

2 - Instruções: A linguagem de Máquina

2.21 Exercícios¹

¹ Contribuição de John Oliver, da Cal Poly, San Luis Obispo, com colaborações de Nicole Kaiyan (Universidade de Adelaide) e Milos Prvulovic (Georgia Tech)

O Apêndice B descreve o simulador do MIPS, que é útil para estes exercícios. Embora o simulador aceite pseudoinstruções, tente não usá-las em qualquer exercício que pedir para produzir código do MIPS. Seu objetivo deverá ser aprender o conjunto de instruções MIPS real, e se você tiver de contar instruções, sua contagem deverá refletir as instruções reais executadas, e não as pseudoinstruções.

Existem alguns casos em que as pseudoinstruções precisam ser usadas (por exemplo, a instrução `la` quando um valor real não é conhecido durante a codificação em assembly).

Em muitos casos, elas são muito convenientes e resultam em código mais legível (por exemplo, as instruções `li` e `move`). Se você decidir usar pseudoinstruções por esses motivos, por favor, acrescente uma sentença ou duas à sua solução, indicando quais pseudoinstruções usou e por quê.

Exercício 2.10

Nos problemas a seguir, a tabela de dados contém bits que representam o opcode de uma instrução. Você deverá traduzir as entradas para o código assembly e determinar que formato da instrução MIPS os bits representam.

a. 0000 0010 0001 0000 1000 0000 0010 0000_{dois}

b. 0000 0001 0100 1011 0100 1000 0010 0010_{dois}

Utilizar MIPS Reference Data Card.pdf					
R Op (6 bits)	Rs (5 bits)	Rt (5 bits)	Rd (5 bits)	Shamt (5 bits)	Funct (6 bits)
I Op	Rs	Rt	Endereço (16 bits)		
J Op	Endereço (26 bits)				
a. 0000 0010 0001 0000 1000 0000 0010 0000 _{dois}					
b. 0000 0001 0100 1011 0100 1000 0010 0010 _{dois}					

2.10.1 [5] <2.5> Para essas entradas binárias, que instrução elas representam?

A a entrada binaria so item a representa uma adiç o (add). Por outro lado, a entrada binaria do item b representa uma subtraç o (sub).

2.10.2 [5] <2.5> Que tipo de instrução (tipo I, tipo R) as mesmas entradas binárias representam?

Elas representam o tipo de instrução R.

2.10.3 [5] <2.4, 2.5> Se as entradas binárias anteriores fossem bits de dados, que número elas representariam em hexadecimal?

A instrução do item a em hexadecimal ficaria:

01084010 e 014B4822, respectivamente.

Nos problemas a seguir, a tabela de dados contém instruções MIPS. Você deverá traduzir as entradas para os bits do opcode e determinar qual é o formato da instrução MIPS.

```
a. addi $t0 , $t0 , 0
```

```
b. sw $t1, 32($t2)
```

R Op (6 bits)	Rs (5 bits)	Rt (5 bits)	Rd (5 bits)	Shamt (5 bits)	Funct (6 bits)
I Op	Rs	Rt	Endereço (16 bits)		
J Op	Endereço (26 bits)				
a. addi \$t0, \$t0, 0					
b. sw \$t1, 32(\$t2)					

2.10.4 [5] <2.4, 2.5> Mostre a representação hexadecimal dessas instruções.

```
001000 01000 01000 000000000000000000
82080000
```

```
101011 01010 01001 0000000000100000
AD490020
```

2.10.5 [5] <2.5> Que tipo (tipo I, tipo R) essas instruções representam?

Essas instruções são do tipo I pois utilizam valor imediato.

2.10.6 [5] <2.5> Qual é a representação binária e hexadecimal dos

campos opcode, Rs e Rt nessa instrução? Para as instruções de tipo R, qual é a representação hexadecimal dos campos Rd e funct? Para as instruções de tipo I, qual é a representação hexadecimal do campo imediato?

Na instrução do item a do tipo I temos:

Opcode: 001000(bin) ou 8(hex)

Rs: 01000(bin) ou 8(hex)

Rt: 01000(bin) ou 8(hex)

Imediato: 0000000000000000(bin) ou 0(hex)

Na instrução do item b do tipo I temos:

Opcode: 101011(bin) ou 2B(hex)

Rs: 01010(bin) ou A(hex)

Rt: 01001(bin) ou 9(hex)

Imediato: 0000000000100000(bin) ou 20(hex)

Na instrução do item a do tipo R temos:

Rd: 10000(bin) ou 10(hex)

funct: 100000(bin) ou 20(hex)

Na instrução do item b do tipo R temos:

Rd: 01001(bin) ou 9(hex)

funct: 100010(bin) ou 22(hex)

Exercício 2.16

Para estes problemas, a tabela mantém diversos valores binários para o registrador \$t0.

Dado o valor de \$t0, você deverá avaliar o resultado de diferentes desvios.

a. \$t0 = 0010 0100 1001 0010 0100 1001 0010 0100_{dois}

b. \$t0 = 0101 1111 1011 1110 0100 0000 0000 0000_{dois}

2.16.1 [5] <2.7> Suponha que o registrador \$t0 contenha um desses valor e \$t1 tenha o valor

\$t1 = 0011 1111 1111 1000 0000 0000 0000 0000_{dois}

Note o resultado da execução de tais instruções em certos registradores. Qual é o valor de \$t2 depois das seguintes instruções?

slt \$t2, \$t0, \$t1

beq \$t2, \$ZERO, ELSE

j DONE

ELSE: addi \$t2, \$0, 2

DONE:

Para o valor do item a, temos que o valor de \$t2 é 1.

Para o valor do item b, temos que o valor de \$t2 é 2.

2.16.4 [5] <2.7> Suponha que o registrador \$t0 contenha um valor da

tabela anterior.

Qual é o valor de \$t2 após as instruções a seguir?

```
        slt $t2, $0, $t0
        bne $t2, $ZERO, ELSE
        j  DONE
ELSE:    addi $t2, $t2, 2
DONE:
```

Para o valor tanto do item a quanto do item b, temos que o valor de \$t2 é 3.