

Arquitetura e Organização de Computadores

Abstrações e Tecnologia Computacionais

Prof. Me Rodrigo Vilela da Rocha

2022



Introdução

- Avanços verificados na tecnologia de Hardware têm permitido que especialistas criem programas extremamente úteis
 - ... quase onipresença dos computadores nas atividades desenvolvidas pelas pessoas.
- Ficção científica nos dias de hoje:
 - Andar sem dinheiro em espécie;
 - Fazer compras sem sair de casa;

Introdução

- Bons programadores:
 - Sempre preocupados com a performance
 - Em 60 e 70 → restrição era tamanho de memória
 - Atualmente com os avanços ...
 - nas técnicas de projeto dos processadores
 - na tecnologia de fabricação das memórias
 - Programadores agora devem se preocupar com a natureza hierárquica dos sistemas de memórias associada ao paralelismo dos processadores
 - Devem ampliar bastante os seus conhecimentos sobre organização dos computadores



Introdução

- Não se trata de um campo do conhecimento que prime pela monotonia
- Essa corrida em direção à inovação tem levado a um progresso sem precedentes desde o aparecimento do primeiro computador
- Comparação:
 - Indústria de transportes: Atravessar a América do Sul em 5 segundos pagando R\$ 0,50

Introdução

- A revolução dos computadores não para
 - Custo decresce → oportunidades do uso dos computadores se multiplicam

Arquitetura vs. Organização

Estrutura e Função

- Um computador é um sistema de grande complexidade
 - computadores modernos contém milhões de componentes eletrônicos elementares
- Como é possível decrivê-los?
 - Reconhecimento da natureza hierárquica
 - Em cada nível, o sistema consiste em um conjunto de componentes e de relacionamentos entre estes
- Em cada nível deve ser considerado a estrutura e o funcionamento

Estrutura e Função

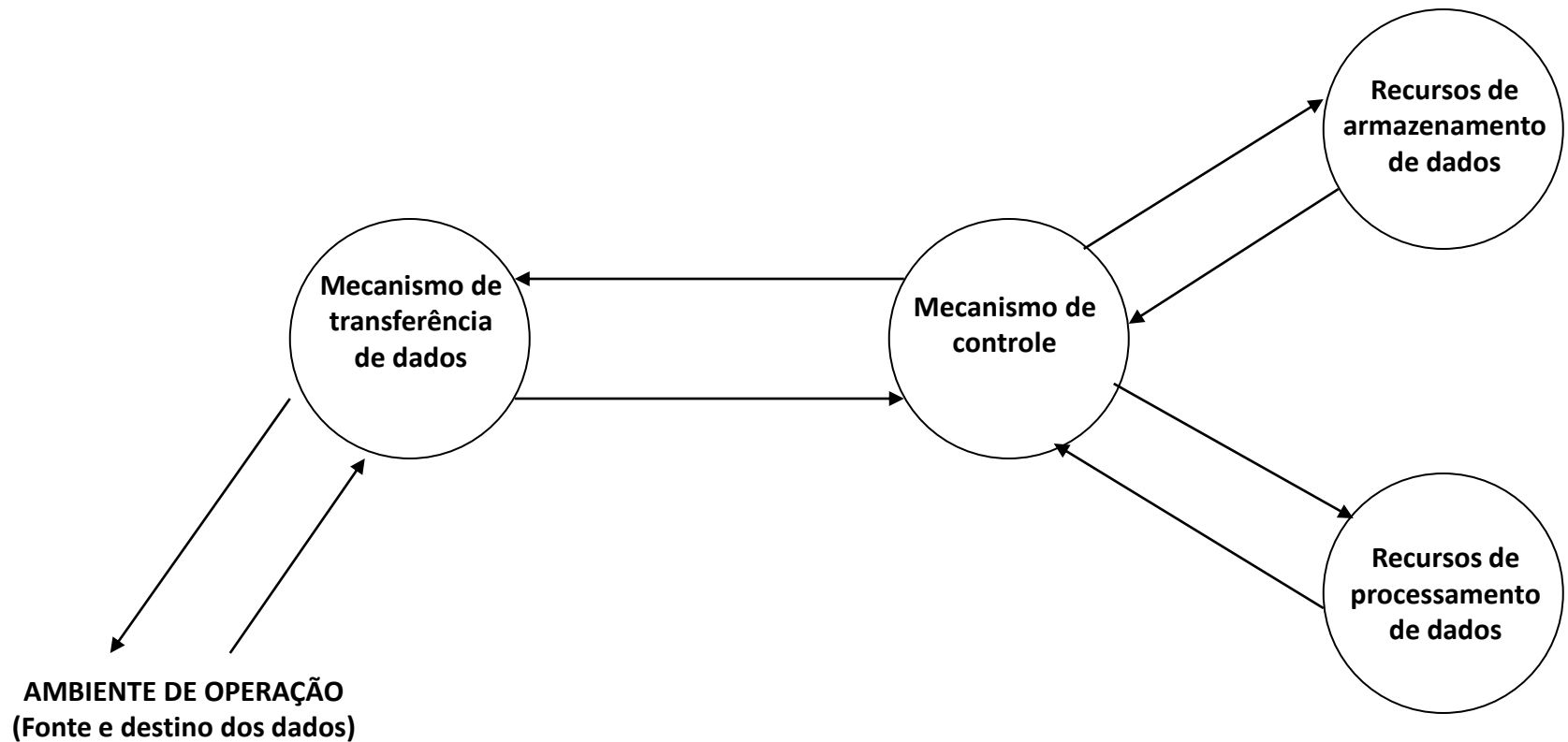
- Estrutura é o modo que os componentes estão inter-relacionados
- Função é a operação de cada componente individualmente como parte da estrutura

