquina analítica – Primeira máquina capaz de executar uma lista de instruções contidas em cartões perfurados.



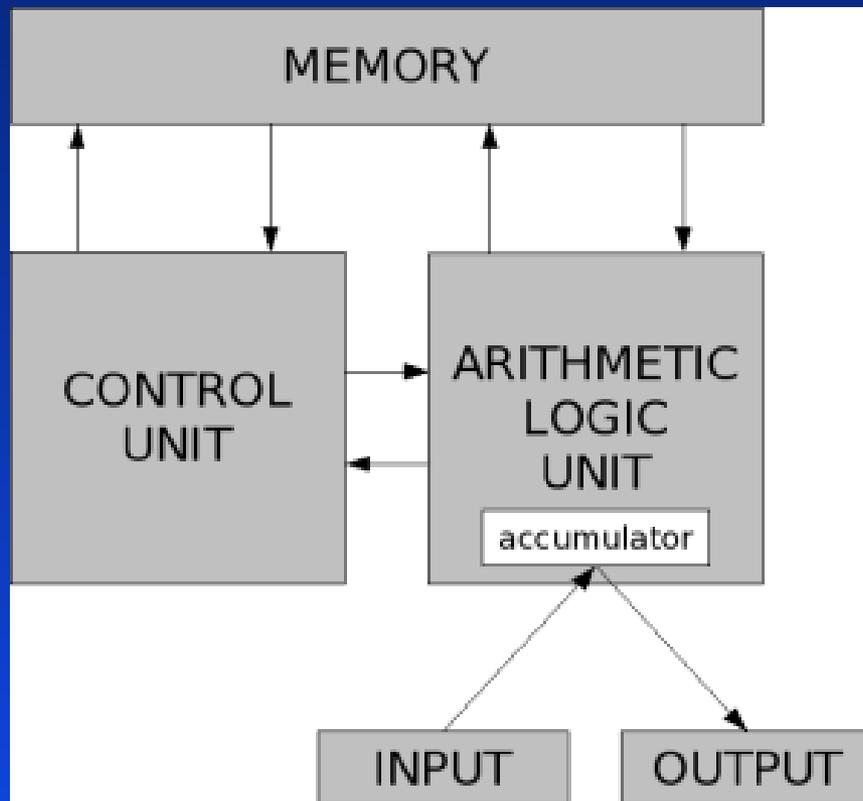
Histórico – Máquina de Turing



- Em 1936, o inglês Alan Turing descreveu uma máquina teórica que serviria como base para descrever os componentes do que futuramente fariam parte de um computador.
- Consiste de uma um cabeçote de leitura/escrita que manipula dados contidos em uma fita com instruções. A posição seguinte da fita a ser lida depende do 'estado atual' registrado.



Histórico – Arquitetura de von Neumann



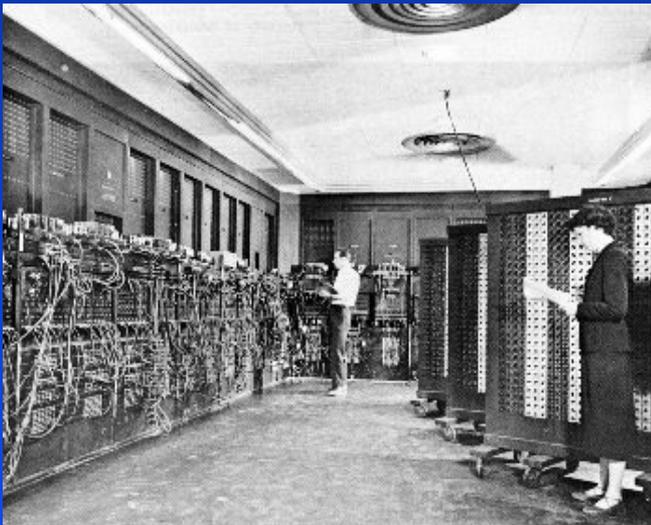
- Em 1945, o húngaro John Von Neumann descreveu uma implementação dos conceitos da máquina de Turing.
- O unidade de controle é capaz de executar individualmente um conjunto de instruções pré-definidas. O programa contido na memória é uma seqüência de instruções deste conjunto.



