

Arquitetura de Computadores

Prof. Marcos Grillo

marcos.grillo@anhanguera.com

Apresentação da Disciplina

PLANO DE ENSINO E APRENDIZAGEM					
CURSO: Ciência da Computação					
Disciplina: Arquitetura de Computadores	Período Letivo: 2º sem/2013	Série: 6ª Série	Período: <i>Não definido</i>	Semestre de Ingresso: 1º	Ano de Ingresso: 2011
C.H. Teórica: 40	C.H. Outras: 20		C.H. Total: 60		

Ementa
Arquiteturas RISC e CISC. Pipeline. Paralelismo de Baixa Granularidade. Processadores Superescalares e Superpipeline. Multiprocessadores. Multicomputadores. Arquiteturas Paralelas e não Convencionais. Microprocessadores e Computadores Pessoais. Organização de Memória. Sistemas de Entrada e Saída, Sistemas de vídeo, Som e Outros.

Objetivos
Compreender e assimilar os componentes de dispositivos que compõem o computador. Formas de organização e de comunicação entre os subsistemas computacionais (processador, memória, disco e etc.) Conhecer a estrutura de funcionamento de uma CPU. conhecer as arquiteturas de computadores do tipo CISC e RISC. Conhecer arquiteturas de computadores pessoais, multicomputadores e multiprocessadores.

Apresentação da Disciplina

Cronograma de Aulas	
Semana nº.	Tema
1	Estrutura básica de um computador pessoal
2	Estrutura e Funcionamento da CPU: conjunto de instruções
3	Estrutura e Funcionamento da CPU: ciclo de instruções
4	Arquitetura RISC e CISC
5	Registradores: tipos de registradores
6	Registradores mais utilizados em computadores pessoais
7	Arquitetura Pipeline
8	Atividades de Avaliação.
9	Memórias: principal
10	Memórias: Secundária, cache
11	Dispositivos de entradas e saída
12	Barramento: Tipos, arquitetura, adaptadores
13	Sistema de video: GPU, Memórias, VGA, HDMI, 3D
14	Sistema multimídia
15	Análise de desempenho de computadores (Benchmark)
16	Arquitetura de computadores com paralelismo: Cluster, Cloud.
17	Computadores dedicados e embarcados
18	Prova Escrita Oficial
19	Exercícios de Revisão.
20	Prova Substitutiva

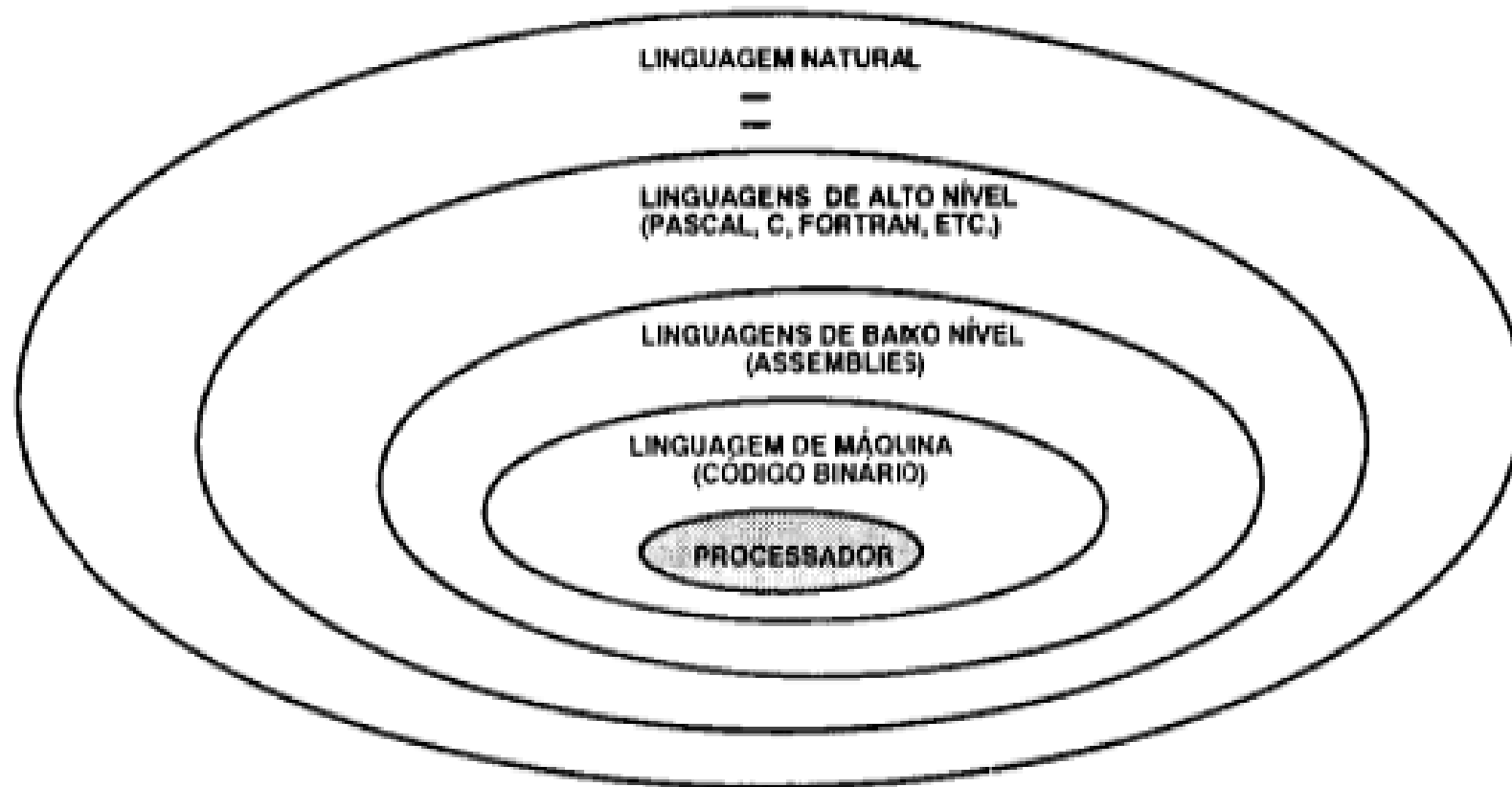
Cronograma de Aulas - 1ª etapa.

