Universidade Federal do Rio Grande do Norte
Centro de Tecnologia
Departamento de Engenharia de Computação e Automação

Exercício de Programação Assembly Isso

Disciplina: DCA3404 - Arquitetura de Computadores
Turma: 1, 46T12, sala 4I2, 2024.2
Professor: Diogo Pinheiro Fernandes Pedrosa (diogo.pedrosa@ufrn.br)

Problema

O vetor a seguir consiste em uma sequência de 42 números inteiros de 4 *bytes* de tamanho, todos representados no formato hexadecimal.

V = [0x00051010, 0x116A23B1, 0x21347582, 0x10061231, 0x11642467, 0x008695AB, 0x21CD6EEF, 0x00071323, 0x11264517, 0x2089A2B0, 0x00E5F601, 0x212360F1, 0x11624533, 0x21676455, 0x00627089, 0x20AB8691, 0x10A6CDB3, 0x21EF6C5D, 0x10E701F2, 0x00071423, 0x0162F345, 0x21677455, 0x10628971, 0x1082AB90, 0x10A4CDB6, 0x016C9DEF, 0x21016031, 0x212362F3, 0x01745545, 0x10626770, 0x10868993, 0x21AB6AFB, 0x00C6DDCD, 0x00E2F0EF, 0x116001E1, 0x0162F323, 0x20454754, 0x00667167, 0x20898290, 0x113AAB1B, 0x113CCD0D, 0x000211EF]

Esta sequência contém uma mensagem (com caracteres ascii) inserida nos dígitos hexadecimais em cada um dos elementos do vetor (ou seja, cada número de 4 *bytes* tem um caractere ascii escondido).

O objetivo principal deste exercício é desenvolver um programa que analise cada um dos 42 números apresentados e extraia os caracteres escondidos, exibindo-os em tela.

Os objetivos secundários deste exercício são a prática de programação em linguagem de montagem, com instruções que permitam acesso à memória, manutenção de laços, operações lógicas diversas e chamadas de sistema para saída de dados padrão.

Para realizar a decodificação, é necessário saber que dada *word* tem seus *bytes* classificados do mais significativo (à esquerda) para o menos significativo (à direita), numerados de 3 até 0.

[*byte* 3][*byte* 2][*byte* 1][*byte* 0]

Cada *byte* é formado por dois *nibbles* de 4 bits, numerados como 1 (mais significativo) e 0 (menos significativo:

[*byte*] = [*nibble* 1][*nibble* 0]

Na codificação, o *byte* 3 (mais significativo da *word*) contém as informações iniciais. O *nibble* 1 informa qual dos outros três *bytes* da *word* não contém informação válida. Assim sendo, ele assume, somente, os valores 0, 1 ou 2. Já o *nibble* 0 do *byte* 3 indica quais *nibbles* dos *bytes* válidos devem ser extraídos e concatenados para formar a informação de 8 bits que representa o caractere ascii desejado. Assim sendo, ele somente pode assumir os valores 1 e 0.

Por exemplo, consideremos a *word* 0x21AB6AFB, em hexadecimal. Ela é formado pelos 4 *bytes*:

[21][AB][6A][FB]

onde:

- *Byte* 3 = 0x21
- *Byte* 2 = 0xAB
- *Byte* 1 = 0x6A
- *Byte* 0 = 0xFB

O *byte* 3 tem, como *nibbles*, os valores 2 e 1 em hexadecimal, respectivamente. Como o *nibble* 1 é igual a 2, então o *byte* 2 não tem informação útil na codificação (descarta-se, assim, o valor 0xAB). Já o *nibble* 0 tem valor 1. Desse modo, deve-se extrair os *nibbles* 1 dos *bytes* 1 e 0, concatená-los e, desse modo, exibir o caractere obtido desta concatenação:

- *Byte* 2 descartado!
- *Byte* 1 = 0x6A → *nibble* 1 = 0x6 e *nibble* 0 = 0xA (descartado!)
- *Byte* 0 = 0xFB → *nibble* 1 = 0xF e *nibble* 0 = 0xB (descartado!)

Concatenando os *nibbles* 0x6 e 0xF, na ordem de suas significâncias dos *bytes* (*bytes* 1 e 0, respectivamente), temos o valor 0x6F (ou valor 111, em decimal), que representa o caractere “o”.

Observações

- Esta é uma tarefa individual;
- É disponibilizado um arquivo, chamado “template.asm”, que contém a especificação dos valores em hexadecimal. Pode renomear o arquivo, caso deseje, completando-o com as instruções necessárias para a execução do programa;
- O envio da tarefa será, apenas, o código no formato .ASM. É obrigatório que você coloque seu nome e matrícula (como comentário, no código do arquivo) e que faça os devidos comentários para explicar o que ocorre nos vários trechos do código de baixo nível. Este envio será pelas Tarefas, no SIGAA da turma virtual;
- Lembre-se de testar o código no MARS antes de enviá-lo pelo sistema. Com isso, evita-se que, caso ainda exista algum erro, ele possa ser corrigido antes da submissão.
- Quaisquer dúvidas, enviem mensagem para diogo.pedrosa@ufrn.br.