Histórico – Computadores Eletrônicos



- Small Scale Experimental Machine (SSEM) - Inglaterra – 1948: Primeiro computador eletrônico baseado na arquitetura de von Neumann.



- Electronic Numerical Integrator and Computer (ENIAC) – Desenvolvido pelo exército americano em 1942. Em 1948 foi modificado para rodar programas armazenados em memória.



1.2 Componentes de um Computador

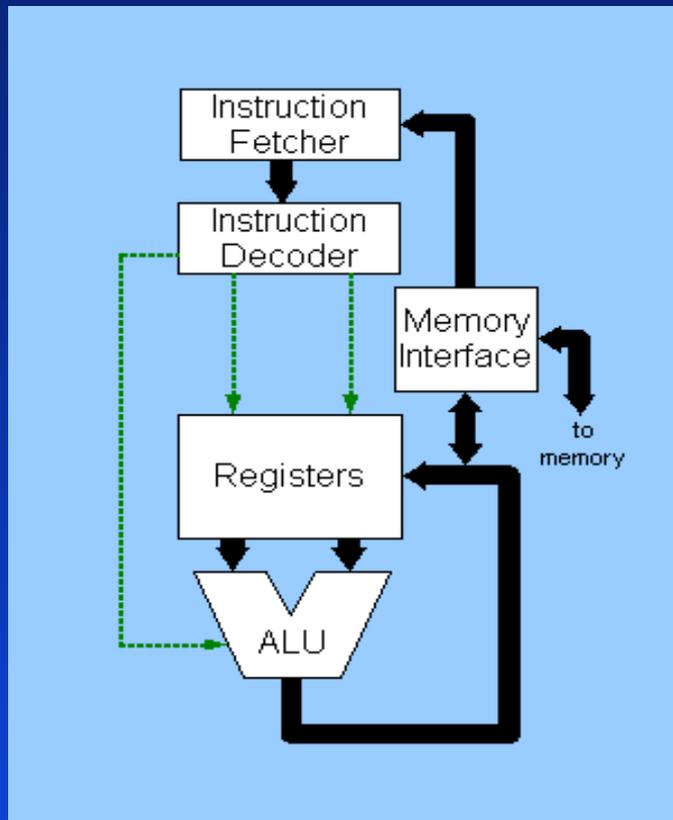
Processador



- É responsável por interpretar dados e processar as informações contidas em um programa.
- Composto por um conjunto de transistores que implementam portas lógicas que realizam as operações matemáticas.
- A miniaturização na construção de transistores tornou possível aplicação destes componentes em outros dispositivos, ao invés de apenas computadores.



Processador - Funcionamento



MIPS32 Add Immediate Instruction

001000	00001	00010	0000000101011110
OP Code	Addr 1	Addr 2	Immediate value

Equivalent mnemonic: **addi \$r1, \$r2, 350**

- O processador carrega uma instrução da memória e a interpreta.
- A instrução interpretada é passada para a Unidade Lógica e Aritmética (ALU).
- A ALU executa a instrução, podendo escrever o resultado em um registrador, que pode ser gravado na memória.
- Exemplo: Essa sequência de 0s e 1s, equivale a somar 350 ao registrador r2 e guardar o resultado no registrador r1.



Memória RAM



- Memória de Acesso Aleatório (RAM) – também conhecida como memória principal. Tem esse nome porque os dados contidos nela podem ser acessados diretamente em qualquer ordem. Em contraste com a fita da máquina de Turing, de acesso sequencial.
- O programas executados pelo processador ficam armazenados na memória RAM, que possuem um tempo de acesso muito superior ao do disco rígido.



Disco Rígido



- Armazena os dados em discos com uma superfície de camada magnética.
- Os dados são escritos/lidos através de um cabeçote de leitura que aplica um campo magnético bem próximo aos discos.
- Os dados armazenados ficam retidos no disco quando o sistema é desligado. Não volátil, conhecida também como memória secundária.



Periféricos

- Dispositivos de saída de dados: Monitor, Caixas de Som, impressora, leitor de cd/dvd...
- Dispositivo de entrada de dados: teclado, mouse, scanner, microfone...
- Dispositivos de entrada e saída de dados: Modem, interface de rede, monitor touch screen...