Função

- Em geral existem quatro:
 - Processamento de dados
 - Armazenamento de dados
 - Transferência de dados
 - Controle

Visão Funcional

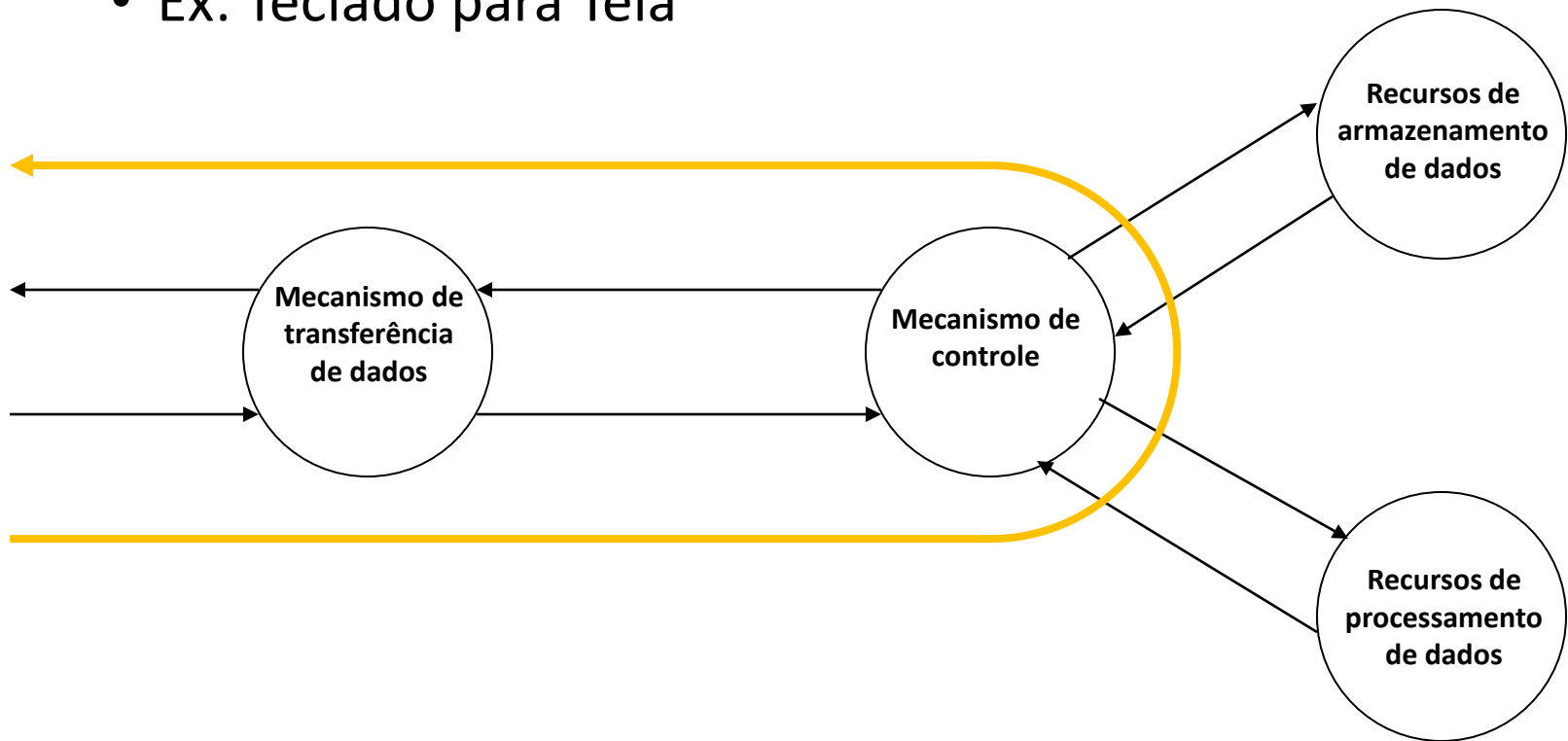
