



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Organização de Computadores

GABARITO DA AP2 - 2º semestre de 2017

- 1) (2,0) Considere um computador, cuja representação para ponto fixo e para ponto flutuante utilize 12 bits. Na representação para ponto flutuante, o número a ser representado deve ser expresso na notação científica normalizada $(+/- (1, b_1 b_2 b_3 \dots b_7)_2 \times 2^{\text{expoente}})$. O bit mais à esquerda representa o sinal e deve ser 0 para números positivos e 1 para números negativos. O expoente está representado em complemento de 2. A representação é mostrada na figura abaixo.

S	expoente	$b_1 b_2 b_3 \dots b_7$
1 bit	4 bits	7 bits

- a) (1,0) Suponha que o conteúdo dos 12 bits seja $(AB8)_{16}$. Indique o valor **em decimal (pode deixar as contas indicadas)** para este conjunto de bits quando considerarmos que ele está representando:

$$(AB8)_{16} = (1010 \ 1011 \ 1000)_2$$

- i) (0,2) um inteiro sem sinal

$$+ 2^{11} + 2^9 + 2^7 + 2^5 + 2^4 + 2^3 = 2744$$

- ii) (0,3) um inteiro em complemento a 2

$$-2^{11} + (2^9 + 2^7 + 2^5 + 2^4 + 2^3) = -1352$$

- iii) (0,5) um número em ponto flutuante utilizando a representação do enunciado

1 0101 0111000

Sendo:

Sinal = 1 => negativo

Expoente = $0101_2 = +2^2 + 2^0 = +5$

Mantissa = 0111000

$$\text{Temos então } \Rightarrow -1,0111000 \times 2^{+5} = 101110_2 = +2^{+5} + 2^{+3} + 2^{+2} + 2^{+1} = +46$$

- b) (0,5) Qual o menor e o maior valor **positivo** normalizado na representação em ponto flutuante, descrita no enunciado, para este computador? Os valores devem ser representados **em decimal**.

Maior valor positivo : $0\ 0111\ 111111 = 1,111111 \times 2^{+7} = +1111111 = +127$

Menor valor positivo : $0\ 1000\ 000000 = 1,0000000 \times 2^{-8} = +0,00000001 = +0,00390625$

- c) (0,5) Mostre a representação em ponto flutuante do valor em decimal -19,25, na representação do enunciado para ponto flutuante.

$$-19,25 = -10011,01 = -1,001101 \times 2^{+4}$$

Sinal: 1 (negativo)

Expoente: $+4 = 0100_2$

Mantissa: 001101

Resposta: 1 0100 001101

- 2) (2,0) Considere uma máquina cujo controlador de teclado possui três registradores para se comunicar com o resto do sistema: o primeiro é utilizado para receber comandos do sistema (pedido de envio de um byte associado a uma tecla, por exemplo), o segundo para indicar se o controlador de teclado possui um byte associado à entrada de uma tecla para enviar para o sistema, e o terceiro onde é armazenado o byte associado a uma tecla pressionada pelo usuário do sistema. O primeiro registrador possui o endereço 31, o segundo 32 e o terceiro 33. O processo de recebimento de um byte associado a uma tecla pressionada pelo usuário consiste no envio do comando 02 para o registrador 31 do controlador de teclado, indicando que o sistema deseja receber um byte associado a uma tecla. O controlador de teclado armazena o valor 02 no registrador de estado 32 quando o controlador de teclado possuir um byte armazenado referente a uma tecla pressionada pelo usuário. Este byte fica armazenado no registrador 33. Descreva detalhadamente os três possíveis métodos de comunicação entre o controlador de teclado 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 neste enunciado. Não serão consideradas descrições gerais dos três métodos.

E/S programada:

A UCP envia o comando 02 para o registrador de endereço 31 do controlador do teclado. Depois, fica lendo o conteúdo do registrador de endereço 32 e verificando se o seu conteúdo é igual a 02. Quando o conteúdo for igual a 02, a UCP envia, pelo barramento de dados, o byte (que corresponde a tecla pressionada no teclado) para o registrador do controlador com endereço 33.

