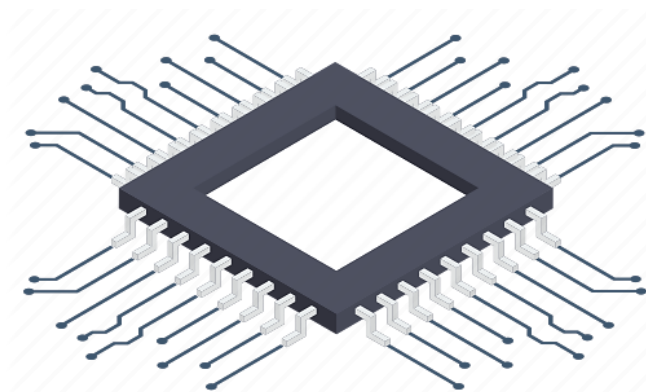


C208 – Arquitetura de Computadores

Capítulo 1 – Introdução à Arquitetura de Computadores (parte 1)

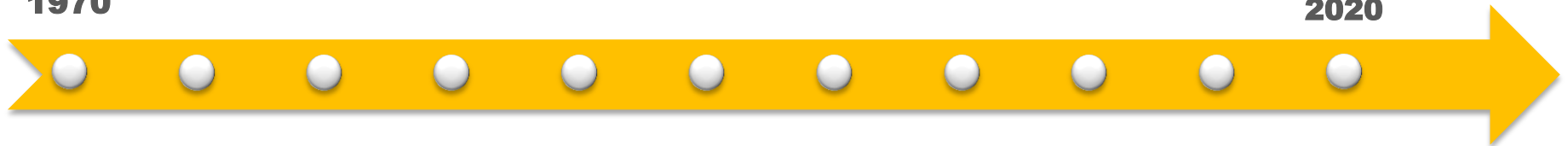


Prof. Yvo Marcelo Chiaradia Masselli



1970

2020



CAPÍTULO: 1

Introdução à Arquitetura de Computadores

- Linguagens, níveis e máquinas virtuais
- Máquinas multiníveis
- As gerações de computadores
- Tipos de Computadores (Descartáveis, Embarcados, Móveis, Pessoais, Servidores e Main-frames)
- **Arquiteturas CISC x RISC**
- **Processadores**
 - Organização da CPU
 - Estrutura Interna
 - Registradores
 - Processamento das Instruções
 - Conjunto de Instruções
 - O Caminho de Dados e de Controle
- **Paralelismo no nível de instrução (pipelining)**

Introdução à Arquitetura de Computadores

Computador

É um sistema de grande complexidade, podendo conter milhões de componentes diferentes se intercomunicando.

É um sistema hierárquico, constituído de um conjunto de subsistemas inter-relacionados, cada um contendo outros subsistemas.

A visão hierárquica facilita o projeto, pois o projetista pode concentrar-se em um nível de cada vez.

Possui uma arquitetura definida e organização de seus componentes.



Arquitetura de Computadores:

Refere-se ao comportamento **funcional de um computador**, do ponto de vista do programador;

- Descreve um nível de compatibilidade entre diferentes processadores de uma mesma linha/família a nível de instruções.

- Exemplos de Arquiteturas:

- IA-86, X-86
- IA-64, X-64
- MIPS
- SPARC
- VAX
- PowerPC
- ARMv5, ARMv6, ARMv7

Definição da **ISA** (*Instruction Set Architecture*)

- Instruções em Hardware (Assembly);
 - Formato de Instrução;
 - Representação numérica;
 - Tamanho das palavras;
 - Registradores disponíveis
 - Modos de endereçamento
- etc...

Termo “Arquitetura” – Introduzido pela IBM, Lyle R. Johnson, Muhammad Usman Khan e Frederick P. Brooks, Jr. Em 1959

Organização de Computadores:

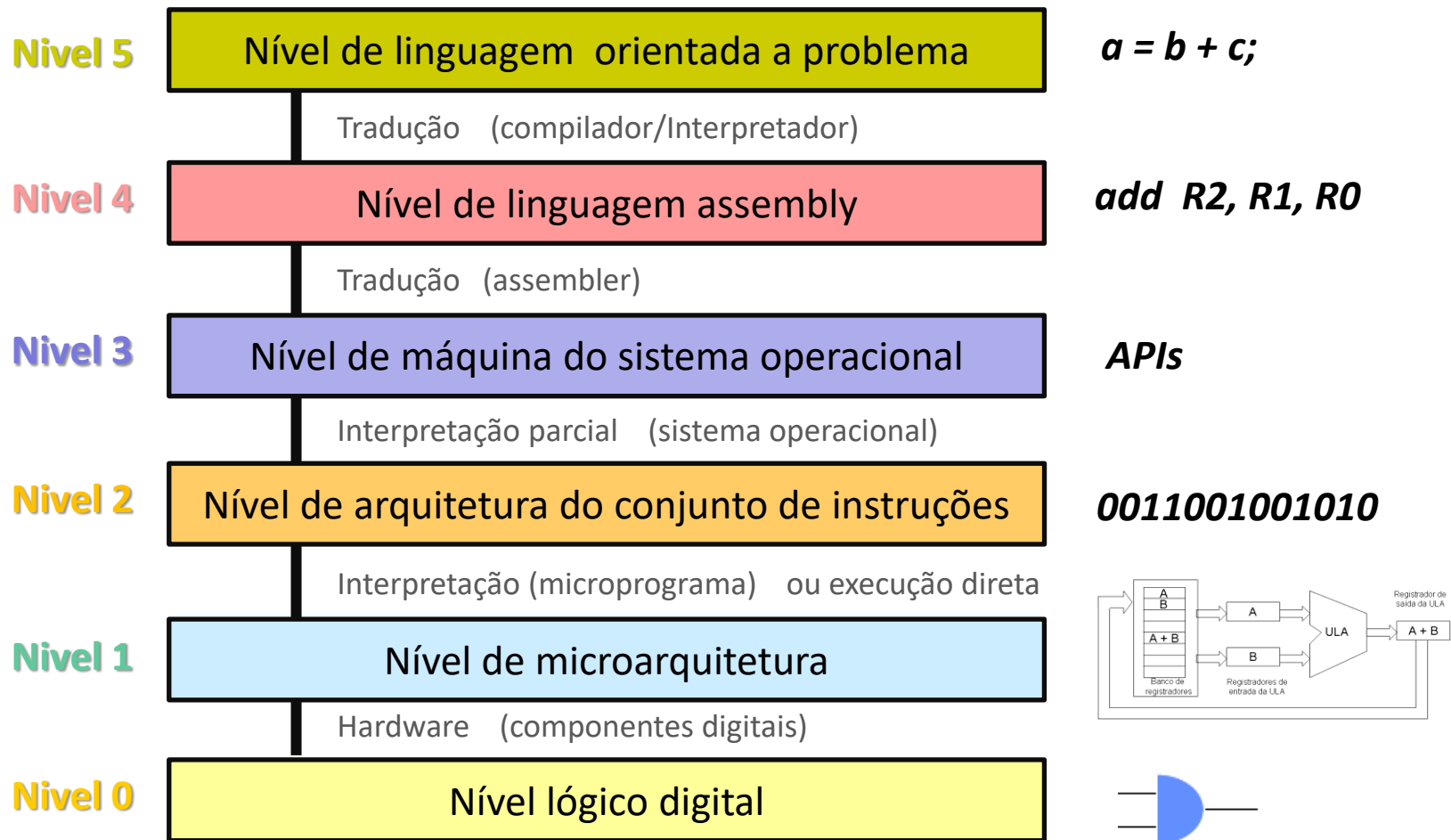
Refere-se aos aspectos estruturais que definem um processador, como:

- Circuitos digitais internos e suas interconexões.
- Tecnologias e capacidades das memórias;
- Implementa as especificações da **arquitetura**.

Detalhes de Hardware transparentes ao programador

- Sinais de Controle da CPU
- Interfaces com o dispositivos de Entrada/Saída
- Tecnologia de Memória utilizada (cache instruções/dados)
etc ...

Estrutura em Camadas – Máquinas Multiníveis



Níveis voltados aos programadores de sistemas:

- **Nível lógico digital (0):** nível formado por portas lógicas (ou *gates*).
- **Nível de microarquitetura (1):** formado por registradores e pela ULA (Unidade Lógica Aritmética). Responsável pelas operações lógicas e aritméticas.
- **Nível de arquitetura do conjunto de instruções (2):** também conhecido como nível **ISA (*Instruction Set Architecture*)**, responsável pela interpretação do conjunto de instruções de máquina através da execução do microprograma ou de forma direta pela UC (unidade de Controle) da CPU.
- **Nível de máquina do sistema operacional (3):** acrescenta novos recursos como uma organização de memória diferente, a capacidade de executar dois ou mais programas simultaneamente, entre outros.

Níveis voltados aos programadores de aplicações

(linguagens baseadas em palavras e abreviações)

- **Nível de linguagem de montagem (*assembly*) (4):** permite aos desenvolvedores escreverem programas para os níveis 1, 2 e 3, em linguagem de baixo nível (*assembly*), traduzidos pelo *assembler* (*montador*). Alta dependência e conhecimento do hardware (níveis 0 e 1).
- **Nível de Linguagem orientada a problema (5):** linguagens projetadas para serem usadas por programadores de aplicações (linguagens de alto nível). Estas linguagens são traduzidas para os níveis 3 e 4 através dos compiladores, ou as vezes são interpretadas.