- Visão funcional do computador



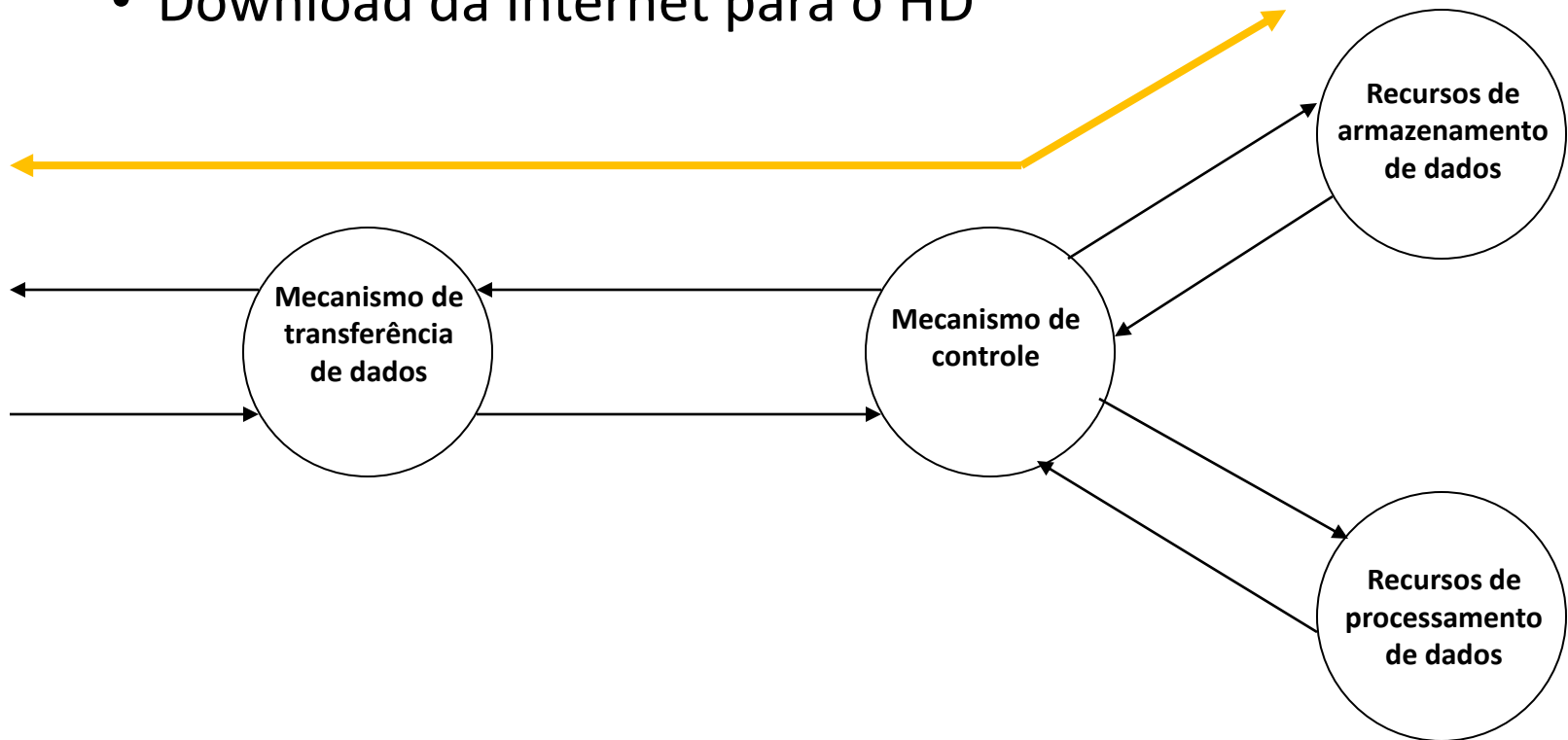
Operações

1

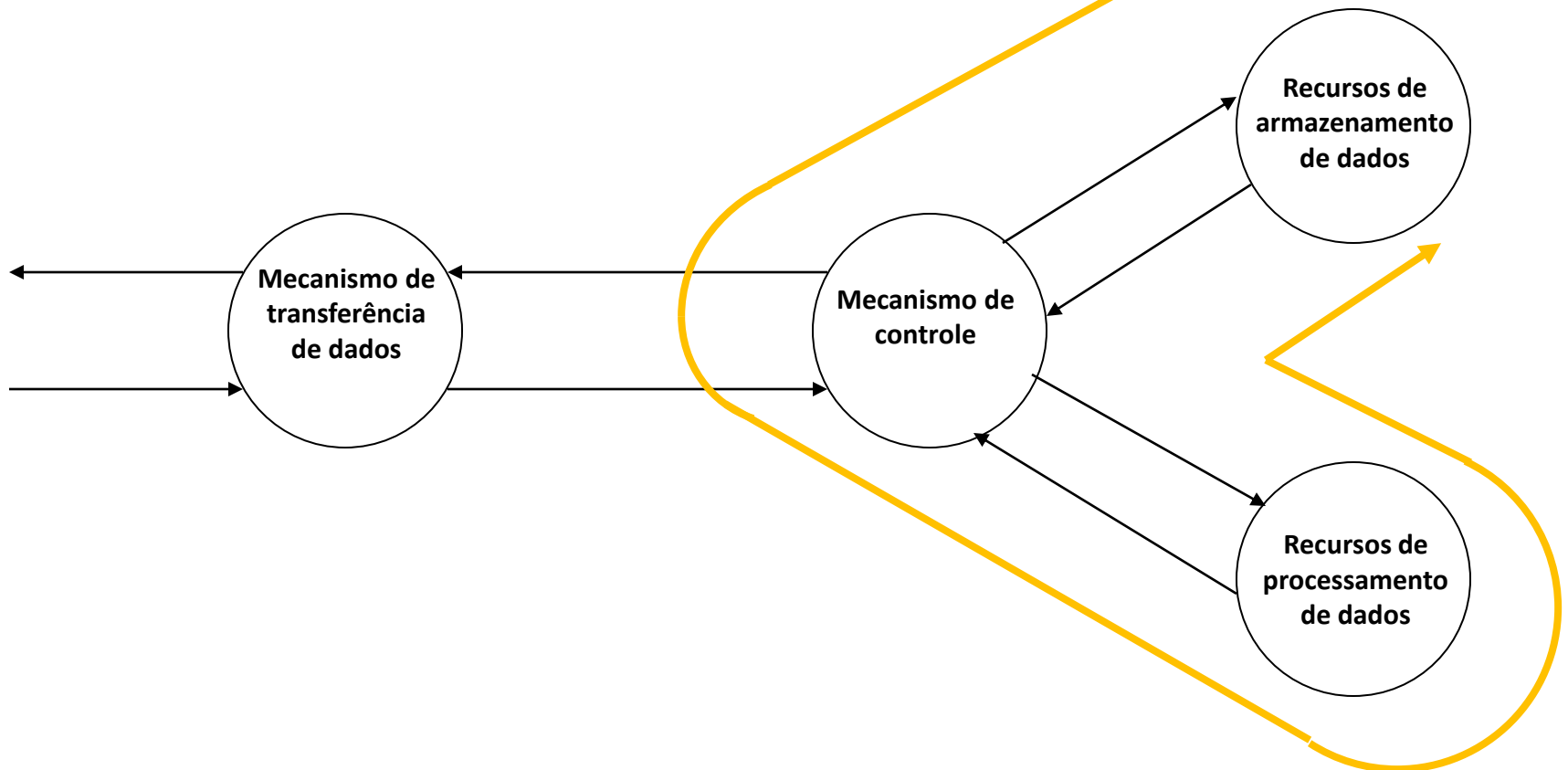
- Transferência de dados
 - Ex: Teclado para Tela



