



Organização e Arquitetura de Computadores
Organização Estruturada de Computadores


Prof. Marcelo Rabello
marcelo.rabello@unifg.edu.br



2

Objetivos de aprendizagem

Introdução a disciplina




1. Descrever como o computador é constituído, em termos de estrutura e função;

3

Computação e Sociedade

- Internet;
- Forma de negócio e interações sociais;
 - Home Office;
- Realidade Virtual e Aumentada;
- Computação Pervasiva e Ubíqua (IoT).
- Supercomputação;
 - Previsão meteorológica;
 - Simulação.



4

Classes de Computadores

Desktops e notebooks



- Propósito geral;
- Custo/desempenho importantes.

Servidores



- Comunicação;
- Escalabilidade;
- Disponibilidade.

Computadores embarcados



- Restrições de desempenho/custo/capacidade;
- Arquiteturas ARM, PowerPC, PIC, AVR, 8051, Coldfire, TMS320.

5

Computador

O que é?

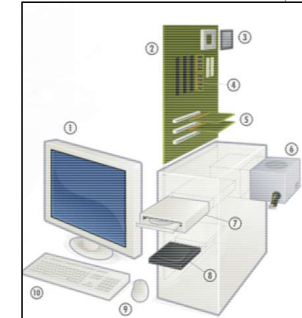
- O computador é uma máquina que processa dados, orientada por um conjunto de instruções e destinada a produzir resultados completos, com um mínimo de intervenção humana.
- É uma máquina constituída por uma série de componentes e circuitos eletrônicos, capaz de receber, armazenar processar e transmitir informações.



6

Introdução - Arquitetura de Hardware

- 01 - Monitor
- 02 - Placa-Mãe
- 03 - Processador
- 04 - Memória RAM
- 05 - Placas de Rede, Som, Vídeo, Fax...
- 06 - Fonte de Energia
- 07 - Leitor de CDs e/ou DVDs
- 08 - Disco Rígido (HD)
- 09 - Mouse
- 10 - Teclado



7

Introdução - Conceitos

Computador Digital

- É uma máquina que pode resolver problemas executando uma série de instruções.
- Máquina Programável.

Programa

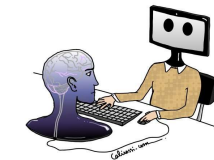
- Uma sequência de instruções que descrevem a maneira de se realizar uma determinada tarefa.

Linguagem de Máquina

- Conjunto de instruções básicas que os circuitos eletrônicos de um determinado computador pode reconhecer e executar diretamente;
- Linguagem primitiva (binária), mas é a linguagem que a máquina realmente entende:
 - Somar 2 números;
 - Comparar o valor de um número com zero;
 - Copiar um conjunto de dados de uma parte da memória para outra parte.
- Complicada para uso humano

8

Comunicação Homem - Máquina



9

Linguagens de Alto Nível

- C, C++, Delphi, Pascal, Fortran, Java,...
- Facilitam a comunicação dos seres humanos com a máquina, sendo linguagens mais próximas dos humanos.
- Devem ser convertidas em linguagem de máquina para serem executadas.
- Métodos de conversão: **Interpretadores, Compiladores e Tradutores.**

10

Interpretadores, Compiladores e Tradutores

Existem três métodos básicos de abordagem na tradução de linguagem de alto nível para linguagem de máquina:

Primeiro Método: Interpretador

O programa conversor recebe a primeira instrução do programa fonte, confere se está escrita corretamente, converte-a em linguagem de máquina e executa esta instrução. Depois repete o processo para a segunda instrução e a primeira é perdida, isto é, apenas uma instrução fica na memória em cada instante.

Se este programa fonte for executado uma segunda vez, novamente haverá uma nova tradução, comando por comando.

11

Interpretadores, Compiladores e Tradutores

Segundo Método: Compilador

- O programa conversor recebe a primeira instrução do programa fonte, confere-a para ver se está escrita corretamente, converte-a para linguagem de máquina em caso afirmativo e passa para a próxima instrução até a última instrução do programa fonte.
- Se este programa for executado uma segunda vez, não haverá necessidade de uma nova tradução, uma vez que todos os comandos em linguagem binária foram memorizados em um novo programa completo.

