Slide 1

**Computador:** conjunto de componentes e circuitos eletrônicos, capaz de armazenar, receber, processar e transmitir dados. Conjunto de hardware.

**Sistema Computacional:** um conjunto de componentes integrados com o propósito de processar dados, realizar algum tipo de operação aritmética/lógica envolvendo dados, de modo a produzir diferentes níveis de informações. Consta o peopleware, software e hardware.

**Hardware:** componente físico de um sistema de computação; os equipamentos que o usuário utiliza para realizar uma tarefa no computador (processar, armazenar, ações de entrada e saída de dados).

**Software:** componente lógico de um sistema de computação; séries de instruções que fazem o computador funcionar (programas de computador).

**Peopleware:** componente humano de um sistema de computação; indivíduos que utilizam o computador para realizar uma tarefa.

**Periféricos:** componentes internos ou externos à CPU.

**Dispositivos on-board:** dispositivos que são embutidos à placa-mãe, utilizando o processador e memória principal para funcionar.

**Dispositivos off-board:** dispositivos que são acoplados à placa-mãe para adicionar uma função, possuem processador e memória própria

**Dispositivos de entrada:** mouse, joystick/controle, microfone, scanner, webcamera...

**Dispositivos de saída:** monitor, impressora, projetor, caiza de som...

**Dispositivos de armazenamento:** usb, CD, floppy disk, HD externo...

**Arquitetura de von Neumann**

Memória é um recurso compartilhado entre dados e instruções. É dividio em cinco componentes:

* Central Aritmética: realiza as operações lógicas e aritmeticas
* Central de Controle: realiza a busca de instruções na memória e gerencia o sequenciamento de instruções. IR contém a instrução atual e PC a próxima instrução a ser executada.
* Memória: dispositivo que armazena dados e instruções.
* Entrada: dispositivo por onde os dados alimentam o sistema.
* Saída: dispositivo por onde os dados retornam.
* Desvantagem: Gargalo de von Neumann – existe porque o processador consegue funcionar mais rápido que a memória, ficando ocioso enquanto espera pela memória.

**Arquitetura de Harvard**

Separar dados e instruções em memórias diferentes. As duas memórias não precisam ser da mesma tecnologia.

Slide 2

**CI/CPU/UCP:** circuito integrado que processa dados através de instruções. Funciona como o cérebro do computador.

**Processadores RISC:** Computador com conjunto reduzido de instruções mais simples. Poucos modos de endereçamento.

**Processadores CISC:** Computador com conjunto complexo de instruções, mais especializadas e com tamanhos variáveis. Muitos modos de endereçamento.

**Clock:** Frequência de ciclo de relógio. Determina a velocidade de um processador (hertz). Contagem de ciclos que o circuito executa em um segundo.

1 Hz = 1 ciclo por segundo

1 MHz = 1.000.000 Hz = 1.000.000 instruções por segundo

1GHz = 1.000 MHz

**Marca:** fabricante do processador

**Tamanho de cache:** quantidade de dados da memória principal que ela pode armazenar

**Níveis de cache:** normalmente são L1, L2 e L3

L1: acesso mais rápido pois está mais próxima do processo. Menor capacidade de armazenamento.

L2: velocidade de acesso média: capacidade de armazenamento médio (últimos dados acessados).

L3: menor velocidade; capacidade de armazenamento maior.

**Core:** é o núcleo do processador. Existem cores e multicores, contém todos os transistores.

**Threads:** é uma forma como o processador vai executar tarefas. Recebe e executa todas as instruções, como uma ordem do que será feito por vez.

**Benchmarks:** é um teste de desempenho que permite comparar o resultado de um produto/software com outro. Leva a exaustão e vê qual é melhor.

**TDP (Potência de Design Térmico):** refere-se ao consumo de energia sob a carga térmica máxima. Determina a potência térmica (calor) máxima gera por um componente eletrônico.