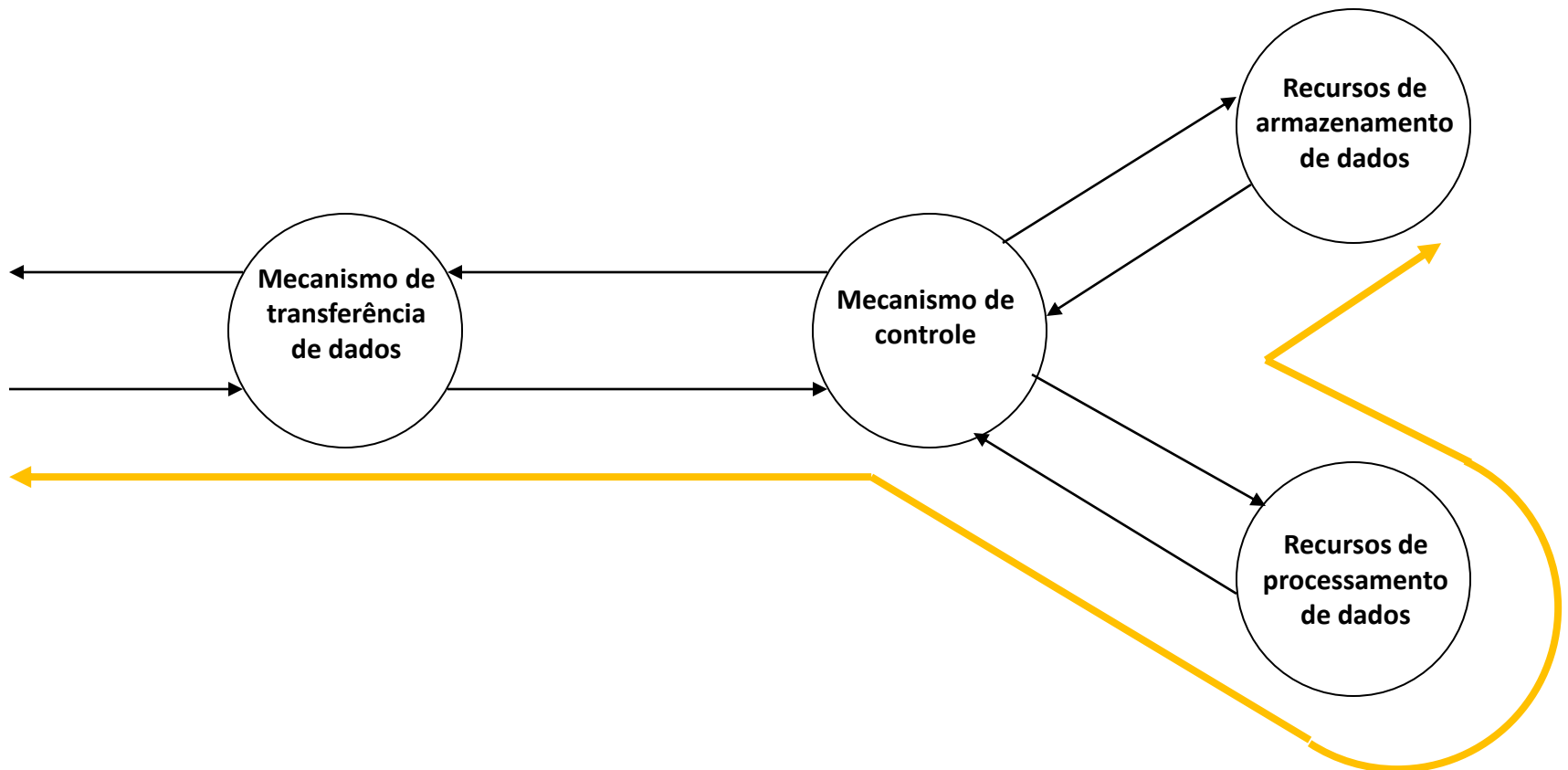
- Armazenamento
 - Download da Internet para o HD