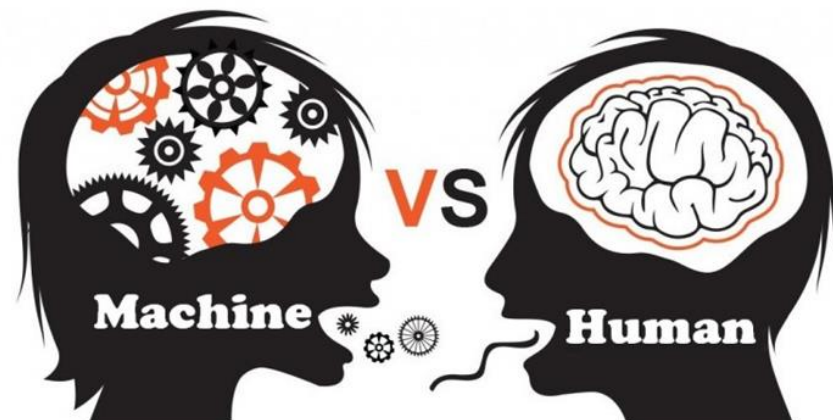
Linguagens, níveis e máquinas virtuais

Linguagem das pessoas \neq Linguagem dos computadores

É necessário um “tradutor” entre eles.

Tradução:

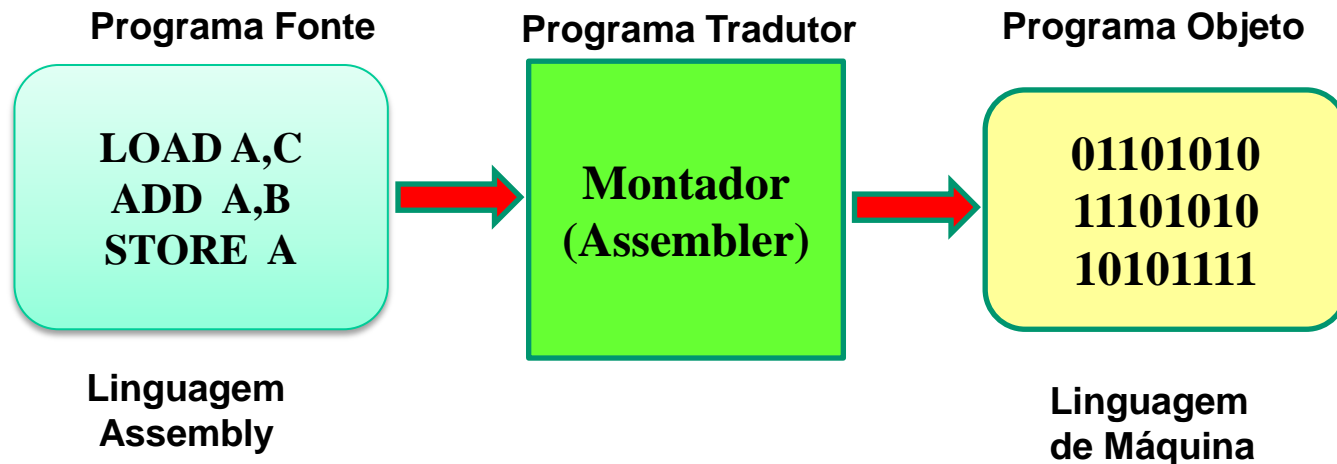
Montador
Compilador
Interpretador



Fonte: <http://www.tridindia.com/blog/industry-insights-to-human-translation-vs-machine-translation/>

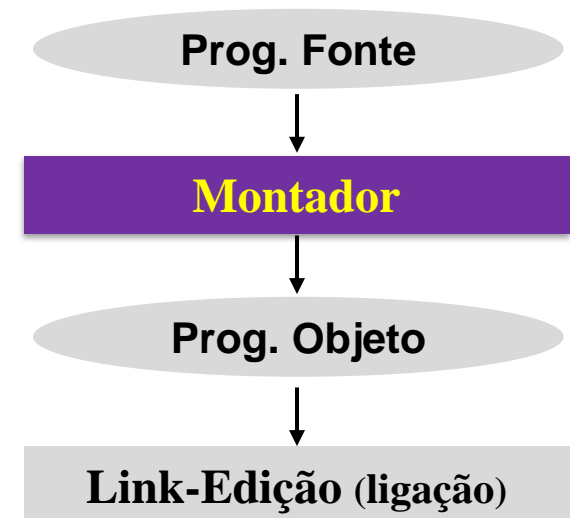
Linguagem de Baixo Nível (Assembly)

Símbolos (mnemônicos) são usados para representar um código de máquina que será gerado.



MONTADOR (ASSEMBLER):

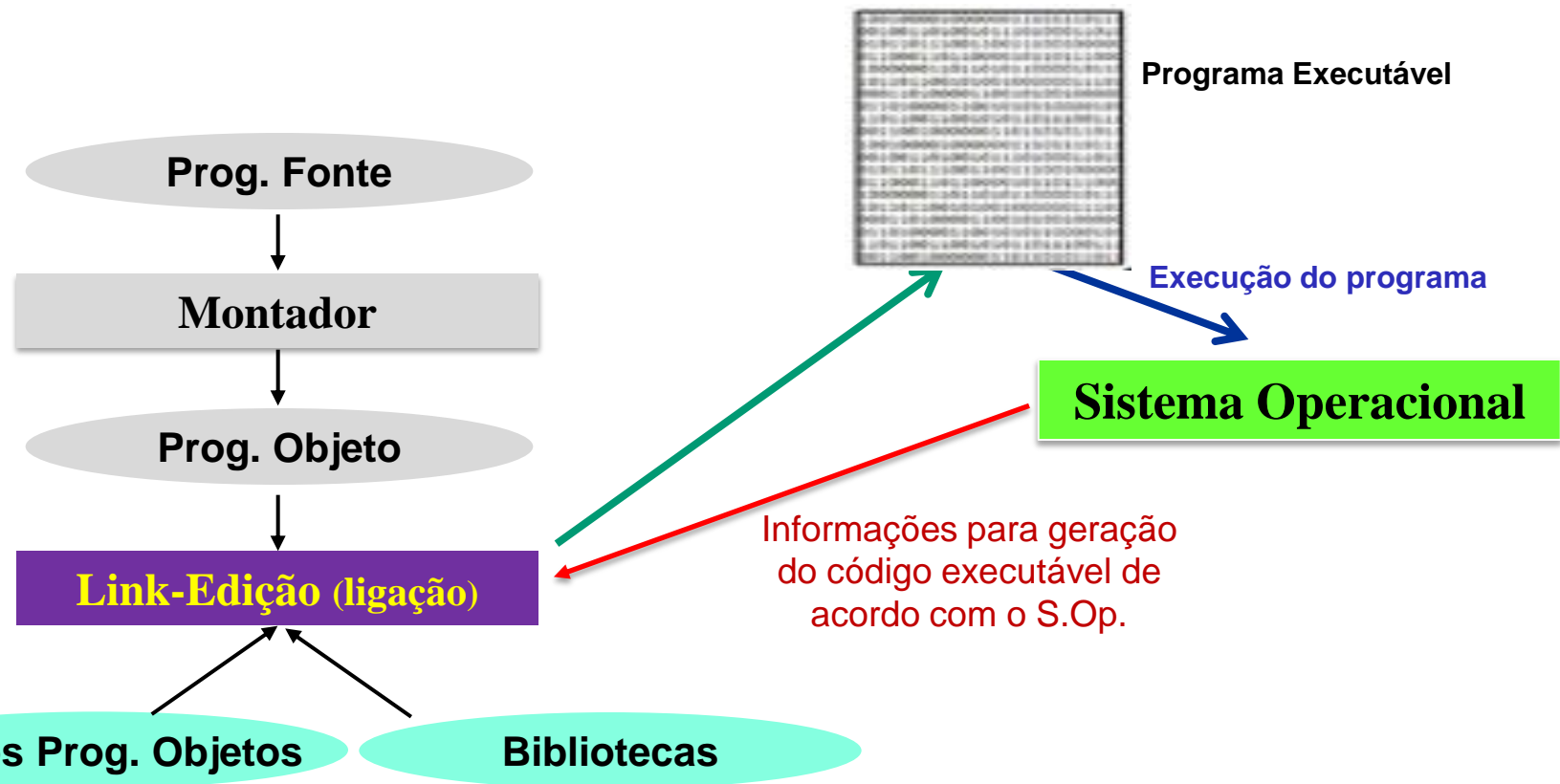
- É um programa tradutor, porém só é utilizado para traduzir um programa fonte em linguagem de baixo nível (**ASSEMBLY**) para o programa objeto.
- Este processo é bem simples, pois é uma tradução direta dos símbolos utilizados no ASSEMBLY (mnemônicos) para seus respectivos códigos de máquina (**Op-codes**).
- A linguagem ASSEMBLY é específica para cada processador, pois ela nada mais é do que a representação simbólica das instruções de máquina do processador.