E/S por interrupção:

A UCP envia o comando 02 para o registrador de endereço 31. Depois, vai executar outras instruções. Quando o controlador do teclado está pronto para receber o byte, ele envia um sinal de interrupção para a UCP. Esta, 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, encaminhando o byte (que corresponde a tecla pressionada no teclado) para o registrador de endereço 33 do controlador do teclado.

E/S por acesso direto à memória:

Neste caso, existe um controlador de DMA (Direct Access Memory) que é responsável pela transferência do(s) byte(s). A UCP informa a este controlador que a operação de transferência deverá ser realizada com o controlador do teclado. Informa também o endereço inicial da memória onde está(ão) os byte(s) a ser(em) enviado(s), a quantidade de bytes e a indicação de que o controlador de DMA deve transferir o(s) byte(s) da memória para o controlador do teclado. Após receber estas informações, o controlador de DMA realiza a transferência do(s) byte(s) entre a memória e o controlador do teclado sem a intervenção da UCP. Após a transferência, o controlador de DMA avisa o fim da operação para a UCP através de um sinal de interrupção.

- 3) (3,0) Crie um conjunto de instruções de um operando definidas em Linguagem Assembly, necessárias para a realização de operações aritméticas e elabore um programa para o cálculo da equação abaixo.

a) $X = B * (D - E/B) + A$

I_ CONJUNTO DE INSTRUÇÕES PARA 1 OPERANDO:

ADD X	=> (ACC) <- (ACC) + (X)
SUB X	=> (ACC) <- (ACC) - (X)
MUL X	=> (ACC) <- (ACC) * (X)
DIV X	=> (ACC) <- (ACC) / (X)
LOAD X	=> (ACC) <- (X)
STORE X	=> (X) <- (ACC)

$$X = B * (D - E/B) + A$$

LOAD E	=> (ACC) <- (E)
DIV B	=> (ACC) <- (ACC) / (B)
STORE T1	=> (T1) <- (ACC)
LOAD D	=> (ACC) <- (D)
SUB T1	=> (ACC) <- (ACC) - (T1)
MUL B	=> (ACC) <- (ACC) * (B)
ADD A	=> (ACC) <- (ACC) + (A)
STORE X	=> (X) <- (ACC)

- 4) (3,0) Responda as questões abaixo:

- a) Explique Compilação e Interpretação, descrevendo vantagens e desvantagens de cada modo de tradução.

A compilação consiste na análise de um programa escrito em linguagem de alto nível (programa fonte) e sua tradução para um programa em linguagem de máquina (programa objeto).

Na interpretação cada comando do código fonte é lido pelo interpretador, convertido em código executável e imediatamente executado antes do próximo comando.

A interpretação tem como vantagem sobre a compilação a capacidade de identificação e indicação de um erro no programa-fonte (incluindo erro da lógica do algoritmo) durante o processo de conversão do fonte para o executável.

A interpretação tem como desvantagem o consumo de memória devido ao fato de o interpretador permanecer na memória durante todo o processo de execução do programa. Na compilação o compilador somente é mantido na memória no processo de compilação e não utilizado durante a execução. Outra desvantagem da interpretação está na necessidade de tradução de partes que sejam executadas diversas vezes, como os loops que são traduzidos em cada passagem. No processo de compilação isto só ocorre uma única vez. Da mesma forma pode ocorrer para o programa inteiro, em caso de diversas execuções, ou seja, a cada execução uma nova interpretação.

b) Explique Sistemas SMP e Sistemas NUMA.

Sistemas SMP (ou UMA) têm como característica o acesso a todas as partes da memória principal com tempo de acesso uniforme. Em sistemas NUMA, todos os processadores possuem também acesso a todas as partes da memória principal podendo diferir o tempo de acesso em relação às posições da memória e processador.

Nos sistemas SMP o aumento no número de processadores tem como consequência problemas de tráfego no barramento comum degradando o desempenho. Uma solução para isto é a utilização de clusters, que tem, usualmente, como consequência alterações significativas na aplicação (software). Nos sistemas NUMA podem-se ter vários nós multiprocessadores, cada qual com seu próprio barramento, resultando em pequenas alterações na aplicação (software).