12

Interpretadores, Compiladores e Tradutores

Segundo Método: Compilador

VANTAGEM:

Neste processo a execução fica mais rápida em relação à anterior, pois se economiza o tempo de re-tradução de cada instrução a cada nova execução.

DESVANTAGEM:

A cada modificação introduzida no programa fonte é necessária uma nova tradução completa para obter um novo programa objeto, o que torna o processo mais dificultoso na fase de desenvolvimento, quando muitas modificações são feitas e geralmente o **compilador** exige mais memória que o **interpretador**.

13

Interpretadores, Compiladores e Tradutores

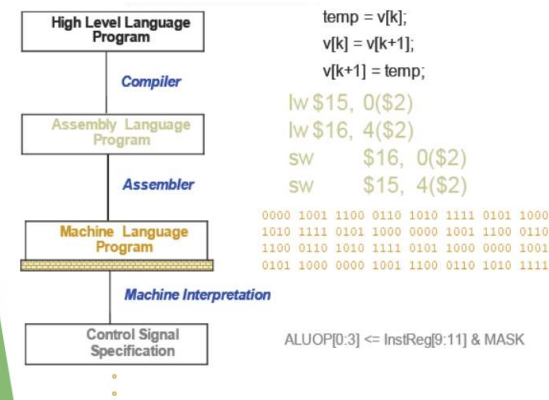
Terceiro Método: Tradutor

Ao invés de armazenar as instruções do programa fonte tal como lhe são fornecidas, ele transforma-as em códigos intermediários, que não exigem tanto espaço de memória como as instruções originais. Estas instruções intermediárias são em seguida transformadas em linguagem de máquina e executadas, funcionando daí diante como no **Interpretador**.

Neste método o programa conversor é conhecido como **Tradutor**.

14

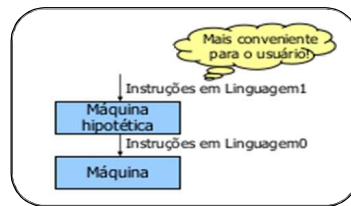
Interpretadores, Compiladores e Tradutores



15

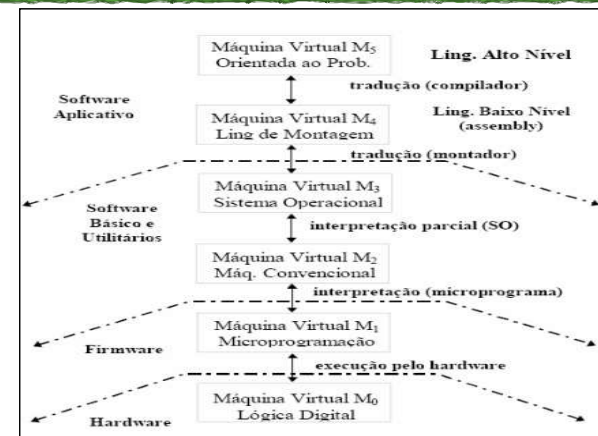
Organização Estruturada de Computadores

- Maneira estruturada de se organizar os computadores em diferentes Níveis de Abstração;
- Cada “**abstração**” é construída com base naquela que encontra-se imediatamente abaixo;
- Projeto de computadores de maneira sistemática e organizada.



16

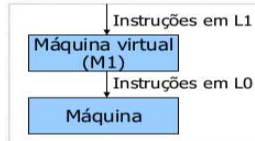
Hardware, Software e Firmware



17

Máquina Virtual

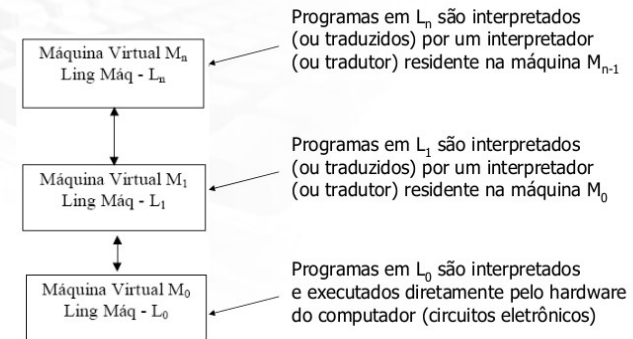
Representa uma abstração capaz de reconhecer e executar diretamente as instruções de uma linguagem específica.