LIGADOR (LINK-EDITOR):

- É o programa que transforma o código objeto gerado pelo compilador em código executável.
- Ele “liga” o código objeto do programa compilado com outros códigos objetos (biblioteca de funções, componentes, DLL's, chamadas das rotinas da BIOS, etc) e recursos do sistema operacional, resolvendo todas as referências simbólicas existentes entre os módulos objetos.
- Gera o programa executável com informações necessárias para o Sistema Operacional poder carregá-lo na memória para execução.

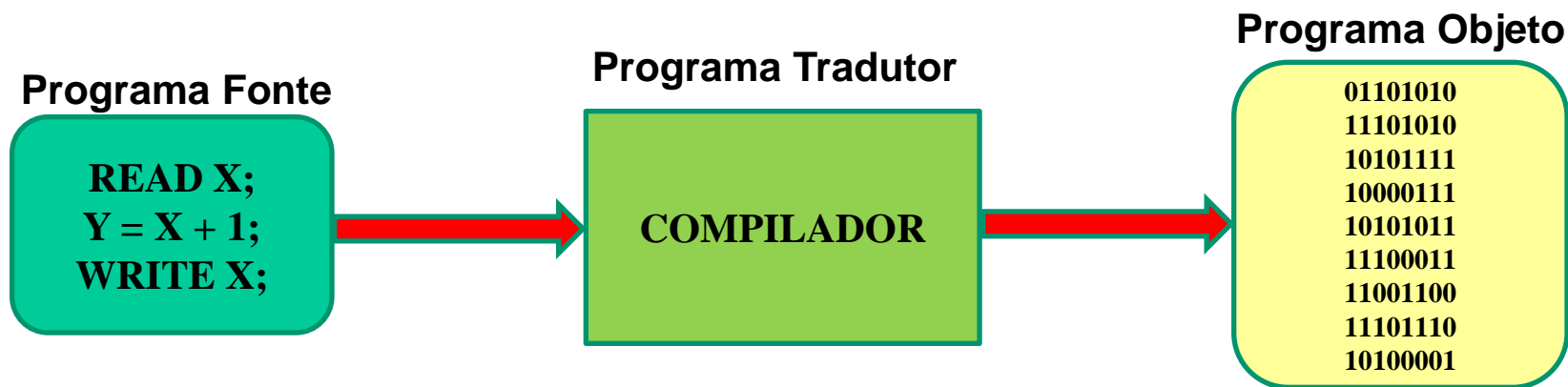
Ligação (LINK-EDIÇÃO) → processo para gerar o código executável para a máquina alvo.



Linguagens de Alto Nível

As linguagens de Alto Nível podem traduzidas para código objeto de forma COMPILADA ou INTERPRETADA.

- **Compilador** → traduz uma linguagem de alto nível (Programa Fonte), numa linguagem alvo ou objeto, que possa ser executada pelo computador.
- Código objeto gerado na máquina fonte for igual a máquina alvo → **Compilador Direto**
- Código objeto gerado na máquina fonte for diferente da máquina alvo → **Compilador Cruzado/**

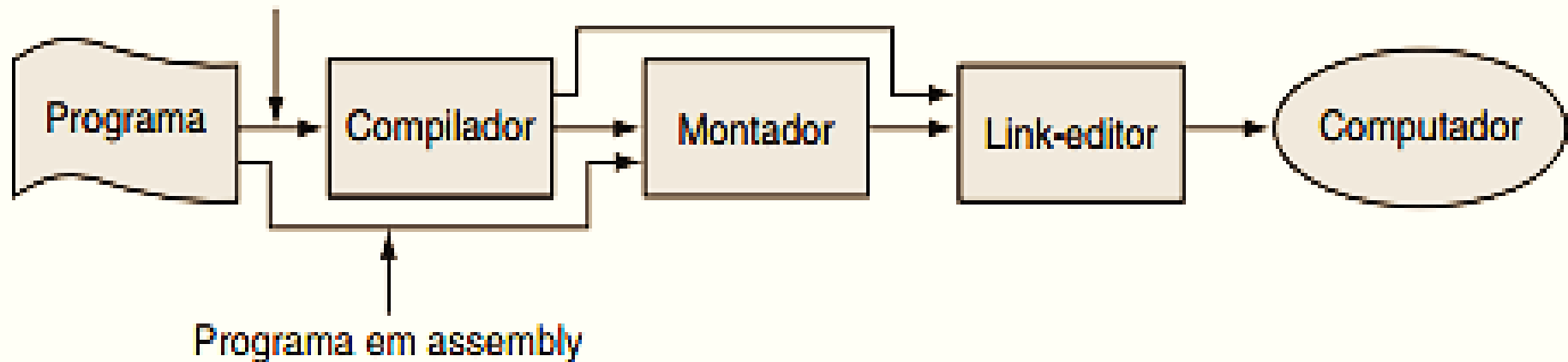


Linguagens de Alto Nível

Alguns compiladores produzem código Assembly que é passado para um montador para posterior processamento (Montagem e Link-edição).

Alguns compiladores produzem o trabalho dos montadores sem gerar o código Assembly;

Programa em linguagem de alto nível



Linguagens, níveis e máquinas virtuais

Linguagem de Alto Nível x Linguagem de Baixo Nível x Linguagem Máquina

```
#include <stdio.h>

int
main (int argc, char *argv [ ])
{
    int i;
    int sum =0;
    for (i =0; i <=100; i =i +1) sum =sum +i *i;
    printf ("The sum from 0 ..100 is %d \n",sum);
}
```

```
.text
.align    2
.globl    main

main:
    subu   $sp,$sp,32
    sw     $ra,20($sp)
    sd     $a0,32($sp)
    sw     $0,24($sp)
    sw     $0,28($sp)

loop:
    lw     $t6,28($sp)
    mul    $t7,$t6,$t6
    lw     $t8,24($sp)
    addu   $t9,$t8,$t7
    sw     $t9,24($sp)
    addu   $t0,$t6,1
    sw     $t0,28($sp)
    ble    $t0,100,loop
    la     $a0,str
    lw     $a1,24($sp)
    jal    printf
    move   $v0,$0
    lw     $ra,20($sp)
    addu   $sp,$sp,32
    jr     $ra

.data
.align    0

str:
.asciiz   "The sum from 0 ..100 is %d \n"
```

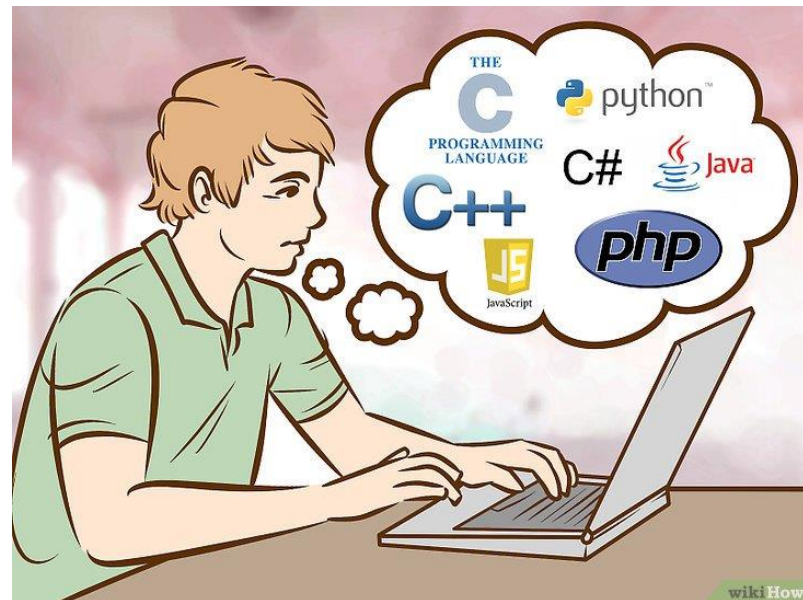
```
001001111011110111111111111100000
101011111011111100000000000010100
101011111010010000000000000100000
1010111110100101000000000000100100
101011111010000000000000000011000
101011111010000000000000000011100
100011111010111000000000000011100
100011111011100000000000000011000
000000011100111000000000000011001
001001011100100000000000000000001
001010010000000100000000001100101
101011111010100000000000000011100
00000000000000000000111100000010010
00000011000011111100100000100001
00010100001000001111111111110111
101011111011100100000000000011000
00111100000001000001000000000000
100011111010010100000000000011000
0000110000010000000000000011101100
001001001000010000000010000110000
100011111011111100000000000010100
00100111101111010000000000100000
00000011111000000000000000001000
000000000000000000001000000100001
```

Linguagens, níveis e máquinas virtuais

Linguagens de Alto Nível

Exemplos de Linguagens de Programação **Compiladas**:

- **FORTRAN** (1957)
- **ALGOL** (1958)
- **COBOL** (1959)
- **PASCAL** (1963)
- **BASIC** (1965)
- **ADA** (1968)
- **C** (1972)
- **C++** (1983)
- **Delphi** (1985)
- **C#** (2001)
- **Swift** (2014)