- ▶ Estrutura básica de um computador pessoal
- ▶ Estrutura e Funcionamento da CPU: conjunto de instruções
- ▶ Estrutura e Funcionamento da CPU: ciclo de instruções
- ▶ Arquitetura RISC e CISC
- ▶ Registradores: tipos de registradores
- ▶ Registradores mais utilizados em computadores pessoais
- ▶ Arquitetura Pipeline
- ▶ Atividades de Avaliação.

Cronograma de Aulas - 2ª etapa.

- ▶ Memórias: principal;
- ▶ Memórias: Secundária, cache;
- ▶ Dispositivos de entradas e saída;
- ▶ Barramento: Tipos, arquitetura, adaptadores;
- ▶ Sistema de vídeo;
- ▶ Sistema multimídia;
- ▶ Análise de desempenho de computadores (Benchmark);
- ▶ Arquitetura de computadores com paralelismo;
- ▶ Computadores dedicados e embarcados;
- ▶ Prova Escrita Oficial;
- ▶ Exercícios de Revisão;
- ▶ Prova Substitutiva;

Linguagens de Abstração



Operandos e registradores

- ▶ Um **operando** é uma das entradas (argumentos) de um **operador**. Por exemplo, em $5 + 2 = 7$:
 - + é o operador
 - 5 e 2 são os operandos.
- ▶ Ao contrário de linguagens de alto nível, os operandos de instruções aritméticas não podem ser quais quer variáveis;

Conjunto de instruções

- ▶ Linguagens de máquina seguem os mesmos princípios, sendo assim teremos mais facilidade de interpretar conhecendo o conceito porque?
 - ▶ Baseiam-se na mesma arquitetura, von Neumann;
 - ▶ Projetistas com o mesmo objetivo, buscando a facilidade de construção de compiladores;
 - ▶ Operações básicas que todas as máquinas devem oferecer;
 - ▶ Compatibilidade.

Linguagens (classificação)

- ▶ LBN – Linguagem de baixo nível:
 - ▶ **Linguagens de máquina;**
 - ▶ 10101010101;

- ▶ LAN – Linguagem de alto nível:
 - ▶ C – C++;
 - ▶ Pascal;
 - ▶ Programadores utilizam linguagem de alto nível.

Linguagens (classificação)

- ▶ As linguagens LBN são definidas por uma série de Mnemônicos, que são, basicamente, símbolos que representam código binários;
- ▶ Por exemplo, no caso do MIPS a instrução de adição é representada por *add a, b, c*:
 - ▶ Esta instrução determina que o processador some o conteúdo de dois registradores (*b, c*) e coloque o resultado em outro registrador (*a*);
- ▶ A instrução ADD segue um formato bem definido:
 - ▶ código correspondente ao mnemônico da instrução *add*;
 - ▶ Identificação do registrador destino;
 - ▶ Identificação do primeiro operando;
 - ▶ Identificação do segundo operando;

Benefícios de utilizar os registradores.

- ▶ Nível mais próximo do processador, muito mais rápido que memórias RAM (barramentos);
- ▶ Mais fácil utilização por compiladores
 - ▶ Local de armazenamento temporário;
 - ▶ Armazenam valores e reduzem o tráfego da memória, já que se o valor constar neste nível não será necessário ir buscar na memória;

Registradores de programas.