Slide 3

**NEANDER**

* Criado por Raul Weber e Taisy Weber para fins didáticos;
* Projetado a partir da arquitetura de von Neumann;

**Principais características:**

* Largura de dados e de endereços de 8 bits
* Dados representados em complemento de dois
* Modo de endereçamento: direto
* 1 registrador acumulador de 8 bits
* 1 apontador de programa de 8 bits
* 1 registrador de estado com 2 códigos de condição: negativo e zero

0001 
0010 
0011 
0100 
0101 
0110 
1000 
1001 
1010 
Instruqåo 
STA 
LOA end 
ADD end 
OR end 
AND end 
NOT 
'MP end 
JN end 
JZ end 
ALT 
Operagåo 
Nenhuma opera $0 
Armazena acumulador no endereco •end' da merr6ria 
Car-rega o acumulÜr com o conteüdo endereqo "end" da memCria 
Soma o conteüdo do endere90 "end" da memüia ao acumulador 
Efetua operaqåo lÖgica •OIT do conteüdo do endereqo "end" da memoria ao acumulador 
Efetua •E" do conteüdo do endereqo "end" da memma ao actJmulador 
Inverte todos os bits do acurnulador 
Desvio incondicional para o endereqo •end" da memoria 
Desvio condicional, se • , para o endereqo •end- da memöria 
Desvio condicional, se •Z:I . para o enderego •end' da rrernöria 
Para o cich de 

Exercícios:

LDA 120 – coloca o valor que estiver na posição 120 no acumulador

ADD 121 – soma o conteúdo na posição 121 no acumulador

STA 122 – salva o conteúdo que estiver no acumulador na posição 122

HLT – para o ciclo

16 STA

16 – opcode (código da operação)

STa - mnemônico (nome abreviado para acessar rapidamente a função desejada)

SUBTRAÇÃO

32 LDA

100 [5]

96 NOT [-4]

48 ADD

102 [1] (-5)

48 ADD

101 [10]

16 STA

105 [5]

240 HLT

Slide 4

**Sistema Operacional (SO):** é um software básico que "dá vida" ao hardware e fornece as bases para a execução de programas. Está entre o hardware e o software do computador.

**Kernel:** é o coração do sistema operacional, responsável pela gerência dos recursos do hardware usados pelas aplicações. Ele também implementa as principais abstrações utilizadas pelos aplicativos e programas utilitários.

**Drivers:** módulos de código específicos para acessar os dispositivos físicos. É um programa que realiza a comunicação entre o SO e um dispositivo.

* De áudio
* De vídeo
* Da placa-mãe
* De adaptador de rede
* De teclado...

**Interface de Usuário:** é o software que habilita o usuário a interagir com o SO.

* GUI (Graphical User Interface): permite interação visual através de janelas, ícones, menus, botões (Exemplos: Unity, KDE, Gnome...)
* CLI (Comand Line Interface): interface de texto, permite interações via comandos de texto simples. (Exemplos: Bash, Dash, Tmux, Zsh...)

**Gerência de Arquivos:** Cria arquivos e diretórios, definindo sua interface de acesso e regras pare seu uso.

**Gerência de Processos:** responsável por alocar recursos, escalonar, iniciar e finalizar processos.

**Gerência de E/S:** Feito em duas etapas:

* Controle de E/S: preocupa-se com aspectos administrativos de solitações de E/S realizadas. Gerencia filas de controle de acesso aos dispositivos de E/S, NÃO EXECUTA AS ATIVIDADES.
* Controle de Periféricos: preocupa-se com a operalização física dos pedidos. Trabalha diretamente com o hardware

**Gerência de Memória:** gerencia arquivos de dados em um sistema de computador. Tem recursos limitados e é projetado para gerenciar arquivos individuais ou em grupo, como documentos e registros de escritório especiais. Ele pode exibir detalhes do relatório, como proprietário, data de criação, estado de conclusão e recursos semelhantes úteis em um ambiente de escritório.

**Gerência de Rede:** é o processo de configurar, monitorar e manter uma rede confiável, garantindo a conectividade entre os dispositivos e as pessoas ou aplicações de software que as utilizam (?).

Pipeline, muitas vezes traduzido para português como paralelismo, é uma técnica que permite os processadores executarem tarefas

diferentes ao mesmo tempo sendo capazes de respeitar a ordem das instruções que chegam ao processador. Essa técnica aumenta o desempenho

do processador e reduz o tempo de execução global de tarefas.