

# Lista 1

## Aula prática Assembly MIPS 2021.1

### 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, e é de seu interesse colaborar com a correção!

**Obs:** os exercícios 1, 2 e 3 deverão ser entregues no mesmo dia da aula na atividade do Classroom (apenas um arquivo compactado por grupo com o número da lista e número do grupo).

Os demais exercícios (4 a 11) deverão ser entregues até a data marcada no calendário como entrega da lista (apenas um arquivo compactado por grupo com o número do mesmo)

1. **(0,75)** Codifique um programa correspondente ao seguinte pseudo-código:

```
int idadeJoao = ? [valor entre 0,100]
int idadePedro = ? [valor entre 0,100]
if (idadeJoao > idadePedro) {
    print("João é mais velho que Pedro")
} else if (idadePedro > idadeJoao) {
    print("Pedro é mais velho que João")
} else {
    print("Eles têm a mesma idade")
}
```

2. **(0,75)** Codifique um programa correspondente ao seguinte pseudo-código:

```

int a = ? [valor entre 0,100]
int b = ? [valor entre 0,100]
if (a == 0) {
    STORE b;
} else {
    while (b != 0) {
        if (a > b) {
            a = a - b
        } else {
            b = b - a
        }
    }
    STORE a;
}

```

3. Responda:

- A. **(0,25)** O que é ISA?
- B. **(1,0)** Como se deu a evolução das ISAs e quais os motivos da popularização da ISA do MIPS.

4. **(0,55)** Codifique um programa correspondente ao seguinte pseudo-código:

```

int A[10] = ?;
for (int i=0; i<10; i++) {
    if (i%2==0)
        A[i]=A[i]+A[i+1];
    else
        A[i]=A[i]*2;
}

```

5. **(0,55)** 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. **(0,55)** Escreva um programa em linguagem de montagem do MIPS que receba dois números inteiros armazenados na memória e realize 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.

7. **(0,8)** 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 o resultado numa variável r, 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 outros.

8. **(0,8)** 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] → b = [2,7,17]

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

9. **(1,25)** 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.

10. **(1,25)** 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:

$$F1 = 1, F2 = 1$$

$$Fn = Fn-1 + Fn-2$$

11.Responda:

- a. **(0,5)** Qual a diferença entre Arquitetura e Organização de um computador?
- b. **(0,5)** Dê exemplos de tais “partes”.
- c. **(0,5)** Qual a diferença entre Big Endian e Little Endian? E como são armazenados na memória?