



Universidade Federal Rural do Semiárido
Centro de Ciências Exatas e Naturais
Departamento de Computação
Ciência da Computação
Prof. Sílvio Fernandes

1.1 – Lista de Exercícios de Arquitetura de Computadores (Assembly MIPS)

1. Dado o seguinte programa:

```
.data
dados: .byte 3 # inicializo uma posição de memória a 3
.text
.globl main # deve ser global
main: lw $t0,dados($0)
```

Indique as etiquetas (também podem ser chamados de rótulos ou *labels*),
diretivas e comentários.

2. Execute o seguinte programa no MARS:

```
.data
msg1: .asciiz "Digite um numero inteiro: "
.text
.globl main
# No interior do main existem algumas chamadas (syscalls) que
# irão alterar o valor do registo $ra o qual contém inicialmente
# endereço de retorno do main. Este necessita de ser guardado.
main: addu $s0, $ra, $0 # guardar o registo $31 em $16
li $v0, 4 # chamada sistema print_str
la $a0, msg1 # endereço da string a imprimir
syscall
# obter o inteiro do utilizador
li $v0, 5 # chamada sistema read_int
syscall # coloca o inteiro em $v0
# realizar cálculos com o inteiro
addu $t0, $v0, $0 # mover o número para $t0
sll $t0, $t0, 2 #
# imprimir o resultado
li $v0, 1 # chamada sistema print_int
addu $a0, $t0, $0 # mover o número a imprimir para $a0
syscall
# repôr o endereço de retorno para o $ra e retornar do main
addu $ra, $0, $s0 # endereço de retorno de novo em $31
jr $ra # retornar do main
```

- a) Descreva o que o programa faz
- b) Crie um novo programa alterando a primeira linha que começa com a etiqueta main para:

```
label1: li $v0, 4 # chamada sistema print_int
```

- c) Usando o passo-a-passo do simulador, preencha a seguinte tabela:

Etiqueta	Endereço (PC)	Instrução Nativa	Instrução Fonte

- d) Agora iremos aprender a colocar *breakpoints* no programa. Vamos supor que queremos parar a execução do programa na instrução [0x00400040] sll \$t0,\$t0,2

Após compilar o *assembly* no MARS, na aba *Execute*, verifique o endereço onde esta instrução reside e adicione um *breakpoint* nesse endereço. Para isso clique no quadrado do lado esquerdo do endereço onde está a instrução. Execute o programa. O programa inseriu um break antes de executar a referida instrução. Anote o valor do registrador \$t0 antes da execução da instrução sll. Agora execute passo-a-passo. Qual o novo valor do registo \$t0?

3. Crie um programa que defina um vetor de 5 palavras associado à etiqueta *vector* que comece no endereço 0x10000000 e que contenha os valores 10,11,12,13,14. Certifique-se que o vetor é armazenado corretamente na memória. Dica: para especificar o endereço de início, escreva o valor desejado logo após a diretiva *.data*.
4. Crie o seguinte programa:

```
.data
pal1: .word 0x10203040
pal2: .space 4
pal3: .word 0xffffffff
.text
main: lw $s0,pal1($0)
      sw $s0,pal2($0)
      sw $s0,pal3($0)
```

- a) Verifique o que o programa faz
- b) Altere o programa para:

```
.data
pal1: .word 0x10203040
pal2: .space 4
pal3: .word 0xffffffff
.text
main: lw $s0,pal1
      sw $s0,pal2
      sw $s0,pal3+0
```

- c) Que diferenças existem entre as duas versões?
-
- 5. Escrever um programa em assembly com 4 variáveis (n1, n2, n3 e n4) e teste no MARS. Leia as 3 primeiras variáveis. Faça um somatório delas. Teste se o valor é maior ou igual a 21. Se for, imprima “O valor X atingiu o esperado com 3 valores” (onde X é o seu resultado). Caso contrário, leia a 4ª. variável e some ao resultado anterior e faça um novo teste: Teste se o somatório é maior ou igual a 24. Se for imprima “O valor X atingiu a meta com 4 valores”. Senão imprima “Não foi dessa vez. O valor X está abaixo do esperado”. Leitura e escrita das variáveis devem ser feita com LW e SW.
 - 6. Escreva um programa em *assembly* que solicite o nome e a data de nascimento do usuário (por meio de *syscalls*) e calcule sua idade. Em seguida escreva o nome do usuário indicando se ele é criança, jovem, adulto ou idoso. Para classificar o usuário por sua idade utilize os critérios: 0-10 anos criança; de 11-19 anos, jovem; 20-59 anos, adulto; maior ou igual a 60, idoso.
 - 7. Escreva um programa em *assembly* que obtenha dois números inteiros e por meio de laços de repetição implemente um algoritmo que calcule a multiplicação desses números usando apenas somas. No final imprima os números e o resultado da multiplicação entre eles.