1.3 Execução de um Programa

- Um programa é um conjunto de instruções que é carregado na memória principal de um computador.
- Cada instrução é interpretada e executada pelo processador até o final do programa.
- O índice referente à próxima instrução a ser executada pelo processador é armazenado em um registrador conhecido como program counter (PC).
- Dependendo dos resultados obtidos na execução de uma instrução, o valor guardado no PC pode se referir a uma instrução com índice anterior ao da instrução executada no momento.
- Um programa pode receber dados provenientes de um dispositivo de entrada e jogar os resultados em um dispositivo de saída.



Exemplo de programa: assembly

```
.text
.globl __start
__start:
    la $a0,prompt    # print prompt on terminal
    li $v0,4
    syscall

    li $v0,5         # syscall 5 reads an integer
    syscall

    mul $t0,$v0,9    # to convert,multiply by 9,
    div $t0,$t0,5    # divide by 5,then
    add $t0,$t0,32   # add 32

    la $a0,ans1     # print string before result
    li $v0,4
    syscall

    move $a0,$t0    # print result
    li $v0,1
    syscall

    la $a0,endl     # system call to print
    li $v0,4        # out a newline
    syscall

    li $v0,10
    syscall         # au revoir...
```

```
.data
prompt: .asciiz "Enter temperature (Celsius): "
ans1: .asciiz "The temperature in Fahrenheit is "
endl: .asciiz "\n"

##
## end of file temp.a
```

```
##
## temp.a ask user for temperature in Celsius,
## convert to Fahrenheit, print the result.
##
## v0 - reads in celsius
## t0 - holds Fahrenheit result
## a0 - points to output strings
##
```

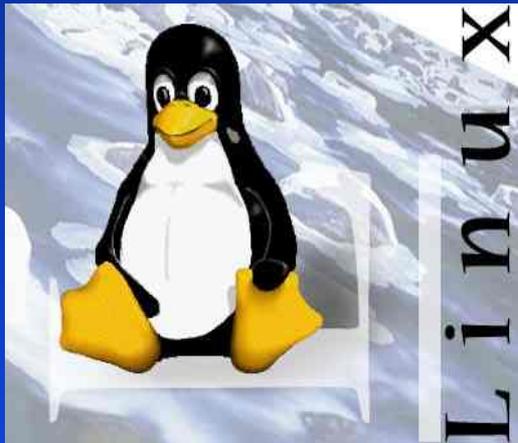
```
#####
#####
#                               #
#           text segment       #
#                               #
#####
```

```
#####
#####
#                               #
#           data segment      #
#                               #
#####
```



1.4 Sistema Operacional

- O sistema operacional é um programa que gerencia os recursos de hardware e software de um computador.



Sistema Operacional

- Quando um computador é ligado o núcleo do sistema operacional, conhecido como kernel, é carregado na memória.
- O kernel é responsável por prover acesso a memória, espaço em disco e organizar o ordem de execução de outros programas.
- Dispositivos como vídeo, som, rede geralmente não são acessados diretamente pelo kernel. Ele os controla através de programas chamados módulos ou drivers.



1.5 Linguagens de Programação

- Linguagem de programação é uma técnica de comunicação usada para controlar o comportamento de uma máquina.
- Assim como as linguagens de comunicação dos seres humanos, uma linguagem de programação deve respeitar regras léxicas, sintáticas e semânticas
- Uma linguagem de programação deve especificar os tipos de dados manipulados por ela e como o fluxo de instruções é controlado.



Linguagens de Programação: exemplo

- Um programa em pascal que calcula o fatorial de um número.

```
program fatorial;
var
    fat, a : integer;
begin;
    writeln('Digite um numero');
    readln(a);
    fat:= a;
    while (a > 1) do
    begin
        a := a-1;
        fat := fat * a;
    end;
    writeln('O fatorial é=',fat);
end.
```



The screenshot shows a terminal window titled "diego@spline:~/Doc/Mestrado/EstagioDocencia/Aula1/Ex" with a menu bar containing "Session Edit View Bookmarks Settings Help". The terminal output is as follows:

```
diego@spline ~/Doc/Mestrado/EstagioDocencia/Aula1/ExemploPas
cal $ ./fatorial
Digite um numero
5
O fatorial é=120
diego@spline ~/Doc/Mestrado/EstagioDocencia/Aula1/ExemploPas
cal $ ./fatorial
Digite um numero
6
O fatorial é=720
diego@spline ~/Doc/Mestrado/EstagioDocencia/Aula1/ExemploPas
cal $
```

The terminal window also shows a taskbar at the bottom with icons for "Shell" and "Shell No. 2".