- ▶ Armazena o endereço da instrução sendo executada ou o endereço da próxima instrução;
- ▶ Contador de programa é automaticamente incrementado para cada ciclo de instrução de forma que as instruções são normalmente executadas.
- ▶ Principal função, indicar a próxima instrução a ser executada.

Registradores de instrução.

- ▶ É um registrador invisível de 2 bits que armazena o código da instrução que está sendo executada;
- ▶ Entrada deste registrador recebe diretamente o valor presente nos dois bits mais significativos da via de dados do barramento da memória e a saída deste registrador é ligada diretamente à unidade de controle do processador.

Nomenclatura padrão:

Para:

Soma = add

Subtração = sub

Valores = \$sn

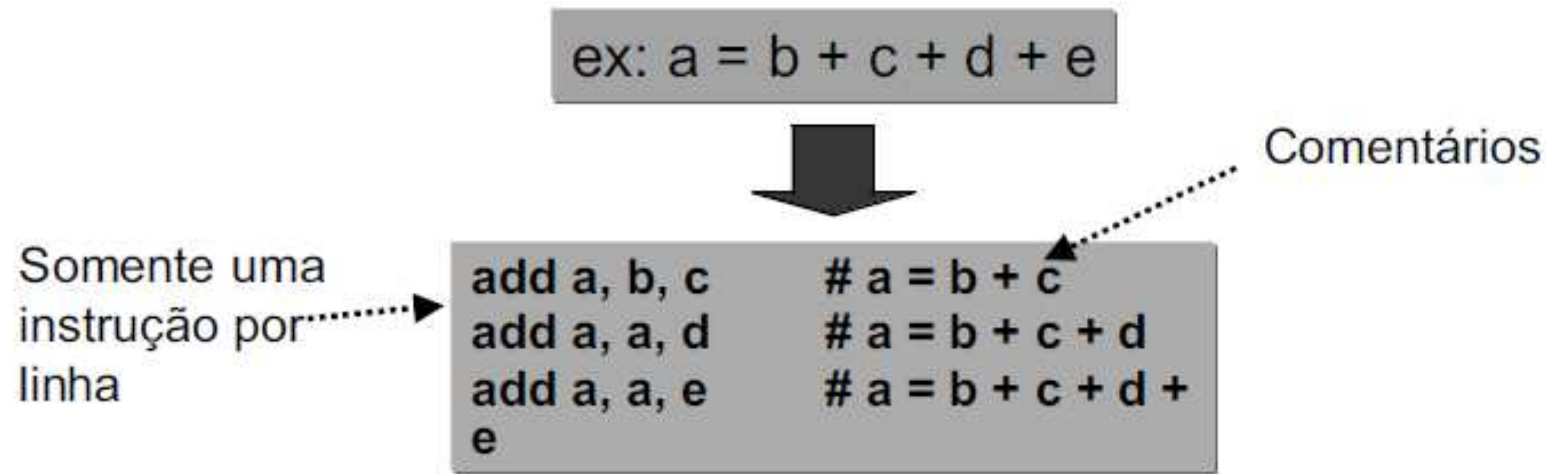
n = número do operando

Temporários = \$tn

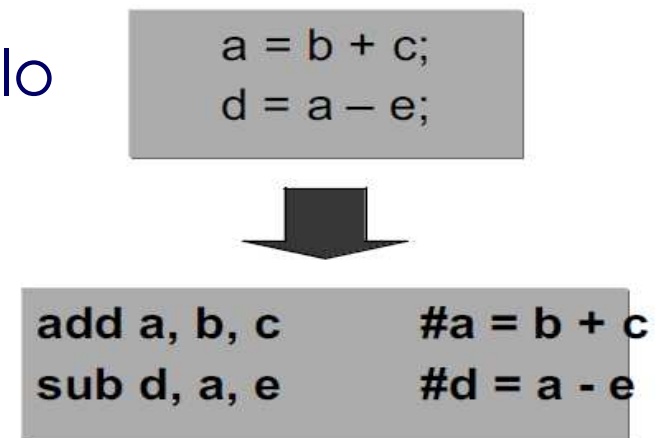
n = número do temporário

Busca externa = lw

Operações de Hardware



Outro exemplo



Qual o código gerado para a seguinte equação?

$$f = (g + h) - (i + j);$$

Resposta!

$f = (g + h) - (i + j);$

Somente uma operação é feita por instrução: necessidade de variáveis temporárias.

add t0, g, h	# temporário t0 = g
+ h	
add t1, i, j	# temporário t1 = i +
j	
sub f, t0, t1	#f = (g + h) - (i + j)