- **L0**: Linguagem de máquina; **M0**: Máquina capaz de executar **L0**.
- **L1**: Linguagem em um nível de abstração acima de **L0**;
- **M1**: Máquina virtual associada a linguagem **L1** (abstração):
 - Os programas escritos em **L1** são traduzidos ou interpretados por **M1**;
 - **M1** representa um programa escrito em **L0**, sendo executado (interpretado) por **M0**.

18

Máquina de Vários Níveis



19

Máquina de Vários Níveis

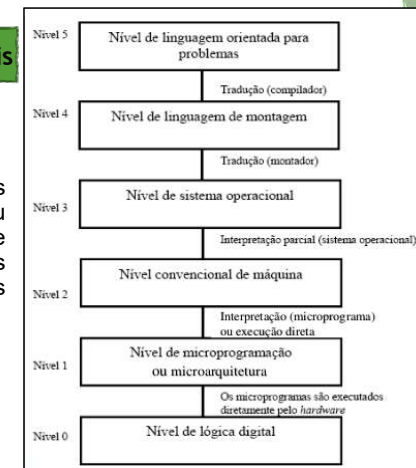
Vantagens desta abordagem:

- Um computador pode ser considerado como composto por "n" máquinas virtuais, cada uma delas com sua própria configuração e linguagem de máquina;
- Gerar programas (aplicativos) para a máquina virtual de nível "k", não exige o conhecimento das particularidades dos níveis inferiores;
- A estrutura de níveis permite ver o computador como um conjunto hierárquico de facilidades que possibilitam flexibilidade e independência ao usuário.

20

Máquina de Vários Níveis

A maioria dos computadores modernos possui dois ou mais níveis, sendo que máquinas com seis ou mais níveis são cada vez mais comuns.



21

Nível 0 ou Nível da Lógica Digital

É composto pelo hardware da máquina e o nível 0 executa diretamente as instruções (ou micro instruções) submetidas pelo nível 1.

Portas Lógicas

- São os objetos de interesse dos projetistas de computadores nesse nível;
- Dispositivos digitais construídos a partir de componentes analógicos (Ex: transistores: portas lógicas AND, OR e NOT).

Combinação de portas lógicas:

- Funções aritméticas;
- Memórias (registradores);
- Processadores.

22

Nível 1 ou Nível da Microarquitetura

Nesse nível, inicia-se o conceito de programa como uma sequência de instruções a serem executadas diretamente pelos circuitos eletrônicos.

Enxerga-se:

- Um conjunto de 8 a 32 registradores (memória local);
- Um circuito chamado **ULA (Unidade Lógica e Aritmética)**;
- Os registradores e a ULA são conectados para formar o **Caminho de Dados (Data Path)**, estrutura sobre a qual os dados fluem.
- A operação básica do caminho de dados consiste na seleção de um ou de dois registradores para que a ULA opere sobre eles.

23

Nível 1 ou Nível da Microarquitetura

O controle do *data path* por software

- Realizado geralmente em máquinas CISC (*Complex Instruction Set Computer*).
- Nas máquinas não microprogramadas, o caminho de dados é quase sempre **controlado por hardware**
 - Realizado geralmente em máquinas RISC (*Reduced Instruction Set Computer*).
- Poucas são as máquinas que têm mais de 20 instruções no nível da microarquitetura;

24

Nível 1 ou Nível da Microarquitetura

Controle por software

Microprograma: Interpretador (residente em memória ROM)

Busca, decodifica e executa as instruções, uma a uma, usando o caminho de dados para a realização de uma tarefa.

Exemplo de execução de uma instrução de SOMA (ADD):

A instrução deve ser buscada na memória, seus operandos devem ser localizados e trazidos para os registradores, a soma deve ser calculada na ULA, e o resultado deve ser encaminhado para o lugar apropriado.