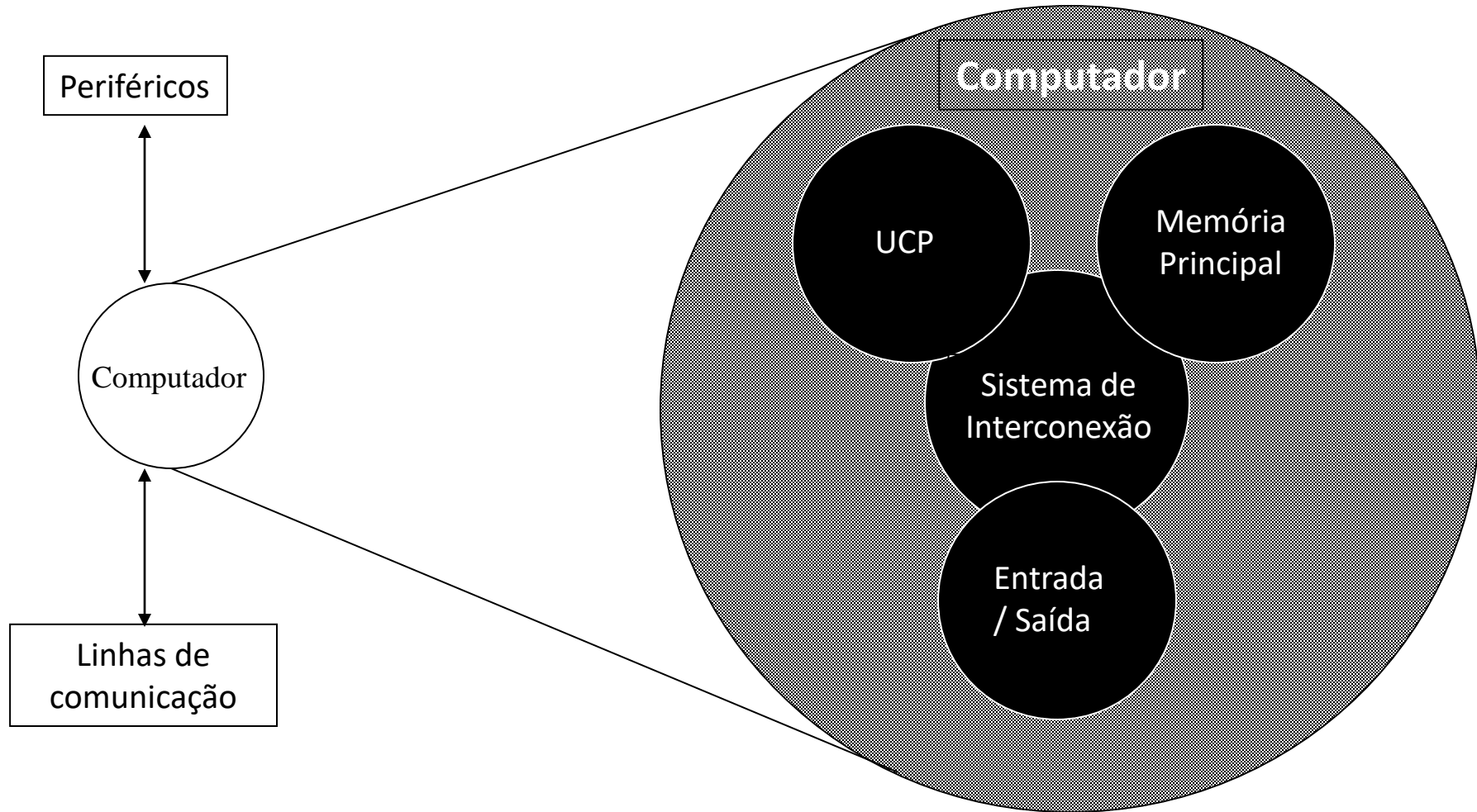
- Processamento de dados armazenados na memória



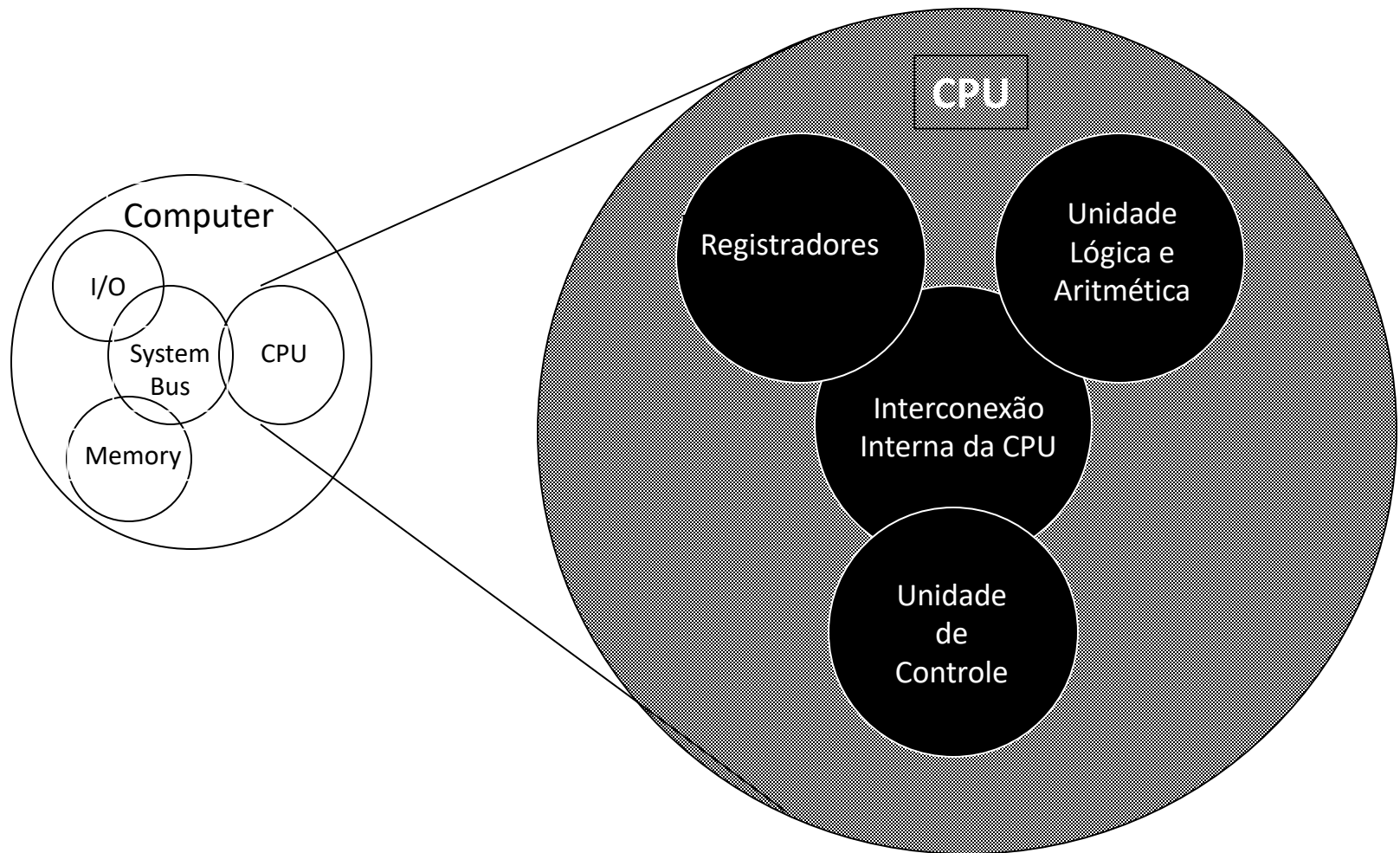
- Processamento de dados para o Ambiente Externo