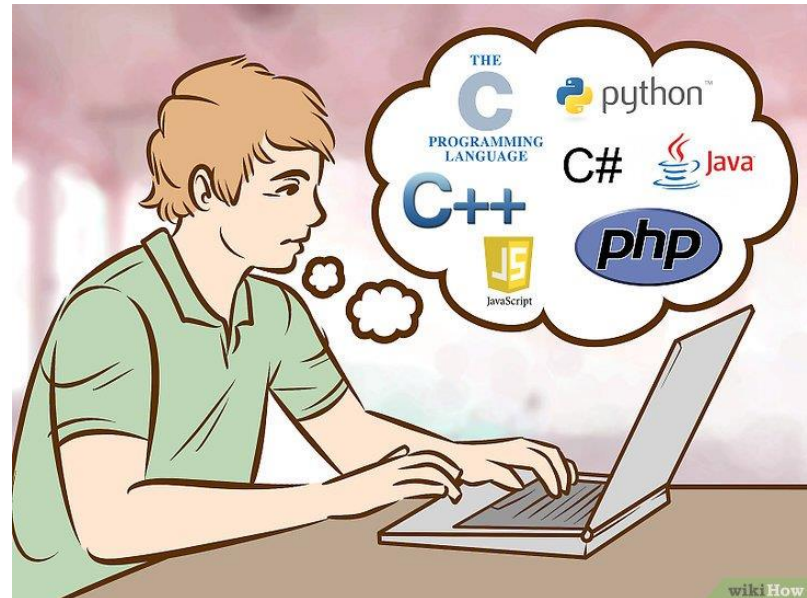
Linguagens de Alto Nível

- INTERPRETADOR → faz a tradução de comando por comando do programa fonte e o executa de imediato, sem gerar um programa objeto.
- A maior desvantagem na utilização de um interpretador é o tempo gasto na tradução das instruções de um programa toda vez que este for executado.
- A grande vantagem é a flexibilidade durante o desenvolvimento de uma Aplicação, já que o interpretador permite ao programador uma maior interação on-line com o código.

Linguagens de Alto Nível

Exemplos de Linguagens de Programação **Interpretadas**:

- ASP
- SMALLTALK
- BASIC
- LISP
- LOGO
- MUMPS
- HTML
- PHYTON
- entre outras ...



Linguagens, níveis e máquinas virtuais

Em linguagens compiladas, como a linguagem C por exemplo, o código é compilado para uma máquina de arquitetura específica, e só poderá ser executado em máquinas de mesma arquitetura e sistema operacional.

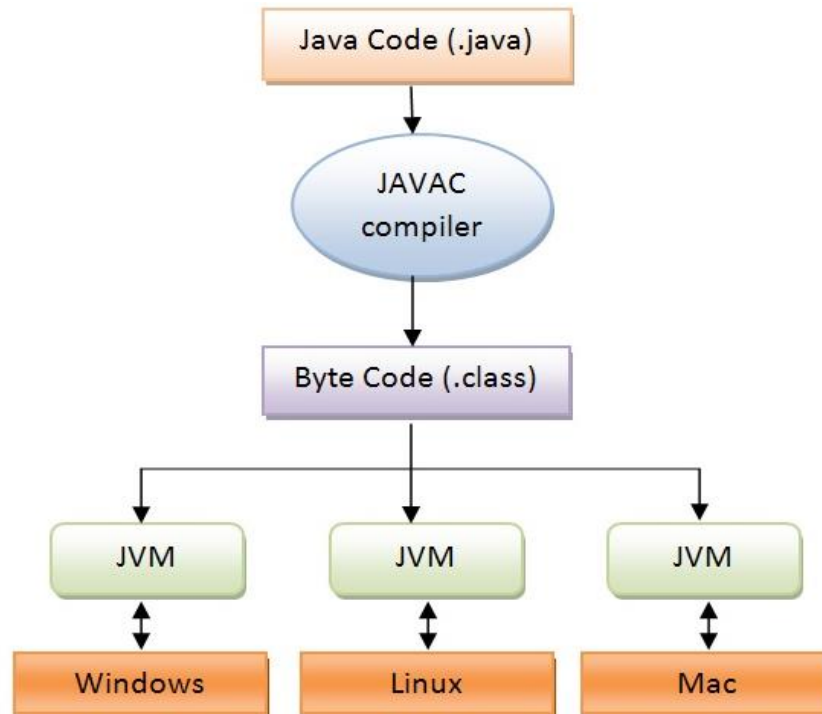
Para execução do código em máquinas com outra arquitetura e/ou outro sistema operacional, é necessário ajustar as bibliotecas de acordo com as necessidades e recompilar todo o código.

Maquina Virtual:

- É um software que **simula uma máquina física** e consegue executar vários programas, gerenciar processos, memória e arquivos.
- Constitui de uma plataforma, onde a **memória, o processador e seus outros recursos, são totalmente virtuais**, não dependendo de hardwares específicos.

Maquina Virtual:

Em um ambiente de máquina virtual, uma linguagem como Java por exemplo, passa por um processo de compilação (geração de ByteCodes) depois por uma interpretação de cada ByteCode em um interpretador nativo em cada SO, conhecido como JVM (Java Virtual Machine). Assim o mesmo código Java pode ser executado em máquinas com diferentes arquiteturas e/ou sistema operacional



Exercícios:

- 1 – Diferencie Organização de computadores e Arquitetura de Computadores.
- 2 – Qual a diferença entre Código Objeto e Código Executável ?
- 3 – Quem transforma o Código Fonte em Código Objeto ?
- 4 – Quem transforma o Código Objeto em Código Executável ?
- 5 – Por que o Código Objeto gerado pelo tradutor ainda não pode ser executado ?
- 6 – Por que a execução de programas interpretados é mais lenta que as de programas compilados ?
- 7 – Quais as funções de um LIGADOR (Link-Editor) ?
- 8 – Qual a diferença entre compilador Direto e Cruzado ?
- 9 – O que é uma máquina virtual?

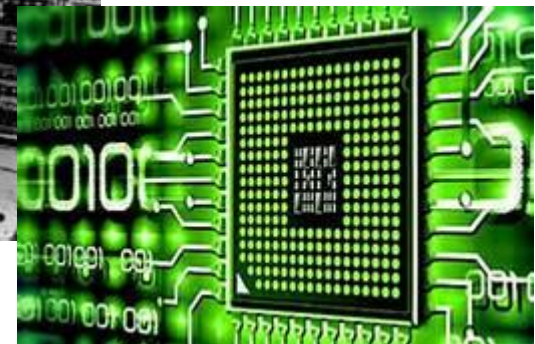
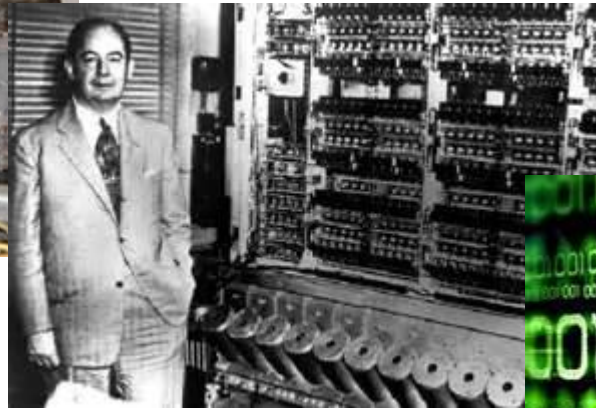
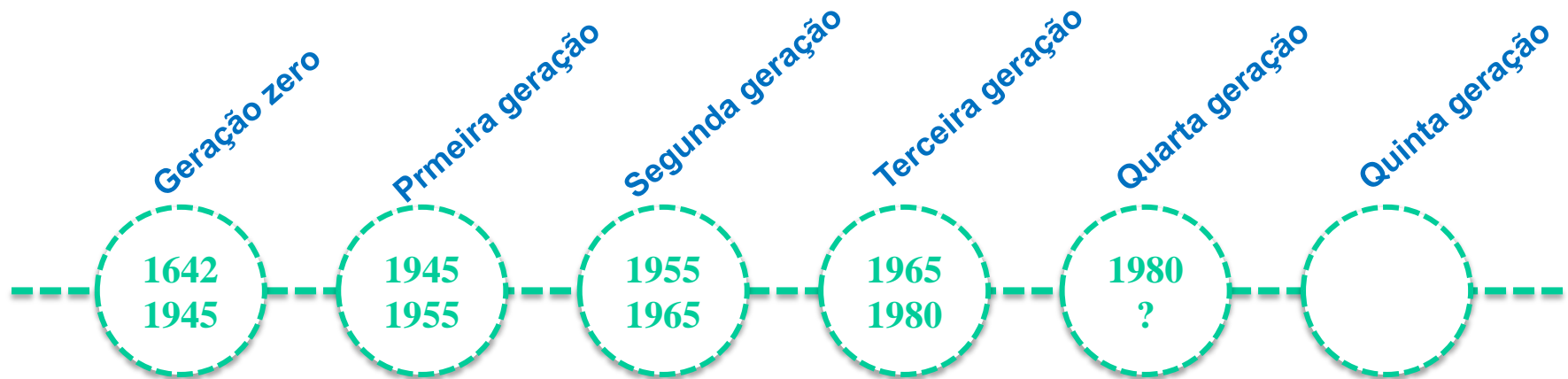
10 – Supondo dois computadores diferentes em arquitetura e SO, (um Mac/OS e um PC/Windows). Nos dois existe um compilador C. Um programador fez um programa fonte em linguagem C para um problema específico. Pergunta-se:

- a) O programa fonte pode ser o mesmo nos dois computadores ? Justifique.
- b) O programa Objeto gerado pelos respectivos compiladores serão iguais ? Justifique.
- c) O programa executável gerado em um dos computadores, se copiado para o outro, será executado normalmente? Justifique.
- d) O programa do compilador (executável) pode ser o mesmo nos dois computadores? Justifique.
- e) Se o programa fosse agora desenvolvido em Java, o programa poderia ser o mesmo para os dois computadores?

