



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Organização de Computadores

AP2 1º semestre de 2007 - GABARITO

Nome –

Assinatura –

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
 2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
 3. Você pode usar lápis para responder as questões.
 4. Ao final da prova devolva as folhas de questões e as de respostas.
 5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

1. (2,0) Considere um processador que possua as seguintes características:

- Um registrador de 8 bits
- Um registrador de 16 bits
- Uma barra de dados de 8 bits
- Uma barra de endereços de 16 bits

Defina instruções que permitam ao processador carregar em um registrador o conteúdo do endereço dado, adicionar a um registrador um valor especificado e carregar no registrador A o conteúdo da posição de memória apontada pelo registrador B. Descreva cada instrução e caracterize o tipo de endereçamento que ela utiliza. Com essas instruções faça um programa que permita carregar no registrador A o elemento de ordem 1F de uma tabela que começa na posição de memória de endereço 013D e que gasta 8 bits de memória para cada elemento.

Um registrador de 8 bits => A
Um registrador de 16 bits => B
Uma barra de dados de 8 bits
Uma barra de endereços de 16 bits

Resposta:

MOV B, X => B <- (X) => O registrador B recebe o conteúdo do endereço apontado por X (sendo X de 16bits) - modo de endereçamento direto

ADD B, Z => B <- B + Z => Adiciona o valor especificado Z ao registrador B (sendo Z até 16 bits) - modo de endereçamento imediato

MOVP A, B => A <- ((B)) => Carrega no registrador A, o conteúdo da posição de memória apontada pelo registrador B. - modo de endereçamento indireto

A solução seria:

Supondo todos os registradores inicialmente carregados com o conteúdo 0000.

ADD B, 013D => B <- 013D => O registrador B recebe o valor 013D

ADD B, 1F => B <- B + 1F => Acrescenta 1F no registrador. Considerando que A tabela inicia em 0, e eu preciso do valor correspondente à posição 1F da tabela.

MOVP A, B => A <- ((B)) => A recebe o conteúdo apontado pelo registrador B

2. (2,0) Responda:

2.1. Por que um programa em linguagem Assembly não é diretamente executável pelo processador? Como esse problema é na prática resolvido?

Resposta:

Porque o Assembly consiste na representação das instruções (em linguagem de máquina) em uma forma que permite uma melhor compreensão (ou lembradas) pelo ser humano. O processador não entende essa representação, e sim a linguagem de máquina cuja simbologia se apresenta apenas em binário (0 e 1).

Para que o processador possa executar um programa em Assembly, antes, este deverá ser traduzido por um Montador para a linguagem de máquina, que desta forma, será compreendido e executado pela CPU.

2.2. Por que um compilador deve ser específico para uma determinada linguagem de programação e para uma determinada UCP?

Resposta:

O compilador tem a função de converter um programa em linguagem de alto nível para linguagem de máquina, de forma a ser entendida e executada pelo processador. O compilador terá que ser específico para uma determinada linguagem de programação devido às variações entre as linguagens: quanto a forma de representação, sintaxe, parâmetros, estrutura do programa, etc.

Mesmo para uma mesma linguagem de programação, um compilador deverá ser diferente em função: Do processador, devido às instruções em linguagem de máquina não serem as mesmas (ex. Risc e Cisc). E do sistema operacional, por causa das chamadas ao sistema para operações como acesso à dispositivos, comunicação entre processo, entre outros.

3. (2,0) Descreva:

3.1.Classificação de arquiteturas segundo Flynn

Resposta:

Consiste em uma das formas mais comuns de classificação de processamento paralelo. São estas as categorias de sistemas de computação:

SISD - Single Instruction Stream, Single Data Stream. Um único processador executa uma única sequência de instruções sobre dados armazenados em uma única memória. Exemplo: Processadores de computadores pessoais.

SIMD – Single Instruction Stream, Multiple Data Stream. Vários elementos de processamento. Cada um tem uma memória de dados. Cada instrução é executada sobre um conjunto de dados diferente. Exemplo: Processadores matriciais.

MISD – Multiple Instruction Stream, Single Data Stream. A sequência de dados é transmitida para um conjunto de processadores, cada um dos quais executa uma sequência de instruções diferente. Não existem processadores comerciais que utilizam este modelo.

MIMD – Multiple Instruction Stream, Multiple Data Stream. Conjunto de processadores executa simultaneamente sequências diferentes de instruções sobre conjuntos de dados diferentes. Exemplo: SMPs, clusters, sistemas NUMA.

3.2.Sistemas SMP

Resposta:

É um sistema computacional independente que contém 2 ou mais processadores similares, estes com capacidade de computação comparável, compartilham a mesma memória principal e facilidades de E/S, podem desempenhar as mesmas funções e sendo controlado por um sistema operacional integrado provendo integração entre programas e processadores.

3.3.Clusters

Resposta:

Consiste em um grupo de computadores completos interconectados, trabalhando juntos, como um recurso de computação unificado, que cria a ilusão de construir uma única máquina. Busca prover alto desempenho e alta disponibilidade, e é particularmente atrativa para aplicações baseadas em servidores.