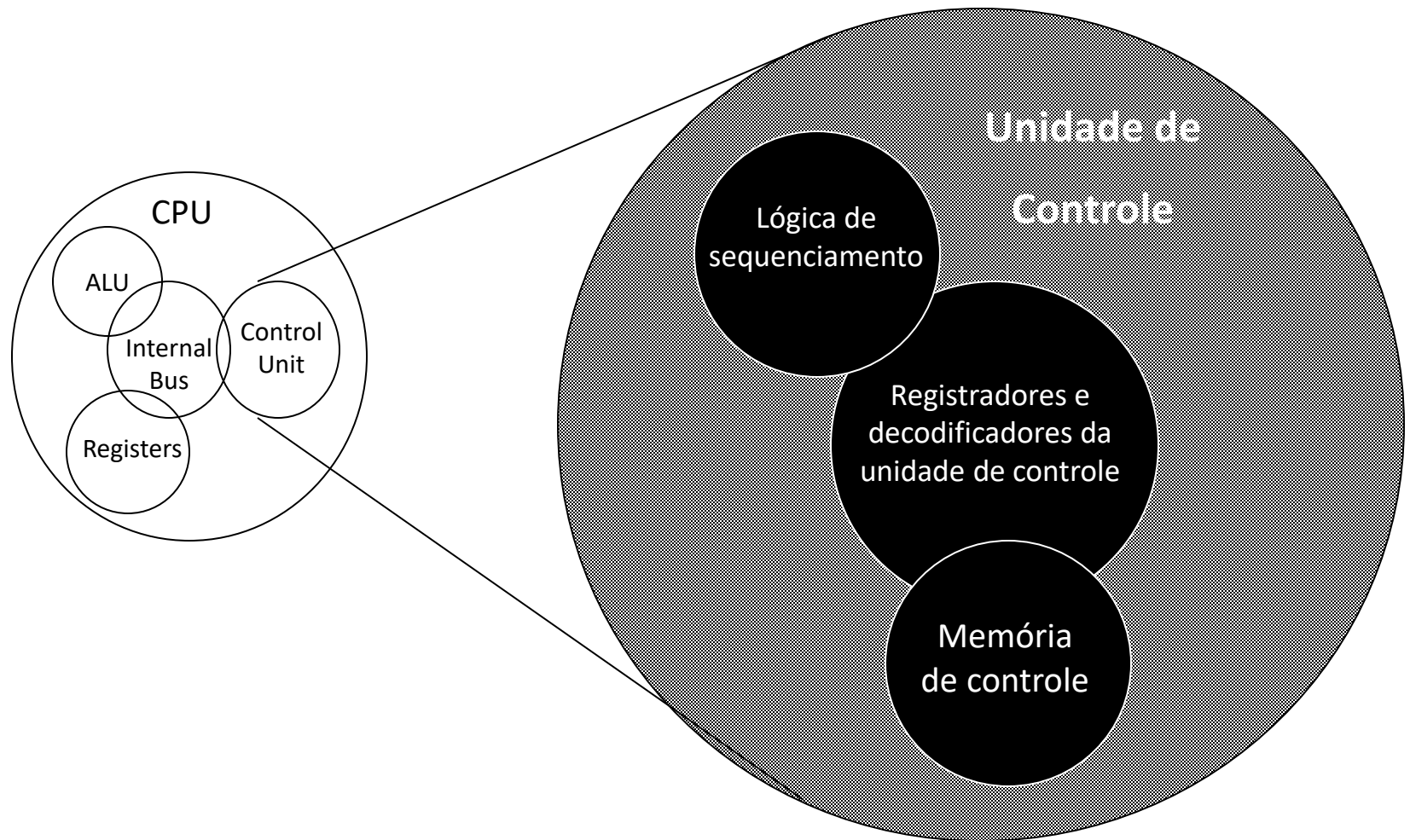
Estrutura –Top Level (nível superior)



