

# TRABALHO DE ASSEMBLER

**DATA DE ENTREGA: 16/11/2010**

Cada grupo deverá fazer o programa especificado, **entregá-lo impresso e na forma digital, seguindo as seguintes regras:**

- Comentar todas as linhas
- Comentar os procedimentos (entradas e saídas)
- Fazer uma “legenda” dos registradores (# \$s0 armazena o endereço base do array mat, por exemplo) antes de começar o programa.
- Poderá ocorrer arguição aos componentes do grupo sobre o trabalho.

## **Grupo 1:**

**Fibonacci:** que armazena em um array[100] os valores da série de Fibonacci

(0,1,1,2,3,5,8,13,21,34,...) enquanto o termo for menor que 5000.

## **Grupo 2**

**Serie\_h:** que recebe um valor de N e calcula o valor de H:

$$H = 1/1 + 1/2 + 1/3 + 1/4 + 1/5 + \dots + 1/N$$

## **Grupo 3**

**Serie\_s:** que recebe um valor correspondente ao termo da série e calcula o valor de S:

$$S = 1 + 2 - 3 + 4 - 5 + 6 \dots N$$

## **Grupo 4**

**Serie\_pot:** que recebe um inteiro x, um inteiro y e retorna de forma recursiva x elevado a potência y.

## **Grupo 5**

**Soma\_serie:** que recebe três inteiros (i, j e k) e devolve a soma da série de valores do intervalo [i, j], com incremento k

## **Grupo 6**

**MDC:** que receba dois valores inteiros e que devolva o maior divisor comum aos dois valores recebidos

## **Grupo 7**

**Ordenacao:** dado um array de inteiros mat1[5] ordenar o array em ordem crescente.

## **Grupo 8**

**Soma\_igual\_z:** dado um vetor: mat[10] já ordenado em ordem decrescente e um inteiro z, informar quantas vezes existe dois valores x e y tais que  $x + y = z$ .