Controle por hardware

- Os mesmos passos acima são executados, mas sem que haja um programa armazenado para controlar a interpretação das instruções.

25

Nível 2 ou Nível ISA (Instruction Set Architecture)

Arquitetura do Conjunto de Instruções

Nível Convencional de Máquina

- Define o conjunto das Instruções executáveis por uma máquina (processador);
- Cada máquina ou processador tem sua linguagem própria de nível 2, chamada "**Linguagem de Máquina**" (documentada em manuais específicos de cada fabricante);
- Pode ser vista como a verdadeira interface entre o **software** e o **hardware**.

Discute-se:

- Tipos de Dados;
- Modelos de Memória e de Endereçamento;
- Formatos e Tipos de Instruções.

26

Abstrações

- Abstrações ajudam a tratar complexidade;
- Revelam detalhes quando necessário;
- Uma das abstrações mais importantes é a interface entre o hardware e o software de nível mais baixo;
- **ISA – Instruction Set Architecture** - Instruções; Registradores; Acesso a memória e E/S.

27

Nível 3 ou Nível do Sistema Operacional (SO)

Esse nível suporta um conjunto de novas instruções, uma organização diferente da memória, sendo capaz de rodar dois ou mais programas de forma simultânea, e outros.

Fornecer serviços básicos para os níveis acima:

- Interface (gráfica ou linha de comando) com o usuário;
- Gerenciamento de memória;
- Escalonamento de processos;
- Acionamento de dispositivos de entrada e saída de dados, etc.

Geralmente desenvolvido de forma híbrida

- Existem instruções do nível 3 idênticas às de nível 2.

28

Nível 3 ou Nível do Sistema Operacional (SO)

Níveis abaixo: dirigidos aos **programadores de sistema**;

- Níveis projetados para rodar interpretadores e tradutores (ex. Compiladores);
- Os programadores de sistema são especialistas em projetar e implementar novas máquinas virtuais/interpretadores/tradutores;
- Predominância de Interpretação;
- Linguagens frequentemente numéricas, bom para as máquinas, mas ruim para as pessoas.

Níveis acima: dirigidos aos **programadores de aplicação**

- Programadores com problemas a serem solucionados;
- Predominância de tradução (mas nem sempre);
- Linguagens contendo palavras e abreviações (significativas para as pessoas).

29

Nível 4 ou Nível de Linguagem de Montagem

Uma forma simbólica de representação das linguagens dos níveis mais baixos.

- Provê um método para as pessoas escreverem programas para os níveis 1, 2, e 3 de uma maneira não tão desconfortável.
 - **Mnemônicos:** é um conjunto de técnicas utilizadas para auxiliar o processo de memorização.
- Os programas escritos em linguagem de montagem são primeiramente traduzidos para a linguagem dos níveis 1, 2 ou 3, e depois interpretados.

30

Nível 5 ou Nível de Linguagem de Alto Nível

Linguagens projetadas para serem utilizadas por programadores de aplicação com problemas a serem resolvidos.

- Ex. de linguagens de alto nível: C, C++, Basic, Java...
- Os programas escritos nessas linguagens são geralmente traduzidos para o nível 3 ou nível 4 por tradutores conhecidos como compiladores, embora às vezes sejam interpretados (como no caso de Java).

31

Mais Níveis

Acima do nível 5 encontram-se coleções de programas projetados para criar máquinas especialmente adequadas para certas aplicações (ou domínios), contendo grandes quantidades de informação acerca da aplicação.

- **Máquinas virtuais voltadas às aplicações**
 - Administração, educação, projeto de computadores, realidade virtual, etc.
- Dependendo do projeto da arquitetura, os níveis podem variar.

32

Arquitetura de Computadores

É o estudo de como projetar as partes de um sistema de computador visíveis aos programadores.

Pontos fundamentais:

- Computadores são projetados como uma série de níveis;
- Cada nível é construído em cima de seus precursores.