Estrutura - CPU

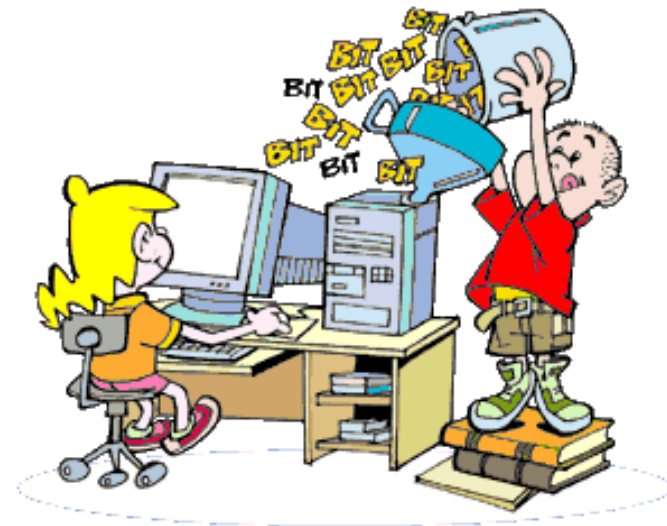


Estrutura – Unidade de Controle



Introdução: Abaixo do seu Programa

- Para se fazer realmente entender por uma máquina eletrônica, é necessário falar com ela através de sinais elétricos
 - Sim / Não
- Alfabeto do computador possui apenas duas letras
 - Símbolos escolhidos para representar foram os algarismos **0** e **1**
 - Visualizada como um conjunto de números na base 2, ou **números binários**
 - **Dígito binário** ou **bit** é cada uma das duas letras



Introdução: Abaixo do seu Programa

- Os computadores são escravos de nossos comandos, razão pela qual chamamos cada comando individual de **instrução**
- As instruções nada mais são que um conjunto de bits inteligíveis pelo computador e que podem ser associadas a números
 - Por exemplo:
os bits 1000110010100000
informam a determinado computador que ele deve somar dois números

Introdução: Abaixo do seu Programa

- Assim faziam os primeiros programadores ...
 - Logo inventaram uma notação mais próxima da maneira como as pessoas pensam.
 - Inicialmente traduzida a mão para linguagem binária
 - Uso da máquina para programar a própria máquina foi a forma de otimizar isso
- “O primeiro destes programas foi chamado de **montador** que traduz programas escritos em linguagem simbólica para linguagem de máquina”

Introdução: Abaixo do seu Programa

Por exemplo: o programador poderia escrever

add A, B

deixando para o montador a tarefa de traduzir esta instrução para

1000110010100000

O nome escolhido para essa linguagem simbólica, ainda em uso nos dias de hoje, foi **linguagem de montagem**

Introdução: Abaixo do seu Programa

- Linguagem de montagem:
 - Sofreu muitas mudanças e melhorias
 - Ainda está longe de representar a notação desejada
 - Obriga o programador a escrever uma linha para cada instrução a ser executada pela máquina
 - Deve raciocinar como a máquina

“Se podemos escrever um programa para traduzir a **linguagem de montagem** para **instruções binárias**, o que nos impede de escrever um programa que traduz uma **notação de mais alto nível** para a notação **binária**?”

Resposta: NADA!

Introdução: Abaixo do seu Programa

- Os programadores de hoje devem sua alta produtividade – e até mesmo sua sanidade – ao sucesso do projeto desse tradutor de alto nível, denominado **Compilador**
- Compiladores são programa que aceitam uma notação mais natural, muito próxima da nossa linguagem
 - A linguagem que eles compilam é denominada **linguagem de programação de alto nível**

Introdução: Abaixo do seu Programa

- Seu uso permite que um programador escreva a seguinte expressão:

A + B

o compilador deve compilar esta expressão para a declaração seguinte, em linguagem de montagem

add A, B

... O montador deve traduzir este comando para a instrução binária ...

1000110010100000

High-level
language
program
(in C)

```
swap(int v[], int k)
{
    int temp;
    temp = v[k];
    v[k] = v[k+1];
    v[k+1] = temp;
}
```

Compiler

Assembly
language
program
(for MIPS)

```
swap:
    muli $2, $5, 4
    add  $2, $4, $2
    lw   $15, 0($2)
    lw   $16, 4($2)
    sw   $16, 0($2)
    sw   $15, 4($2)
    jr   $31
```

Assembler

Binary machine
language
program
(for MIPS)

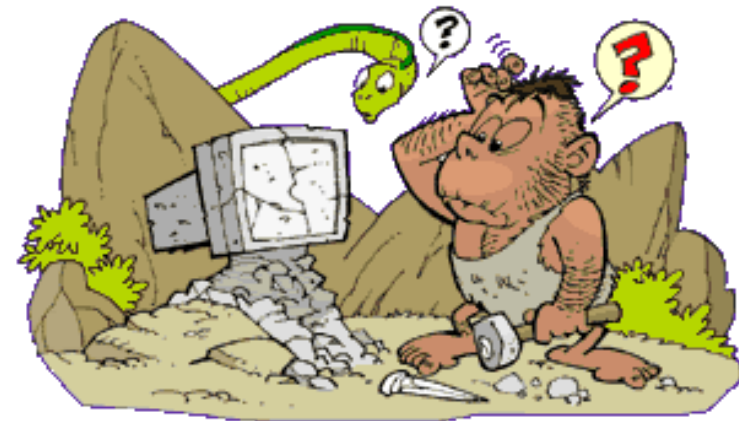
```
0000000010100001000000000011000
00000000000110000001100000100001
1000110001100010000000000000000
10001100111100100000000000000100
1010110011110010000000000000000
10101100011000100000000000000100
0000001111100000000000000001000
```


Introdução: Abaixo do seu Programa

- Linguagem de Programação de Alto Nível

Vantagens:

- Permitem raciocinar de uma forma mais natural, usando palavras em inglês e notações algébricas
- Projetadas de acordo com o tipo de programa a ser escrito
- Aumento de produtividade
- Independentes do computador no qual foram desenvolvidos