3.4.Processadores vetoriais e matriciais

Resposta:

Os processadores vetoriais são freqüentemente associados às organizações de ULAs com pipeline de operações, possuem operações especiais para manipular um vetor de dados unidimensional; e processadores matriciais apresentam sob a forma de uma organização de ULAs paralelas cuja tarefa principal é efetuar operações aritméticas sobre vetores ou matrizes de números de ponto flutuante.

4. (2,0) Considere o seguinte conjunto de 32 bits representado em hexadecimal D1E40000. Indique o valor **em decimal (deixe as contas indicadas)** para este conjunto de bits, quando considerarmos que ele está representando:

Resposta:

$$(D1E40000)_{16} = (11010001 \ 11100100 \ 00000000 \ 00000000)_2$$

4.1.(0,3) um inteiro sem sinal

$$2^{31} + 2^{30} + 2^{28} + 2^{24} + 2^{23} + 2^{22} + 2^{21} + 2^{18} = 3.521.380.352$$

4.2.(0,3) um inteiro representado em sinal e magnitude

$$-(2^{30} + 2^{28} + 2^{24} + 2^{23} + 2^{22} + 2^{21} + 2^{18}) = 1.373.896.704$$

4.3.(0,6) um inteiro em complemento a 2

$$-2^{31} + (2^{30} + 2^{28} + 2^{24} + 2^{23} + 2^{22} + 2^{21} + 2^{18}) = -773.586.944$$

4.4.(0,8) um número em ponto flutuante que utiliza a representação padrão IEEE 754 para precisão simples, ou seja, um bit para o sinal (0 para positivos e 1 para negativos), 8 bits para o expoente representado em excesso de 127 e 23 bits para a parte fracionária. Os números a serem representados devem estar no seguinte formato: $N = +/-(1, b_{-1}b_{-2}b_{-3} \dots b_{-m})_2 \times 2^{\text{Expoente}}$.

$$(11010001 \ 11100100 \ 00000000 \ 00000000)_2$$

Sinal: 1 (negativo)

Expoente: 10100011 (excesso de 127) = $163 - 127 = 36$

Mantissa: 0,110010000000

$$\text{Resposta: } -1,11001 \times 2^{36} = -(2^{36} + 2^{35} + 2^{34} + 2^{31}) = -122.406.567.936$$

5. (2,0) Considere uma máquina cujo controlador de impressora possui dois registradores para se comunicar com o resto do sistema: um para receber comandos do sistema (pedido de impressão de um byte, por exemplo) e para indicar se a impressora está pronta para receber bytes, e outro para armazenar o byte a ser impresso. O primeiro registrador possui o endereço 08 e o segundo 16. O processo de impressão de um byte consiste no envio do comando 01 para o controlador de impressora, utilizando-se o endereço 08. O controlador de impressora armazena o valor 01 no registrador de estado (registrador 08) quando a impressora estiver pronta para receber um byte e o byte a ser impresso deve ser armazenado no registrador 16. Descreva **detalhadamente** os três possíveis métodos de comunicação entre o controlador de impressora com a unidade central de processamento e memória principal: por E/S programada, por interrupção e por acesso direto à memória. Você deve descrever estes métodos dentro do contexto do ambiente descrito. Não serão consideradas descrições gerais dos três métodos.

Resposta:

E/S programada:

A UCP envia o comando 01 para o registrador com endereço 08. Depois, fica lendo o conteúdo do registrador com endereço 08 e verificando se o seu conteúdo é igual a 01. Quando o conteúdo for igual a 01, a UCP envia, pelo barramento de dados, um byte para ser armazenado no registrador com endereço 16. Este procedimento é repetido até que todos os bytes tenham sido transferidos para a impressora.

E/S por interrupção:

A UCP envia o comando 01 para o registrador com endereço 08. Depois, vai executar outras instruções. Quando a impressora está pronta para receber um byte, ela envia um sinal de interrupção para a UCP. A UCP, ao receber o sinal de interrupção, finaliza a instrução que estiver executando, salva o contexto do programa que estava sendo executado e atende a interrupção, enviando o byte a ser impresso para ser armazenado no registrador de endereço 16. Este procedimento é repetido até que todos os bytes tenham sido transferidos para a impressora.

E/S por acesso direto à memória:

Neste caso, existe um controlador de DMA (Direct Access Memory) que é responsável pela transferência dos bytes. A UCP informa a este controlador que a operação de transferência deverá ser realizada com a impressora, o endereço inicial da memória onde se localizam os bytes a serem impressos, o número de bytes a serem transmitidos e a indicação de que o controlador deve transferir bytes da memória para a impressora. Após receber estas informações, o controlador realiza a transferência dos bytes entre memória e impressora sem a intervenção da UCP. Após a transferência de todos os bytes, o controlador avisa o fim da operação para a UCP através de um sinal de interrupção.