Cada nível representa uma abstração distinta, com diferentes objetos e operações presentes

- Arquitetura do Nível: conjunto de tipos de dados, instruções e características
- Abstrai-se o que é irrelevante, reduz a complexidade e foca no que interessa.

33

Hardware, Software e Firmware

Hardware

- É composto por objetos tangíveis (parte física) - circuitos integrados, placas de circuito impresso, cabos, fontes de alimentação, memórias, impressoras, etc.

Software

- É composto de instruções, algoritmos e por suas representações computacionais - os programas;
- Hardware e Software são logicamente equivalentes;
- Qualquer operação efetuada pelo software pode também ser implementada pelo hardware... E qualquer instrução executada pelo hardware, pode também ser simulada pelo software.

Firmware

- Software embarcado (embutido no dispositivo durante a fabricação);
- Controla o hardware diretamente. Ex.: BIOS (*Basic Input/Output System*);
- Em muitos computadores o microprograma está em firmware.

34

Desempenho de um Programa

Algoritmo

- Determina o número de instruções do código fonte e o número de operações de entrada e saída.

Linguagem, compilador e arquitetura

- Determina o número de instruções de máquina para cada instruções em nível da fonte.

Processador e sistema de memória

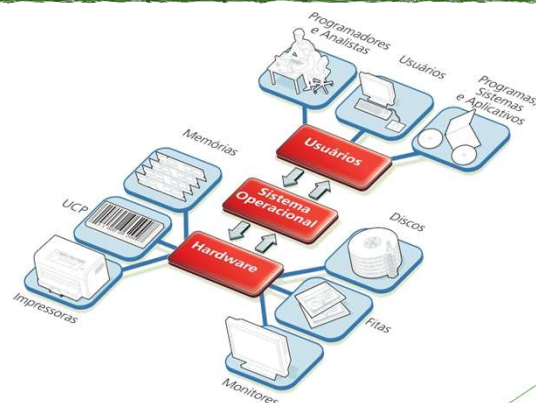
- Determinam a velocidade com que as instruções podem ser executadas(Ex.:Arm x Xeon).

Sistema de E/S(inclusive SO)

- Determina a velocidade em que as operações de E/S podem ser verificadas.

35

Desempenho de um Programa



36

Desempenho de um Programa

Software de aplicações

- Camada mais externa;
- Escrito em linguagem de alto nível(HLL);
- Pode ter múltiplas camadas.

Internet Banking:
- Aplicação; - Banco de dados;
- Servidor Web.

Sistema de software

- Compilador:** traduz HLL(Alto Nível) para *assembly*(Depende da arquitetura – AMD/ITEL e ARM);
- Montador:** traduz instruções simbólicas para binário;
- Sistema Operacional:** programa de supervisão que gerencia os recursos de um computador em favor dos programas executados nessa máquina.

Hardware

- Núcleo das camadas de abstração;
- Processador, memória, controlador de E/S.

37

Níveis de Programação

- **Representação para Hardware**
 - Bits;
 - Instruções e dados codificados.
- **Linguagens Assembly**
 - Representação textual das instruções.
- **Linguagem de alto nível**
 - Próximo do domínio do problema;
 - Portabilidade;
 - Reusabilidade.

38

Atividade Extraclasse

- Ler o capítulo 1 do livro Organização estruturada de computadores. 6º ed. São Paulo. Pearson Pratic Hall, 2013. (Disponível na Biblioteca Virtual - Person).

39

Dúvidas? Sugestões?

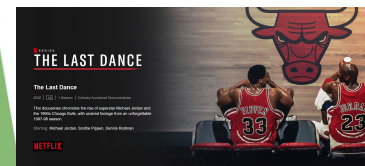


40



"Why would I think about missing a shot I haven't taken?"

Michael Jordan



*"Seems that needlessly it's getting harder
To find an approach and a way to live
Are we getting something out of this
All encompassing trip?"*

*You can spend your time alone.
Redigesting past regrets
Or you can come to terms and realize
You're the only one who can't forgive yourself
Makes much more sense
To live in the present tense"*

https://www.youtube.com/watch?v=653rA5RtO_s

Present Tense by Pearl Jam

41