MC404 Organização de Computadores e Linguagem de Montagem

IC – UNICAMP

Primeira Lista de Exercícios

1. Converta os seguintes números binários para hexadecimal
   (a) 100011100011
   (b) 1111000011001010
   (c) 100000011110011001
2. Converta os seguintes números hexadecimal para binário
   (a) 5A
   (b) F81
   (c) 3C0B
3. Converta os seguintes números hexadecimal para decimal
   (a) 35A
   (b) 1001
   (c) 9CD4
4. Determine a representação em complemento de dois e em sinal e magnetude dos seguintes números decimais, supondo números de 16 bits (mostre o resultado em binário e em hexadecimal)
   (a) -151
   (b) 32760
   (c) -32000
5. Escreva uma sub–rotina *Mult* que multiplica dois inteiros passados como parâmetros na pilha, devolvendo o resultado em um parâmetro (na pilha), através do método *multiplicação rápida*, como mostra o procedimento em Pascal a seguir:

```
procedure Mult(a,b:integer; var p:integer);
 { p = a * b; a >= 0; b >= 0}
begin
  p := 0;
  while a > 0 do
    begin
      if odd(a) then p := p + b;
      a := a div 2;
      b := b * 2
    end
end; { Mult }
```

6. Escreva um procedimento que receba como parâmetro, pela pilha, o endereço de um vetor de inteiros sem sinal, cujo comprimento é também passado como parâmetro pela pilha. O procedimento deve retornar no registrador `AX` o número de elementos ímpares do vetor e em `BX` a somatória dos elementos pares.

7. Escreva uma sub–rotina de comparação de inteiros de 32 bits, **sem sinal**, chamada `CMP32` que deve comparar 2 valores de 32 *bits*, X e Y passados como parâmetros na **pilha** o resultado deve retornar nos *flags* `C` e `Z`, da seguinte forma:

$$\begin{aligned} \mathtt{C} &= 1 \text{ se } \mathtt{X} < \mathtt{Y} \\ \mathtt{C} &= 0 \text{ se } \mathtt{X} >= \mathtt{Y} \\ \mathtt{Z} &= 1 \text{ se } \mathtt{X} = \mathtt{Y} \\ \mathtt{Z} &= 0 \text{ se } \mathtt{X} <> \mathtt{Y} \end{aligned}$$

   Usando a sub–rotina acima, escreva uma sub–rotina `VER32` que verifica se um inteiro de 32 *bits*, sem sinal ocorre em um vetor. O número de elementos e o endereço inicial do vetor e o valor a ser testado são passados como parâmetros na pilha.

8. Supondo que você tenha uma lista ligada na memória, como mostrado na figura abaixo. O endereço do primeiro elemento da lista está na memória, no endereço APTLIST.

Info	Prox

   Onde:

Info (16 bits)	informação do elemento
Prox (16 bits)	offset (em relação à DS) do endereço do próximo elemento da lista

   Escreva uma rotina que “deleta” da lista o nó cujo elemento Info é igual ao conteúdo de AX e desligue o bit de Carry (0). Se nenhum elemento da lista satisfizer à condição, nada é feito e o Carry é ligado (1)

9. Escreva uma rotina que calcula o número de verificação (checksum) de uma seqüência não vazia de valores de um byte, cujo endereço inicial (offset em relação a DS) está no registrador BX e o tamanho no registrador CX. O cálculo do checksum é realizado através de operações ou–exclusivo entre os valores da seqüência, o resultado deve retornar em Dl. Exemplo:

$$DL \ = \ 28H \ \oplus \ 55H \oplus \ 26H \ = 5BH$$

10. Escreva uma rotina que soma dois números em BCD com vários dígitos. O número de bytes (dois dígitos BCD) que compõem os números é um parâmetro (passado na pilha). Os outros dois parâmetros (também passados na pilha) são apontadores para os Bytes menos significativos dos dois operandos. O resultado da soma será armazenado no primeiro operando. OBS.: Nome da subrotina MADD

11. Implemente uma subrotina chamada MSUB que realiza operação de subtração análoga à apresentada para a rotina MADD (exercício anterior) com os mesmos parâmetros. O resultado da operação também deve ficar no primeiro operando.

12. Escreva uma rotina que procura um dado padrão de caracteres em um vetor de caracteres. Os endereços do vetor de caracteres e do padrão são passados como parâmetros na pilha e ambos terminam com o caracter NUL (valor 0H). Caso o padrão ocorra no vetor o endereço do primeiro caracter do padrão no vetor deve retornar no registrador BX, caso contrário deve-se retornar o valor 0H em BX.

13. Escreva uma rotina que converta 4 dígitos BCD presentes no registrador AX no número binário correspondente. O resultado deve retornar no próprio registrador AX e deve-se preservar os valores de todos os outros registradores.

14. A conversão de dígitos hexadecimais para caracteres ASCII pode ser realizadda utilizando-se uma tabela (vetor) de conversão, como indicado abaixo: //

```
DigitoH[0] = '0'
DigitoH[1] = '1'


DigitoH[F] = 'F'
```

    Escreva uma rotina que faz a conversão Hexadecimal para ASCII usando essa abordagem.

15. Escreva uma rotina que encontre o maior e o menor elemento e calcule o valor médio de uma série de valores, cujo endereço do primeiro valor(segmento e offset) é dado na posição de memória dado pelo rótulo SERIE e termina com o valor 0H.

    (a) Os valores da série são inteiros sem sinal

    (b) Os valores da série são inteiros com sinal (em complemento de 2)