Introdução: Abaixo do seu Programa

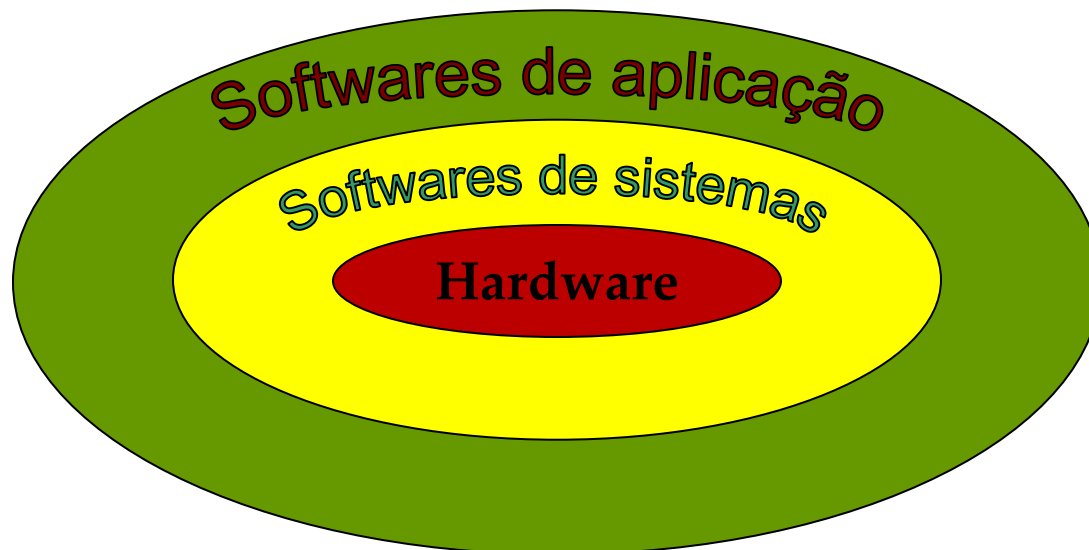
- A medida que a disciplina de programação atingia sua maturidade, observou-se a vantagem da reutilização de código
 - Começaram a compartilhar as rotinas mais utilizadas em bibliotecas chamadas de **bibliotecas de sub-rotinas**
- A experiência mostrou que um conjunto de programas poderia rodar mais eficientemente se houvesse um programa separado encarregado de supervisionar a execução dos demais

Introdução: Abaixo do seu Programa

- Tão logo um programa terminasse, o programa supervisor já poderia executar procedimentos necessários e iniciar o processamento do próximo programa na fila, evitando perda de tempo com intervenção humana
 - Esses programas ditos supervisores foram a base do que hoje chamamos sistemas operacionais
- “**Sistemas Operacionais** são programas que gerenciam os recursos de um computador”

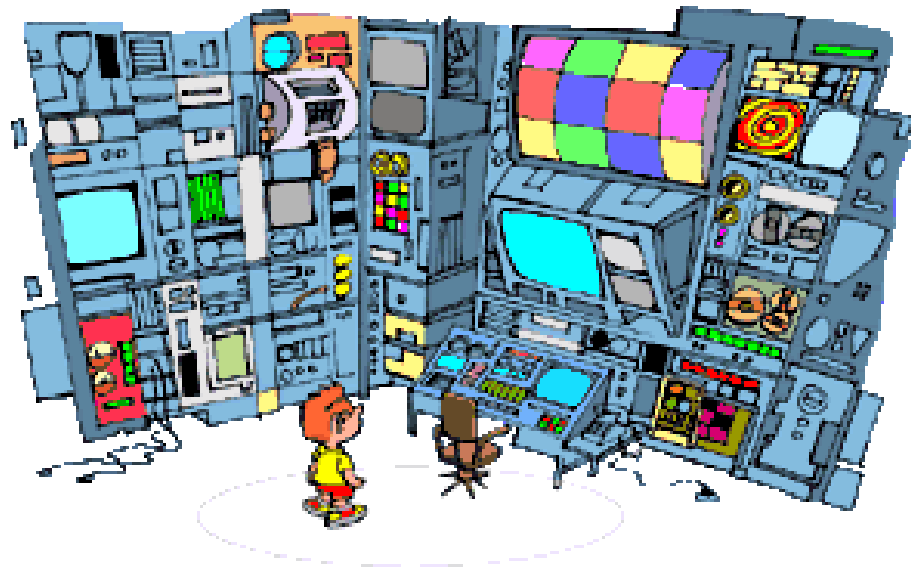
Introdução: Abaixo do seu Programa

- Os softwares tendem a ser classificados por seu uso
 - **Softwares de Sistema**: Sistema Operacional, Compiladores e montadores
 - **Software Aplicativo**: planilhas eletrônicas, editores de texto



Introdução: Debaixo das tampas

- Depois de ter dado uma olhada no que vai por baixo do seu programa, vamos abrir o gabinete e aprender um pouco sobre o hardware



Introdução: Debaixo das tampas

- Típico computador é formado por:

Dispositivos de entrada e saída

- Teclado
- Mouse
- Video



Introdução: Abrindo o gabinete

- Removendo a tampa do gabinete temos o conjunto de placas:
 - Placa-mãe



Introdução: Abrindo o gabinete

- **Memória** é o local onde os programas ficam armazenados enquanto estão sendo processados
- Na memória ficam também os dados necessários à execução dos programas
- O **processador** é justamente a parte ativa da placa-mãe, responsável direto pela execução das instruções de um programa.
 - Soma, subtrai, compara valores ...
 - Chamado de **CPU** ou **central processing unit**

Introdução: Abrindo o gabinete

- Descendo ainda mais baixo na hierarquia do hardware temos detalhes do processador
- Ele é constituído por dois componentes básicos:
 - Caminho de dados (músculo)
 - Unidade de controle (cérebro)
- É no caminho de dados que as instruções são processadas
- A unidade de controle informa ao caminho de dados, à memória e aos dispositivos de entrada/saída o que cada um deve fazer

Componentes clásicos

