

Lista 1 2019.2  
Infraestrutura de Hardware  
Ciência da Computação

Aula Prática ASSEMBLY MIPS

**Atenção: o código de todas as questões deverá estar claramente comentado, pois caso contrário a correção será muito dificultada. É de seu interesse colaborar com a correção!**

**Obs: os exercícios 1 e 2 deverão ser entregues no mesmo dia da aula na atividade do Classroom.**

**Os demais exercícios (3 a 8) deverão ser entregues até a data marcada no calendário como entrega da lista.**

1. Codifique um programa correspondente ao seguinte pseudo-código:

```
int a = ?  
int b = ?  
int r = 0  
  
if (a>0 && b>0)  
    r = 1  
else if (a<0 && b>0)  
    r = 2  
else  
    r = 3
```

2. Codifique um programa correspondente a:

```
int a = ?  
int b = ?  
int x = 0  
if ( a >= 0 && a < b )  
    x = 1;  
else if( a < 0 && a > b )  
    x = 2;  
else  
    x = 3;
```

3. 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 não deverá ser utilizada na implementação dessa questão. O resultado deverá ser armazenado em uma variável RESULT na memória.

4. Escreva um programa em linguagem de montagem do MIPS que receba três números inteiros armazenados na memória e realiza a definição se aqueles três números podem ser comprimentos de lados de um triângulo, e caso sejam, defina qual tipo de triângulo ele é, armazenando numa variável **r** o valor 0 caso não sejam comprimentos de lados de um triângulo, 1 para equilátero, 2 para isósceles, 3 para escaleno. Lembrando que para três números serem os comprimentos de um lado de um triângulo um dos lados deve ser menor que a soma dos dois outros.

5. Leia uma string da memória transforme as letras em letras minúsculas e coloque seus caracteres em ordem alfabética decrescente e salve-os (já ordenados) no mesmo lugar da memória. Se houver algum caractere que não seja letra, o valor 1 deverá ser armazenado no registrador **v1** e o programa encerrado.

Exemplo:

entrada - "EntradA"

saída - "aadenrt"

6. Desenvolva um programa que recebe como entrada um array de inteiros **a** e escreve em um outro array **b** todos os números primos contidos em **a**.

Ex.:  $a = [2, 4, 7, 17, 25] \rightarrow b = [2, 7, 17]$

Obs.: assumo que só serão testados números no intervalo  $[1, 100]$ .

7. Implemente (recursivamente) na linguagem de montagem do MIPS, uma função que receba dois números **a** e **b** ( $b \geq a$ ) e retorne a soma de todos os números contidos no intervalo  $[a, b]$ . Se  $a > b$ , o valor 1 deverá ser armazenado no registrador **v1** e o programa encerrado.

8. Implemente a função de Fibonacci, de forma que ela receba um inteiro **N** como parâmetro e retorne o número de Fibonacci do mesmo.

Use recursão.

Sequência de Fibonacci:

$F_1 = 1, F_2 = 1$

$F_n = F_{n-1} + F_{n-2}$