

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACOM - FACULDADE DE COMPUTAÇÃO
SISTEMA DE INFORMAÇÃO

LUCAS ALBINO MARTINS

Matrícula: 12011ECP022

TRABALHO 04: Macros, memória e laços.

Disciplina: Arquitetura e Organização de Computadores.

Uberlândia
2020

Escrever um programa no MARS que leia os elementos de um array de 10 posições e salve estes números no array. A seguir o programa deve calcular a média dos 10 elementos do array e então imprimir o valor computado. Utilize aritmética inteira e macros para as partes do programa referentes a chamadas do sistema.

```
#-----  
  
#media.asm  
  
#  
#DDA 03.09.2020  
  
#  
#Aluno: Lucas Albino Martins, Matricula 12011ECP022  
  
#  
#DESC: programa que soma 10 valores adicionados em um vetor  
# e imprimir a média  
#  
#-----  
  
.data  
array: .word 1, 2, 3, 4, 5, 6, 7, 8, 9, 15  
nelem: .word 10  
soma: .word 0  
media: .word 0  
imprimimedia: .asciiz "\nO valor da média eh: "  
  
.text  
.globl main  
main:  
  
# loop para calcular a soma dos valores
```

la \$t0, array # endereço inicial do array

li \$t1, 0 # loop do índice, i=0

lw \$t2, nelem

li \$t3, 0 # inicializa a soma=0

loop:

lw \$t4, (\$t0) # pega o valor do array[i]

add \$t3, \$t3, \$t4 # soma = soma + array[i]

add \$t1, \$t1, 1 # i = i+1

add \$t0, \$t0, 4 # atualiza o endereço do array

blt \$t1, \$t2, loop # condição de i < nelem, continua loop

sw \$t3, soma # salva a soma

calculando a média.

div \$t5, \$t3, \$t2 # ave = soma / nelem

sw \$t5, media

imprimindo a média

li \$v0, 4

la \$a0, imprimimedia

syscall

li \$v0, 1

addi \$a0, \$t5, 0

syscall

finaliza o programa.

```
li $v0, 10
```

```
syscall
```

Escreva um programa no MARS que calcule a série de Fibonnaci para N. O valor N deve ser lido do teclado e a série computada impressa no terminal.

```
#-----  
#seriefibonacci.asm  
#  
#DDA 03.09.2020  
#  
#Aluno: Lucas Albino Martins, Matricula 12011ECP022  
#  
#DESC: programa calcula a serie fibonacci e imprime o valor  
#  
#-----  
  
.data  
srt1: .asciiz "Digite o valor N valor da serie Fibonacci: "  
str2: .asciiz "\n A "  
str3: .asciiz " posição da Sequência Fibonacci é "  
str4: .asciiz " "  
.text  
  
# t0 = a, t1 = b, t2 = FIB, t3 = i, t4 = n, t5 = aux  
li $t0,0 # t0 = a  
li $t1,1 #t1 = b  
li $t2,0 #t2 = FIB  
li $t3,0 #t3 = i
```

```
la $a0,srt1
```

```
li $v0,4
```

```
syscall
```

```
li $v0,5
```

```
syscall
```

```
move $t4,$v0
```

```
# calcula o fribonacci pelo laço
```

```
FOR:
```

```
beq $t3, $t4, PRINT
```

```
# t5 = aux
```

```
li $t5,0
```

```
# aux = aux + a
```

```
add $t5,$t5,$t0
```

```
# aux = aux + b
```

```
add $t5,$t5,$t1
```

```
# FIB = aux
```

```
move $t2,$t5
```

```
# a = b
```

```
move $t0,$t1
```

```
# b = FIB
```

```
move $t1,$t2
```

```
addi $t3,$t3,1
```

```
j FOR
```

```
# carregando os endereços da frase do msg para syscall
```

```
PRINT:
```

```
la $a0,str2
```

```
# especifica o serviço de Exibição de String
```

```
li $v0,4
```

```
syscall
# carregando os endereços da frase do msg para syscall
move $a0,$t4
# especifica o serviço de Exibição de inteiro
li $v0,1
syscall
# carregando os endereços da frase do msg para syscall
la $a0,str3
# especifica o serviço de Exibição de String
li $v0,4
syscall
la $a0,str4
# carregando os endereços da frase do msg para syscall
li $v0,4
# especifica o serviço de Exibição de String
syscall
# carregando os endereços da frase do msg para syscall
move $a0,$t0
# especifica o serviço de Exibição de inteiros
li $v0,1
syscall
```

Escreva um programa no MARS que leia uma string do teclado e também uma palavra chave. O Programa deve então pesquisar a string para encontrar o ponto inicial e final da palavra chave na string.

```
#-----  
#media.asm  
#  
#DDA 03.09.2020  
#  
#Aluno: Lucas Albino Martins, Matricula 12011ECP022  
#  
#DESC: programa que soma 10 valores adicionados em um vetor  
# e imprimir a média  
#  
#-----  
  
.data  
array: .word 1, 2, 3, 4, 5, 6, 7, 8, 9, 15  
nelem: .word 10  
soma: .word 0  
media: .word 0  
imprimimedia: .asciiz "\nO valor da média eh: "  
  
.text  
.globl main  
main:  
  
# loop para calcular a soma dos valores  
  
la $t0, array # endereço inicial do array
```

li \$t1, 0 # loop do indice, i=0

lw \$t2, nelem

li \$t3, 0 # inicializa a soma=0

loop:

lw \$t4, (\$t0) # pega o valor do array[i]

add \$t3, \$t3, \$t4 # soma = soma + array[i]

add \$t1, \$t1, 1 # i = i+1

add \$t0, \$t0, 4 # atualiza o endereço do array

blt \$t1, \$t2, loop # condição de i < nelem, continua loop

sw \$t3, soma # salva a soma

calculando a média.

div \$t5, \$t3, \$t2 # media = soma / nelem

sw \$t5, media

imprimindo a média

li \$v0, 4

la \$a0, imprimimedia

syscall

li \$v0, 1

addi \$a0, \$t5, 0

syscall

finaliza o programa.

li \$v0, 10

syscall