11- Faça a mesma análise da questão anterior, supondo que os dois computadores possuem a mesma arquitetura (PC) porém, um usando Windows e o outro usando o Linux. Analise também se o programa fosse feito em java para este caso.

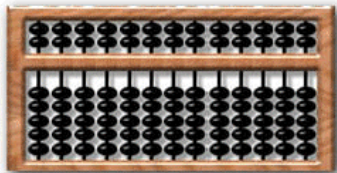
As Gerações de Arquitetura de Computadores

Marcos das Arquiteturas de Computadores

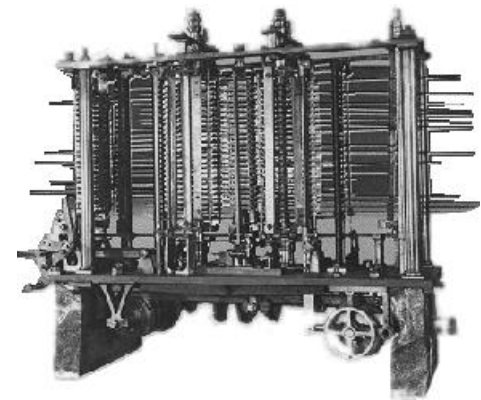
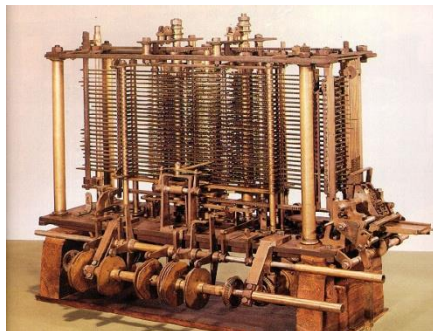
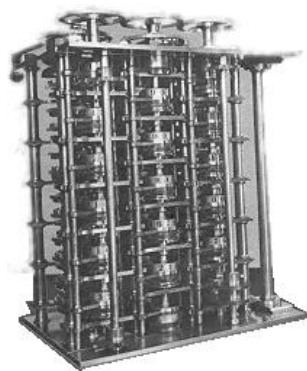
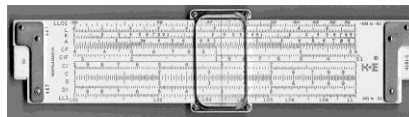


1642/1945 – Geração Zero – Computadores Mecânicos

- São considerados precursores dos computadores todos os instrumentos ou máquinas que contribuíram com ideias para a criação dos mesmos. Dentre eles, o surgimento de uma máquina programável foi um grande marco na história dos computadores.



Ábaco Chinês



1945/1955 - Computadores de Primeira Geração - Válvulas:

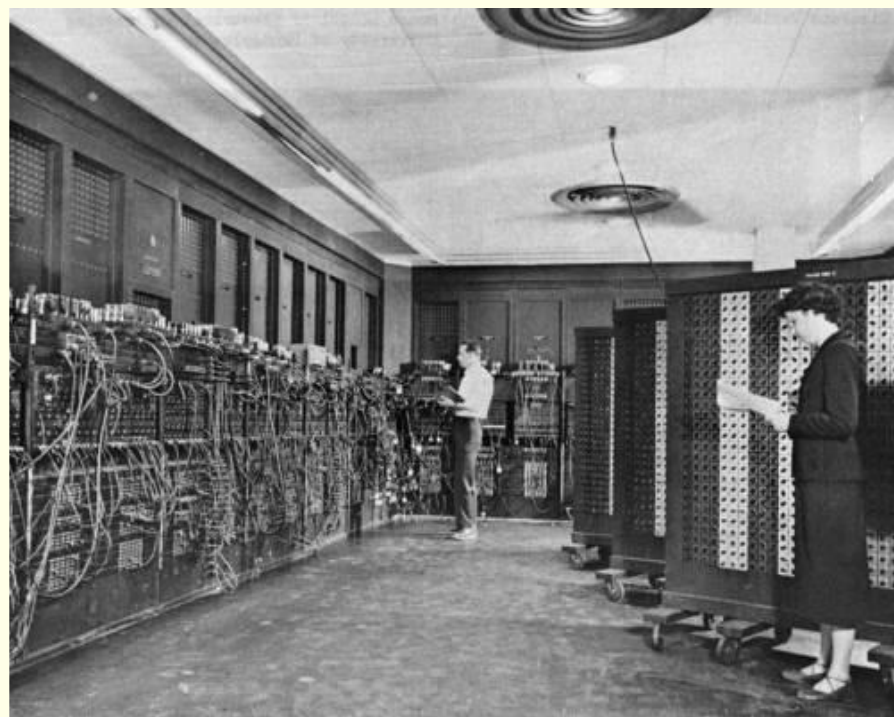
- Circuitos eletrônicos a válvulas (muito aquecimento, maior tamanho e peso)



- Uso restrito (somente por pessoal técnico especializado)
- Precisava ser reprogramado a cada tarefa (monotarefa)
- Grande consumo de energia
- Normalmente, as válvulas quebravam/queimavam após algumas horas de uso e tinham o processamento bastante lento.

1945/1955 - Computadores de Primeira Geração - Válvulas:

- Nesta geração, os computadores calculavam com uma velocidade de milésimos de segundo e eram programados em linguagem de máquina (0s e 1s).



ENIAC (Electronic Numerical Integrator And Computer)

- Peso: 33 toneladas
- Tamanho: área de 180 m²
- Custo: US\$ 500 mil (na época)
- 18 mil válvulas
- consumo: 200KWatts/hora
- Algumas operações por segundo
(capacidade de processamento de uma calculadora de bolso moderna)

1955/1965 - Computadores de Segunda Geração - Transistores:

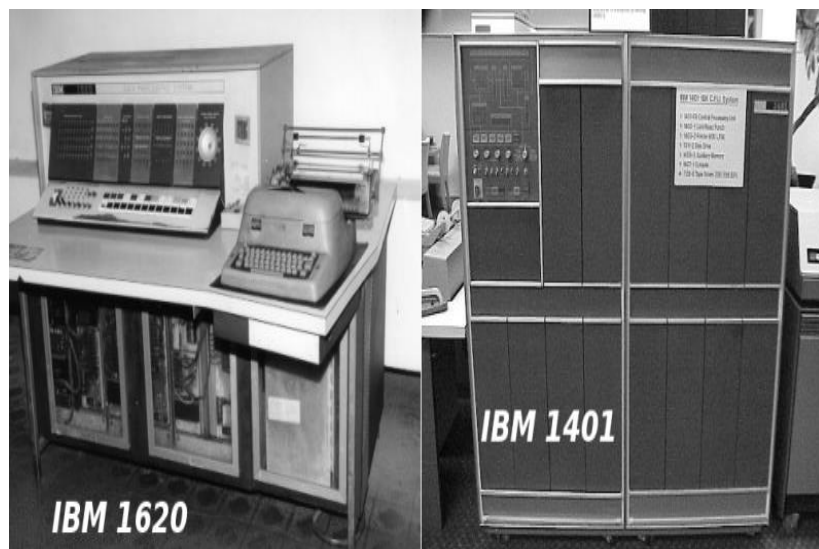
- Início do uso comercial dos computadores
- Uso de transistores em substituição às válvulas
- Redução de tamanho, peso e custo



- Tamanho de 100 vezes menor que o da válvula, não precisa de tempo para aquecimento, consome menos energia, é mais rápido e confiável.

1955/1965 - Computadores de Segunda Geração - Transistores:

- Os computadores desta geração já calculavam em microssegundos e eram programados em linguagem de montagem (símbolos/mnemônicos).



SYSTEM/360



Foi o primeiro computador de uso comercial do mundo.

Pesando “apenas” 400 Kg, sua configuração original vinha com 7 Mbytes de memória secundária e 128 Kbytes de memória RAM.

