**Infra-Estrutura de Hardware**

**Lista de Exercícios - Aula Prática do MipsIt - 2009.1**

**Data para entrega: 25/03/2009**

**Emails para entrega:** [**ensb@cin.ufpe.br**](mailto:ensb@cin.ufpe.br) **e** [**jpfernandes83@gmail.com**](mailto:jpfernandes83@gmail.com)

1- Escreva um programa em linguagem de montagem do MIPS que receba dois números inteiros armazenados na memória e realiza a multiplicação dos dois números. Considere números positivos e negativos. A instrução **mult ou semelhante** não deverá ser utilizada na implementação dessa questão. O resultado deverá ser armazenado em uma variável na memória e o seu nome deverá ser “resultado”.

2 - Escreva um programa em linguagem de montagem do MIPS no qual, dados dois números inteiros x e y, calcule as operações de módulo (x%y) e divisão inteira (x/y) entre os dois números. Os resultados deverão estar disponíveis nas variáveis na memória “modulo” e “divisao”, respectivamente. Não é permitido o uso de qualquer instrução do Mips que faça o módulo ou a divisão dos números de entrada.

3 - Faça um programa em linguagem de montagem Mips que receba como entrada uma string com n caracteres e gere como saída uma nova string com a inversão da ordem dos caracteres. Essa nova string também terá a troca das letras maiúsculas por minúsculas e vice-versa. Por exemplo: se a entrada for HArdwArE a saída deverá ser eRaWDRah. A entrada deve ser lida da memória e a saída deve ser escrita na memória. Caso a string possua algum caractere que não seja letra o valor 1 deverá ser armazenado no registrador v1 e o programa deverá ser encerrado. Lembramos que o fim da string é dado pelo caractere nulo e que para manipular string nessa questão recomendamos que sejam usadas variáveis do tipo ASCIIZ.

***Atenção: os caracteres deverão ser armazenados em seqüência na memória.***

4 - Escreva um programa em linguagem de montagem Mips que recebe uma string de caracteres e um caractere. Tal programa deverá contar a quantidade de vezes que o caractere aparece na string. Todas as entradas serão feitas através da memória e a saída deverá ser dada através da variável “num-caracteres”.

***Exemplo:***

***Para as entradas “Protocolo” e “o” teremos após a execução do programa teremos como resultado num-caracteres = 4***

5 - Escreva um programa em linguagem de montagem do MIPS que calcule o número de palavras de um texto. O texto deverá ter sido declarado como uma string “texto”, e o número de palavras será uma variável inteira “num-palavras”. Todas as variáveis deverão estar armazenadas na memória.

6 - Escreva um programa em linguagem de montagem Mips que receba uma string de caracteres como entrada. Tal string deverá representar um número em notação hexadecimal (os caracteres poderão ser maiúsculos ou minúsculos, seu programa deve tratar isso) e terá tamanho fixo (quatro caracteres não incluindo o caractere nulo). Seu programa deverá convertê-la em sua representação binária e armazenar o resultado da conversão no registrador v0. Caso algum caractere inválido esteja presente na string de entrada o registrador v0 terá valor 0 e v1 terá valor 1.

***Exemplo:***

***Para a entrada “02Fa” teremos v0 = 00000000000000000000001011111010***

***Para a entrada “5FhB” teremos v0 = 0 e v1 = 1***

7 - Escreva um programa em linguagem de montagem do Mips que converta uma string de caracteres (essa string contém apenas caracteres que representam números) na sua representação binária. Números positivos e negativos devem ser considerados. O número de entrada será uma string de tamanho indefinido e será lido da memória. O resultado deverá ser armazenado no registrador v0. Caso o número a ser convertido não seja válido, o registrador v1 retornará o valor 1.

***Exemplo:***

***Para a entrada “956” o resultado em v0 será 00000000000000000000001110111100.***

***Para a entrada “32z” v0 = 0 e v1 = 1.***

8 - Faça um programa em linguagem de montagem Mips que receba como entrada dois números, n e s, e que tenha como saída o resultado da combinação de n tomados s a s. Os números n e s devem ser carregados da memória e o resultado da combinação deve ser colocado na variável “resultado” que também deverá estar na memória. Caso n e/ou (n - s) seja menor que zero, o valor 1 deve ser armazenado no registrador v1. Segue abaixo a fórmula da combinação.

******

***Bom Trabalho!***