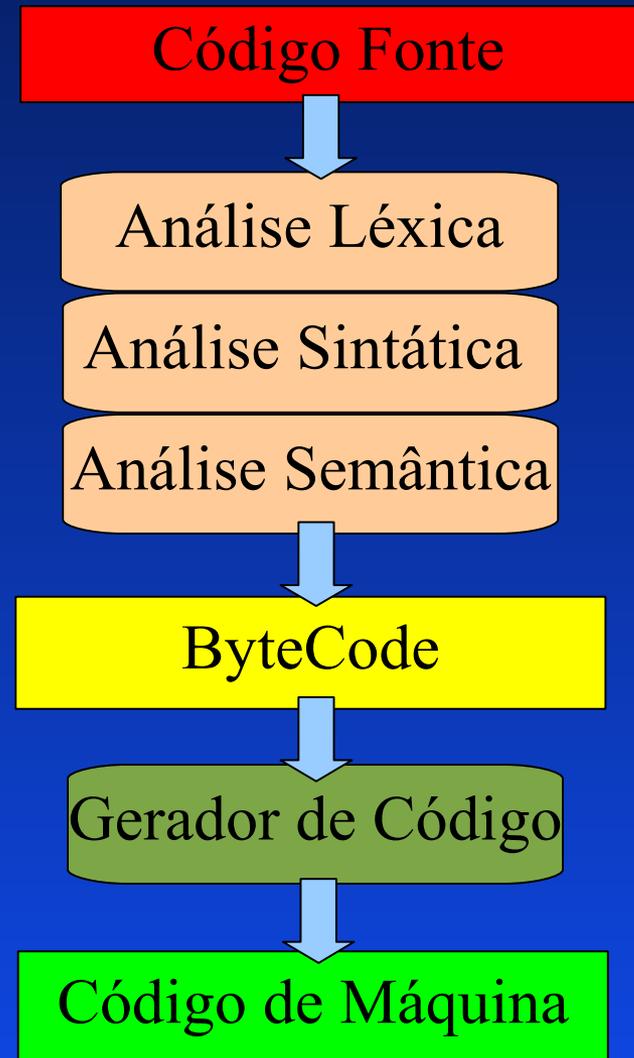
Linguagens de Programação

- Um processador e não é capaz de executar diretamente as instruções de uma linguagem como o Pascal. Ele 'entende' instruções apenas em uma linguagem chamada linguagem de máquina.
- Os programas não são escritos diretamente em linguagem de máquina porque ela é muito distinta da linguagem natural dos seres humanos.
- Um programa deve ser 'traduzido' para a linguagem que o processador 'entende'.
- Existem dois tipos de programas que executam esta tradução: o compilador e o interpretador.