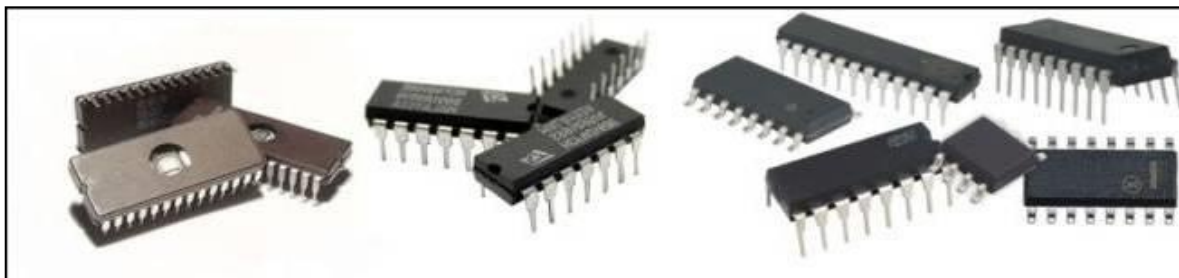
1965/1980 - Computadores de Terceira Geração - Circuitos Integrados:

- Surgem os CIs (circuitos integrados) SSI, MSI e LSI

Escala de integração de circuitos integrados

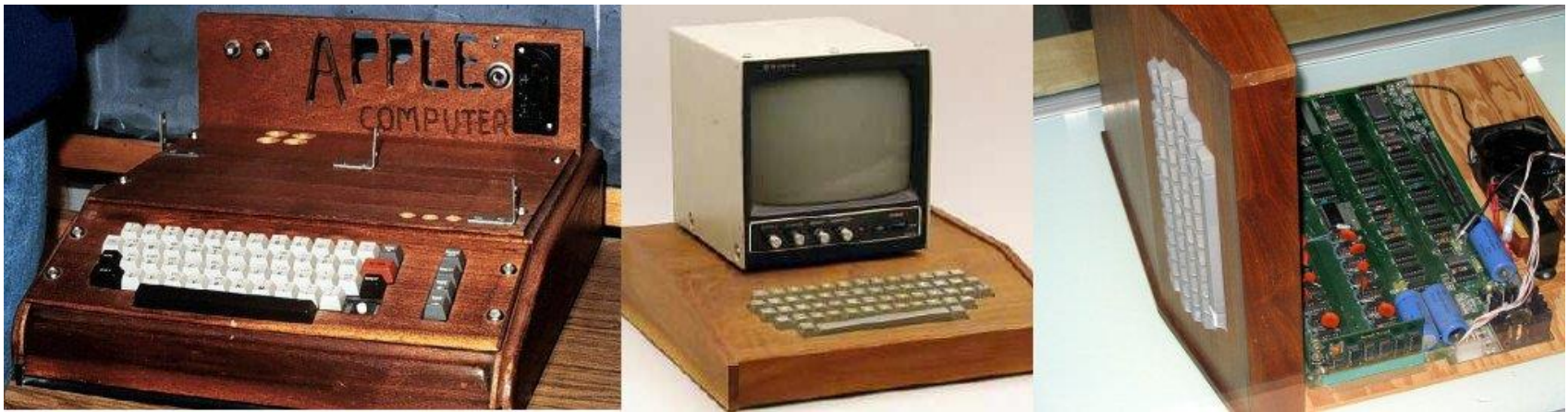
Abrev.	Denominação	Complexidade (números de transístores)		
		Interpretação comum	Tanenbaum ^[7]	Texas Instruments ^[8]
SSI	Small Scale Integration	10	1–10	em baixo de 12
MSI	Medium Scale Integration	100	10–100	12–99
LSI	Large Scale Integration	1.000	100–100.000	100–999

- Diminuição do tamanho e maior capacidade de processamento
- Início da utilização dos microcomputadores pessoais



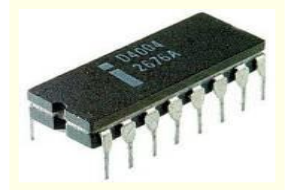
1965/1980 - Computadores de Terceira Geração - Circuitos Integrados:

- Os computadores com o CI são muito mais confiáveis, bem menores, tornando os equipamentos mais compactos e rápidos possuem baixíssimo consumo de energia e menor custo.
- Nesta geração surgiram as linguagens de alto nível, orientada para as aplicações.



1965/1980 - Computadores de Terceira Geração - Circuitos Integrados:

- Em 1971 surge o microprocessador INTEL 4004
- Foi o primeiro microprocessador do mundo, inicialmente criado para operar uma calculadora da empresa japonesa Busicom.
- Contudo, os dirigentes da Intel perceberam que aquele microprocessador poderia realizar muito mais funções, e empregaram uma evolução do 4004, o microprocessador 8080, para o desenvolvimento do primeiro computador pessoal, o ALTAIR 8800.



Microcomputador ALTAIR, com processador Intel 8080, **40 vezes** mais rápido que o ENIAC.

1965/1980 - Computadores de Terceira Geração - Circuitos Integrados:

- Em 1977 surge o Apple II
 - A *Apple Computers* lança, em 1977, o primeiro computador doméstico de sucesso: o Apple II, sucessor do menos afortunado Apple I.

Parecendo mais um eletrodoméstico que um computador, o Apple II deu início a revolução dos Computadores pessoais.



1965/1980 - Computadores de Terceira Geração - Circuitos Integrados:

- Em 1978: A família PC (**arquitetura x86**)
 - Também desenvolvido pela pioneira Intel, a evolução do 8080, o **Intel 8086**, deu origem a uma das mais populares linhas de computadores domésticos, os x86, empregados nos primeiros PCs da IBM.
 - Foram os chips da linha X86 que popularizaram os computadores domésticos como o IBM PC-XT e depois seu sucessor o IBM PC-AT.



PC-XT 4,7MHz, 64KB RAM



PC-AT 8MHz, 64KB RAM

1980 / ??? – Quarta Geração - CIs com alta escala de integração:

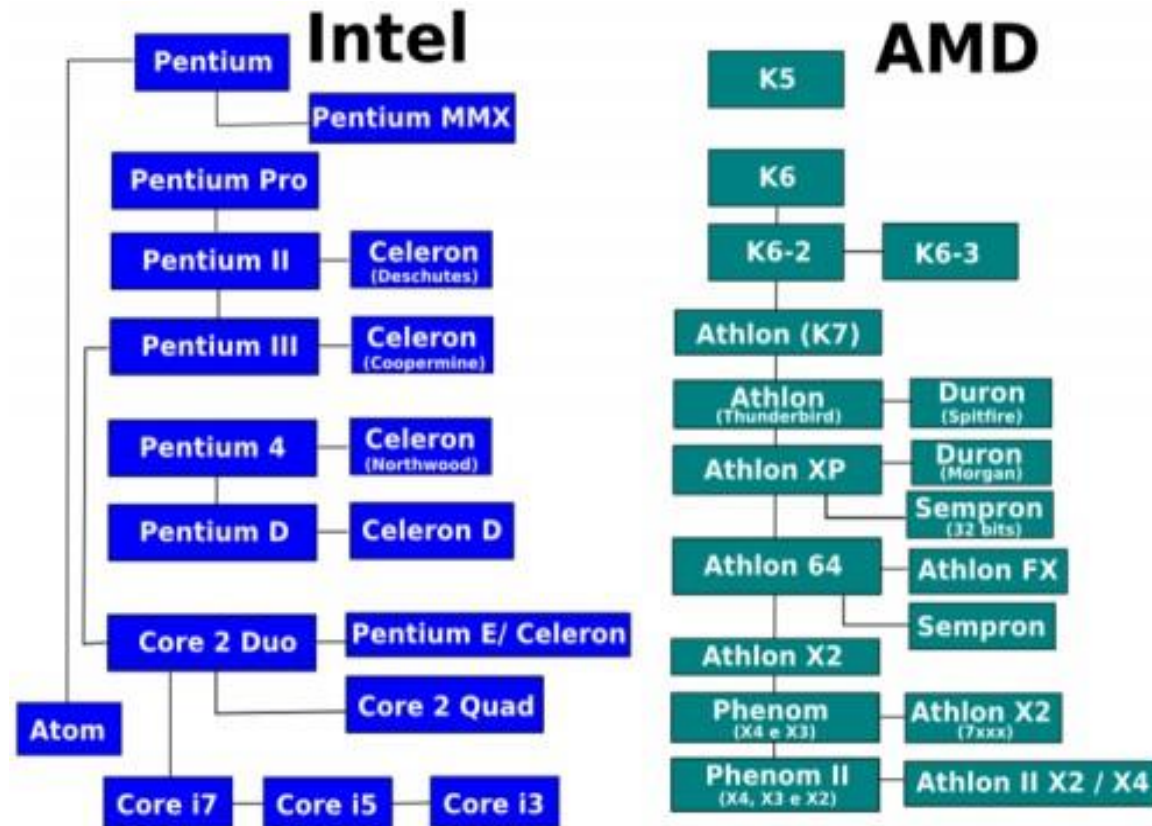
- Surgiram os microprocessadores, os microcomputadores e os supercomputadores.
- O processo de miniaturização continuou e surgiram escalas de integração de circuito em grau muito mais elevado (VLSI, ULSI e SLSI)

Escala de integração de circuitos integrados

Abrev.	Denominação	Complexidade (números de transístores)		
		Interpretação comum	Tanenbaum ^[7]	Texas Instruments ^[8]
SSI	Small Scale Integration	10	1–10	em baixo de 12
MSI	Medium Scale Integration	100	10–100	12–99
LSI	Large Scale Integration	1.000	100–100.000	100–999
VLSI	Very Large Scale Integration	10.000–100.000	a partir de 100.000	ab 1.000
ULSI	Ultra Large Scale Integration	100.000–1.000.000	—	—
SLSI	Super Large Scale Integration	1.000.000–10.000.000	—	—

1980 / ??? – Quarta Geração - CIs com alta escala de integração:

Houve uma explosão no mercado de microprocessadores, sendo fabricados em escala comercial e a partir daí a evolução foi sendo cada vez maior, até chegar aos microprocessadores atuais.



Quinta Geração - Computadores de Baixa Potência e Invisíveis:

- É mais uma mudança de paradigma do que de arquitetura
- Dispositivos muito pequenos (PDA, Smartphones, Tablets, etc)
- Interconexão de todos os dispositivos (Redes)
- Virtualização de Computadores
- Nuvem de Computadores
- Inteligência Artificial



PDA, Iphone e Surface

Curiosidades na Evolução dos Computadores

Os “disquetes” ou discos flexíveis tiveram diferentes tamanhos e formatos desde que foram inventados.



Tipo de disco	Ano	Capacidade
8-inch	1971	80 kB
8-inch	1973	256 kB
8-inch	1974	800 kB
8-inch dual-sided	1975	1MB
5¼-inch	1976	160 kB
5¼-inch DD	1978	360 kB
5¼-inch QD	1984	1.2 MB
3-inch	1984?	320 kB
3½-inch	1984	720 kB
3½-inch HD	1987	1.44 MB
3½-inch ED	1991	2.88 MB

Curiosidades na Evolução dos Computadores

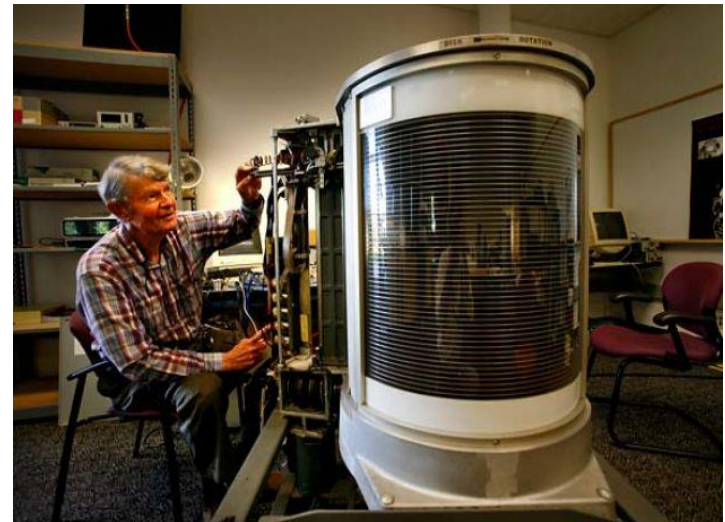
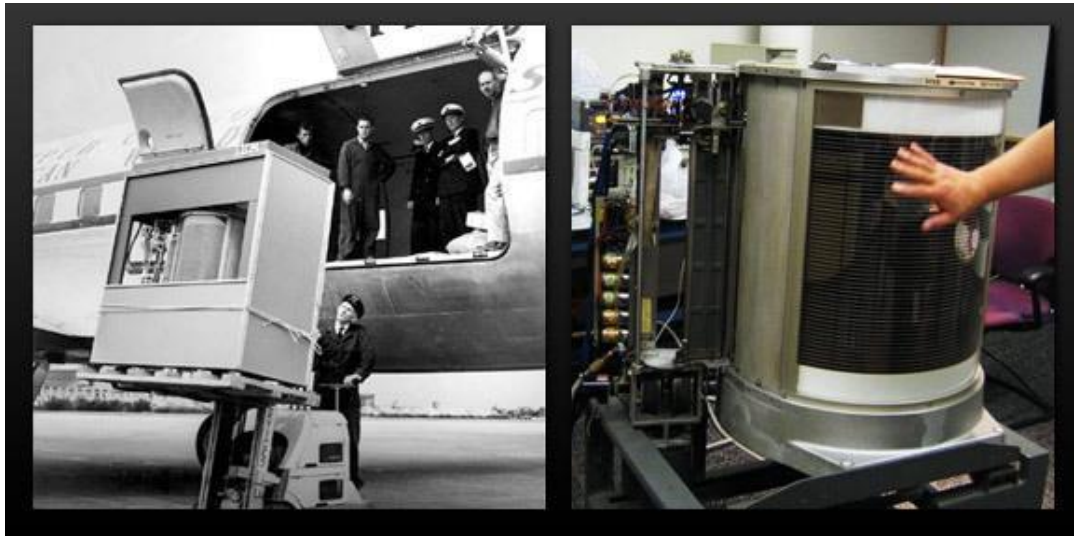
Em 35 anos mudamos para um novo tipo de armazenamento de dados



Curiosidades na Evolução dos Computadores

O primeiro disco rígido foi construído pela *IBM* em 1957, e era formado por nada menos que 50 discos de 24 polegadas de diâmetro, com uma capacidade total de 5 megabytes, incrível para a época.

Este primeiro disco rígido foi chamado de *305 RAMAC (Random Access Method of Accounting and Control)*.

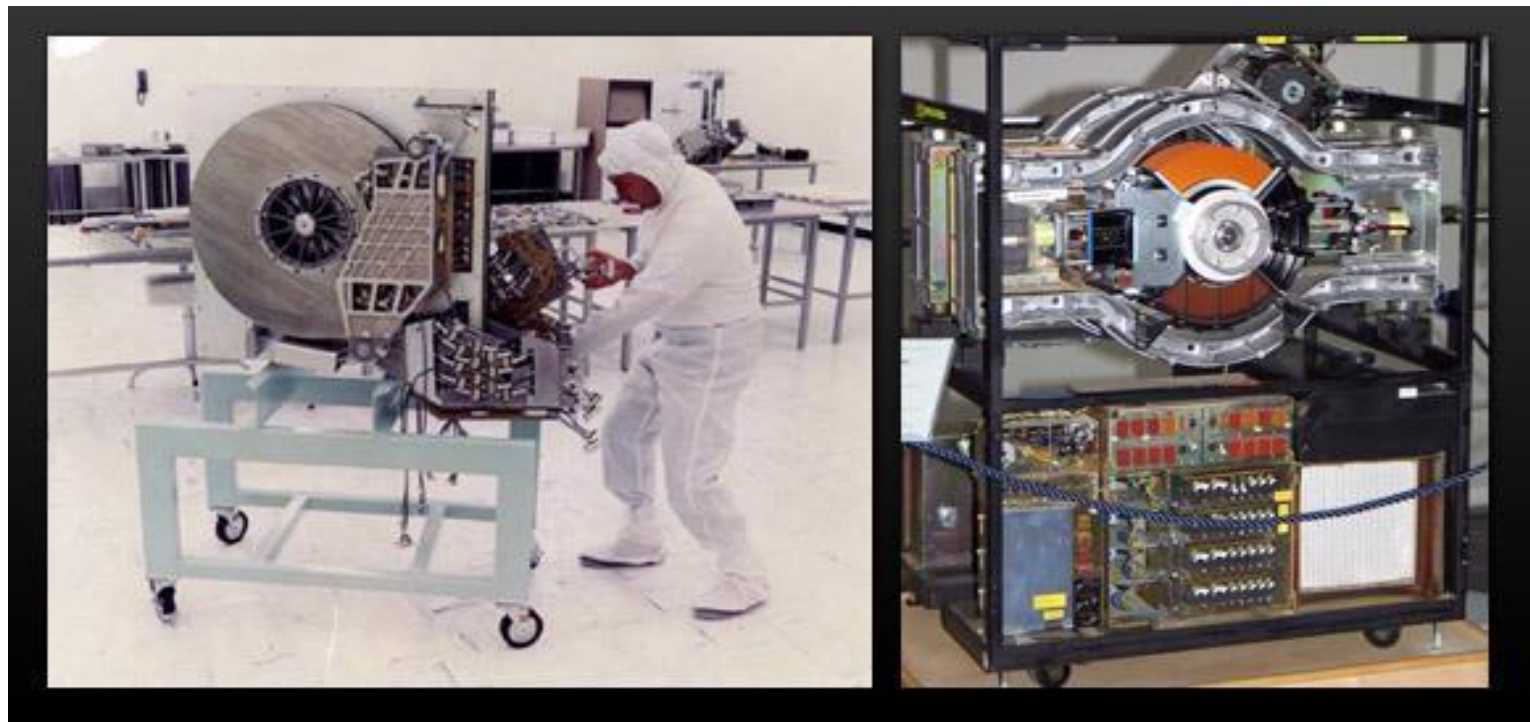


Chronicle / Michael Maloney

Curiosidades na Evolução dos Computadores

Em 1960 surgiu o primeiro disco rígido (HD) de alta capacidade.