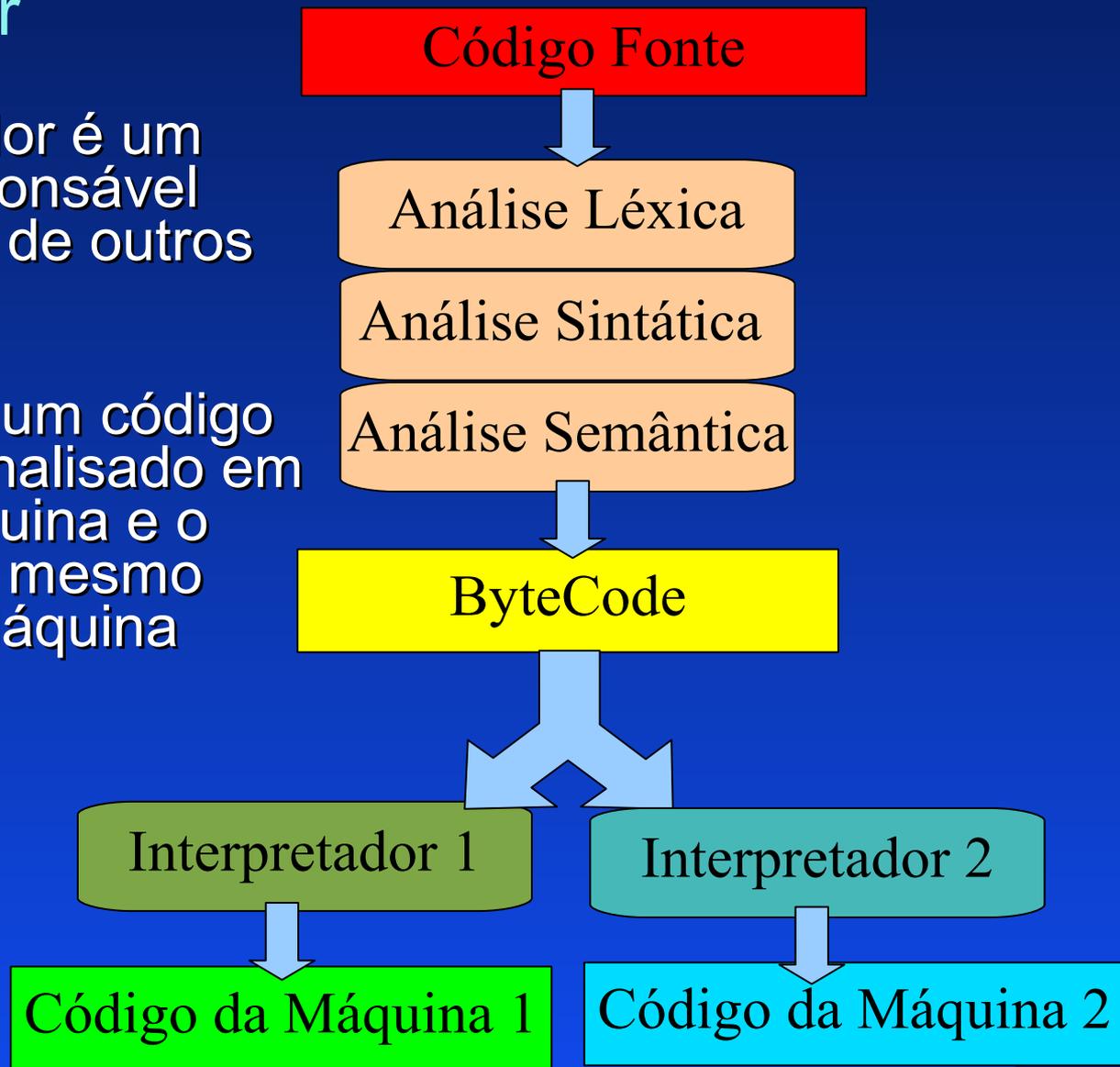
1.6 Compilador

- É um programa que traduz um programa escrito em uma linguagem (código fonte) para uma linguagem (código de máquina) que pode ser executada por um processador.
Exemplo: Turbo Pascal, Borland Delphi, gcc...



1.7 Interpretador

- Um interpretador é um programa responsável pela execução de outros programas.
- Ele 'interpreta' um código previamente analisado em código de máquina e o executa nesse mesmo instante. Ex: Máquina Virtual Java.



Tipos Diferentes de Linguagens

Linguagens Imperativas.

- É um estilo de programação que descreve os estados de um programa e as transições capazes de alterar esse estado.
- Programação Estruturada: pascal, c, fortran, matlab
- Programação Orientada a Objeto: object pascal, java, c++



Tipos Diferentes de Linguagens

Linguagens Declarativas

- Uma linguagem de programação é declarativa se ela descreve **o que** é algo, ao invés de **como** criá-lo. Exemplo: html. Descreve como é uma página: título, fontes, imagens... Ao invés de dizer **como** ela deve ser criada.
- Programação Funcional: lisp, scheme
- Programação em Lógica: prolog



Referências

- How Stuff Works: www.howstuffworks.com
- Wikipedia: www.wikipedia.org
- Computer History: www.computerhistory.org