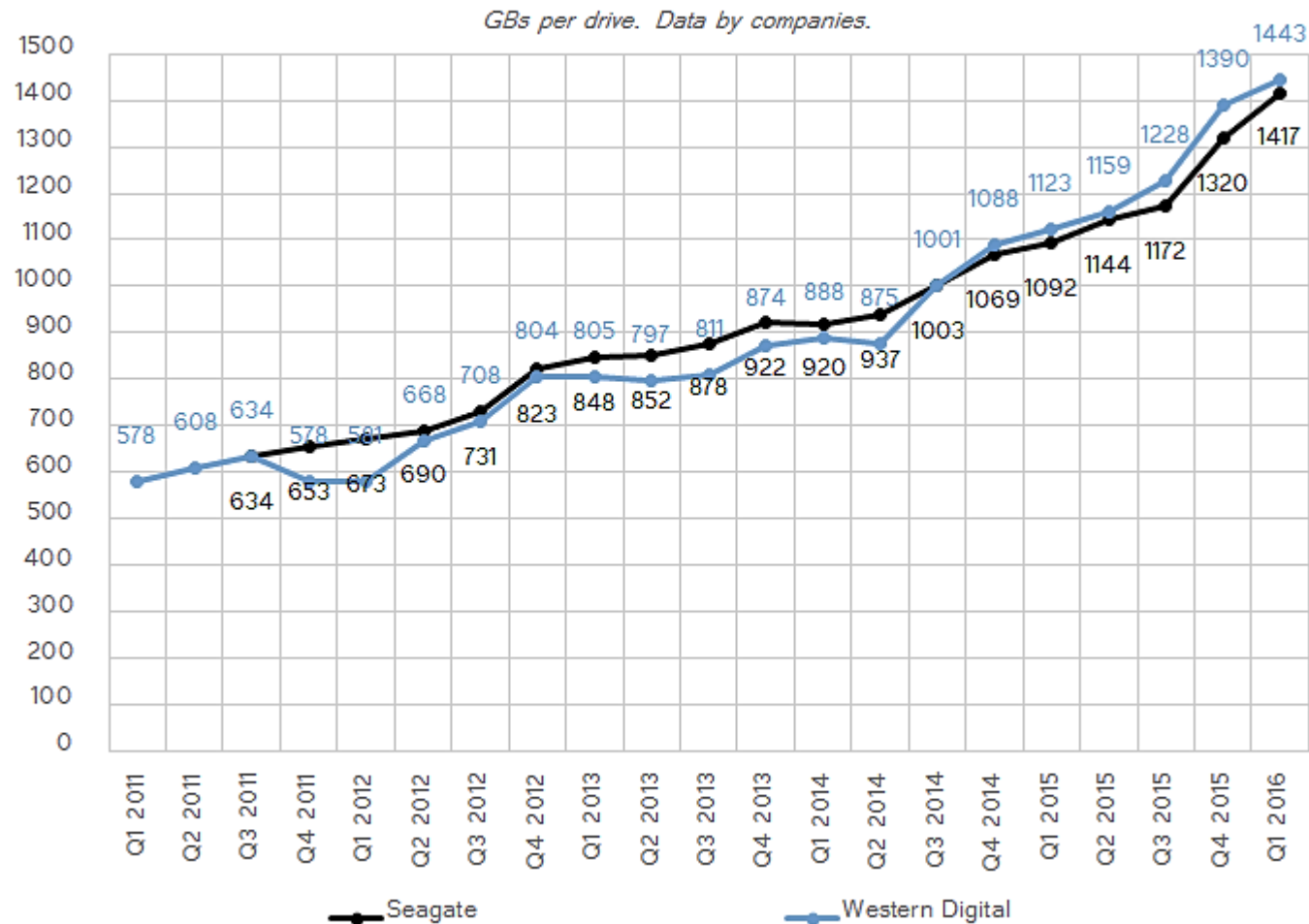
Era de 1GBytes, pesava 250Kg e custava US\$ 81.000,00



Curiosidades na Evolução dos Computadores

A evolução dos discos rígidos (HDs)

Average Capacities of HDDs by Seagate and Western Digital

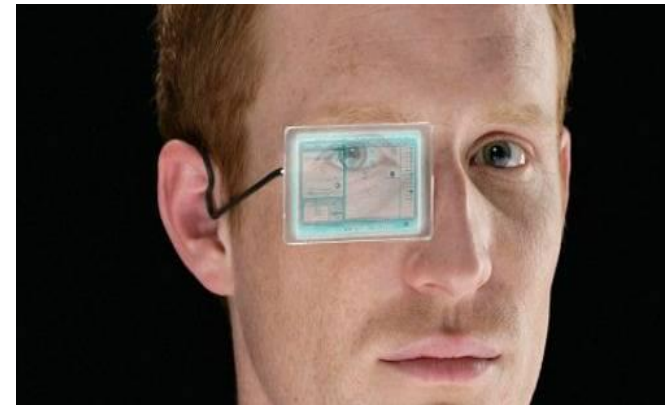


Curiosidades – E o futuro?

Processadores quânticos?

Processadores biológicos?

Tecnologias como vidro inteligente, e-paper e a nanotecnologia provavelmente revolucionarão novamente nossa interação com os computadores.



Caneta com 1.3GB de memória que grava o que voce escreve (Nokia)



Telas Aderentes



standard



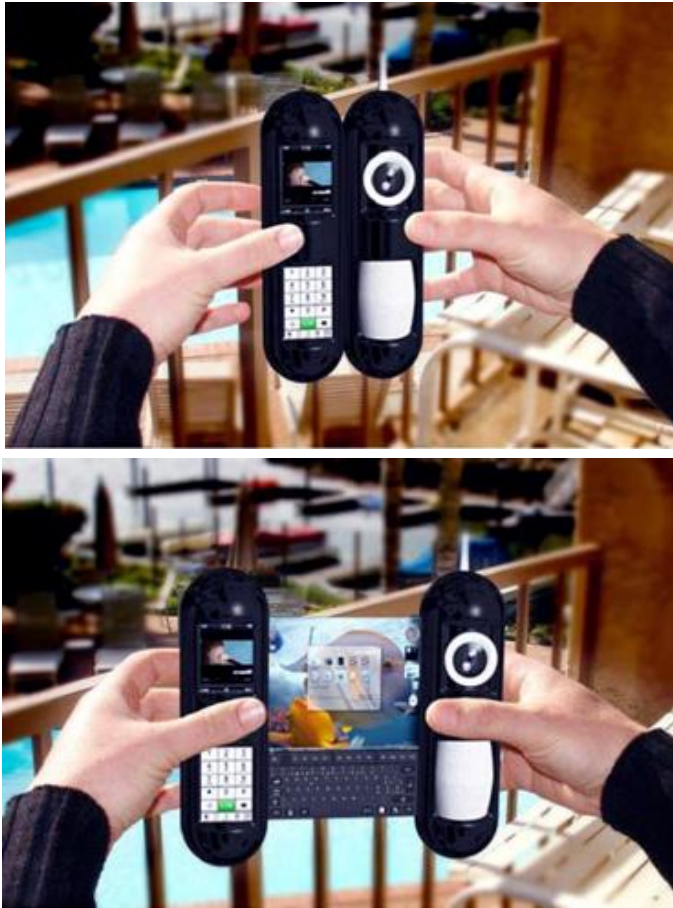
16:9



4:3



Celular/SmartPhone com tela expansível



Celular dobrável (SAMSUNG)

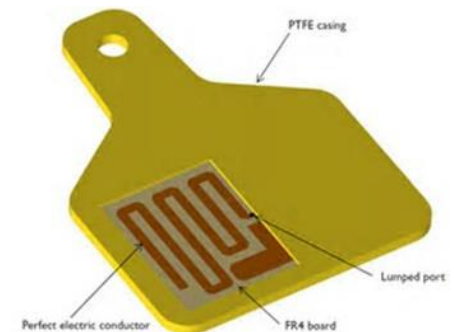


Tipos de Computadores

Esta parte do material cedido pelo prof. Gabriel Dias Scarpioni (2019)

DESCARTÁVEIS:

- Baseados em tecnologia RFID (*Radio Frequency Identification*)
 - Contém um minúsculo *transponder* e um número único de 128 bits embutido
 - Algumas aplicações:
 - Substituição aos códigos de barras
 - Rastreamento de veículos (Sem Parar)
 - Transporte de bagagens
 - Identificação de valores
 - Identificação de animais



EMBARCADOS:

- Computadores ou sistemas embarcados

Pequenos computadores compostos de processador, memória e capacidade de E/S

- Algumas aplicações:

- Eletrodomésticos
- Aparelhos de comunicação
- Periféricos de computadores
- Equipamentos de entretenimento
- Aparelhos de reprodução de imagem
- Equipamentos médicos
- Etc.



MOVEIS e de JOGOS:

- Computadores normais, mas ...
 - Recursos gráficos especiais
 - Capacidade de som
 - Software limitado
 - Pouca extensão
 - Baixo consumo (móveis)
 - Algumas aplicações:
 - Videogames
 - Smartphones



PESSOAIS:

- Abrangem tanto os desktops quanto os notebooks
 - Gigabytes de memória
 - HDs na ordem de terabytes
 - Placa de som
 - Interface de rede
 - Monitor
 - Outros periféricos



SERVIDORES:

- Computadores com alta capacidade
 - Processamento, memória, armazenamento
 - Aplicações:
 - *Clusters (data centers)*



Mainframes:

- Computadores do tamanho de uma sala (volta ao passado)
 - Maior capacidade de armazenamento (Tbytes)
 - Maior desempenho de processamento (TFLOPS)
 - Empresas evitam migrar o software antigo (COBOL) para novos servidores



Tipo	Preço (US\$)	Exemplo de Aplicação
Computador descartável	0,5	Cartões de identificação
Sistemas embarcados (microcontroladores)	5	Relógios, carros, eletrodomésticos
Computador móvel e de jogos	50	Videogames domésticos e smartphones
Computador pessoal	500	Computador de desktop ou notebook
Servidor	5K	Servidor de rede
Mainframe	5M	Processamento de dados em bloco em um banco

Exercícios:

- 1) Faça uma pesquisa dos maiores fabricantes de Microprocessadores e cite alguns modelos de cada um deles
- 2) Faça uma pesquisa dos “top” 5 supercomputadores existentes atualmente. Liste as características deles como desempenho, memória, tamanho e peso.
- 3) Faça uma pesquisa dos microprocessadores com arquitetura IA-32 e IA-64 listando os modelos, velocidade, tamanho de barramento, memória cache, etc.
- 4) Faça uma pesquisa semelhante ao item 3 para os processadores ARM.