



$d_0 \ d_1 / d_2 \ d_3 \ d_4 \ d_5 \ d_6 \ d_7 \ d_8$

Nome:

GABALITO

Matrícula: /

Prova 1

(4.0) 1) Você, trabalha na XingLing Inc. empresa especializada em “fabricar” aparelhos de DVD de baixo custo. Como todo bom engenheiro desta empresa, você está fazendo um trabalho de engenharia reversa no último produto lançado no mercado pelo seu concorrente, um GreenRay holográfico.

Dado o *dump* de memória ao lado, onde o segmento de código inicia-se no endereço 0x00400000 e o segmento de dados a partir do endereço 0x10010000.

(1.0) a) Desassemble o código de máquina em Assembly MIPS.

(1.0) b) Qual o conteúdo, em hexadecimal, do endereço 0x10010008 após a execução do código?

(2.0) c) Escreva um código que realize a mesma tarefa, porém de forma mais eficiente (menor tempo de processamento e processador com menor área de chip).
Dica: Use aritmética em ponto fixo.

```
0x00400000 : 3c 01 10 01
0x00400004 : 34 28 00 00
0x00400008 : c5 00 00 00
0x0040000c : c5 01 00 04
0x00400010 : 46 01 03 02
0x00400014 : 20 02 00 02
0x00400018 : 00 00 00 0c
0x0040001c : 11 00 ff f8
0x00400020 : e5 0c 00 08
0x00400024 : 03 e0 00 08
...
0x1001000 : 40 00 00 00
0x1001004 : 40 10 00 00
0x1001008 : FF FF FF FF
...
```

(3.0) 2) Dado o procedimento PROC em Assembly MIPS ao lado, sendo executado em um processador onde as instruções tipo-J são executadas em 1 ciclo, instruções tipo-R são executadas em 2 ciclos, instruções tipo-I em 3 ciclos, e instruções de acesso à memória de dados são executadas em 4 ciclos de clock.

(1.0) a) Qual deve ser a velocidade do processador para que o procedimento seja executado em menos de $1\mu s$? Qual deve ser o período máximo de clock?

(1.0) b) Qual a CPI média deste procedimento?

(1.0) c) Se vc como projetista quisesse melhorar o desempenho deste trecho, modificando de 1 ciclo o número de ciclos de apenas um dos tipos de instruções. Qual seria esse tipo de instrução? Quanto tempo seu novo processador demandaria, caso mantivesse a frequência mínima de clock do item a)?

```
PROC: add $t0,$zero,$zero
      li $t1,5
LOOP: beq $t0,$t1,FIM
      sw $t0,0($t0)
      add,$t0,$t0,1
      j LOOP
FIM:   jr $ra
```

(3.0) 3) Os montadores podem implementar várias pseudo-instruções em máquinas RISC que são muito úteis aos programadores e que são típicas de máquinas CISC. Implemente as pseudo-instruções abaixo com instruções originais da ISA MIPS.

(1.0) a) seqi \$t0,\$t1,0x1234 # Set if EQual Immediate: Se \$t1=0x1234 então \$t0=1 senão \$t0=0

(1.0)b) addv END3,END1,END2,N # Adição vetorial: memoria[END3+i]=memoria[END1+i]+memoria[END2+i]
para i=0 a N-1 ENDS=32bits N=16bits

(1.0) c) multadd.d \$f0,\$f2,\$f4 # Multiply Add : \$f0 = \$f0 + \$f2 × \$f4

Dica: Use END1[31:16] e END1[15:0] para representar as partes mais significativa e menos significativa de uma word END1.

(1.0) 4) Qual a diferença entre workload e benchmark? Qual é o workload ideal?

Gabarito

10

a) $\alpha x0040$: 0000 0000 0000 0000 0000 0000 0000 0000 ; \text{LU stat, Dxbal}

0004: 0011110000000000 0001000000000000 ; LUI \$t2, \$at, 0

0008: 0010001000000000 0000000000000000 ; ORI \$t2, \$t2, 0

000C: 1100010100000000 0000000000000000 ; LWCI \$f0, 0(\$t2)

0010: 1100010100000000 0000000000000000 ; LWCI \$f1, 4(\$t2)

0014: 0010000000000010 0000000000000000 ; MUL.S \$f2, \$f0, \$f1

0018: 0000000000000000 0000000000000000 ; addi \$t2, \$t2, 2

001C: 0000000000000000 FFFFFFFF 0000000000000000 ; syscall

0020: 1110001000001100 0000000000000000 ; log \$t2, \$t2, -8

0024: 0000000000000000 0000000000000000 ; SWCI \$f12, 8(\$t2)

0028: 1100000000000000 0000000000000000 ; JR \$t2

$$b) \$f_0 = 0 \times 1001000 + 8 = 0 \times 10010008 \rightarrow \$f_{12}$$

$$\$f_{12} = \$f_0 \times \$f_1$$

$$\$78 = 8 \times 4000\ 000 = 0100\ 000\ 000\ 000$$

$$f(0) = (-1)^0 \cdot (1+0) \cdot 2^{120-127} = 2^{100}$$

$$\$f7 = 0 \times 4010\ 0000 = 0100\ 0000\ 0000\ 0000\ 0000\ 0000$$

$$0,25 \quad \$61 = (-1)^0 \cdot (1 + 0,001) \times \{^{128-127} = 1,125 \times 2 = 2,25$$

$$\text{Preis} \times \text{Menge} = 2 \times 2,25 = 4,5$$

$$L_3 = 0100,10 = 1,001 \times 2^2$$

$$E=2 \text{ EXP}=129$$

~~\$flz = 0100000100100000~~

Q5

$$\text{MEM}(0x10010008) = 0x40900000 //$$

Q5

c) multiplica $2,0 \times 2,25 \rightarrow 450$ Q2
ocorre na memória

1,0

La \$t0, 0x100

0x10010000; 00000008

Lw \$t1, 0(\$t0)

0x10010004; 00000009

Lw \$t2, 4(\$t0)

0x10010008; 00000048

MUL \$t1, \$t2

2

MFL0 \$t3, raixt

Q4

Li \$t0, 1 \rightarrow rjt

00 0100, 1000

move \$t0, \$t3

4,5 //

SXScall

SW \$t3, 8(\$t0)

↳ ou reserve uma
rotina SXScall //

Jr \$ra

mostar \$t0 com \$a1=62

na Tela //

2)

Q5

a) A relocação do processador pode

ser qualquer, desde que WLC ($rel, W8$)

é muito menor //

$$\begin{aligned}
 &\text{add addi beq sw addi j beg jr} \\
 n\text{ ciclos} &= 2 + 3 + (3+4+3+1) \times 5 + 3 + 2 \\
 &= 65
 \end{aligned}$$

$$\text{Logo: } t = \text{nº ciclos} \times T \leq 1\mu$$

$$65 \times T \leq 1\mu$$

$$T \geq 15,3846 \text{ ns} //$$

05

$$f = \frac{1}{T} \geq 65 \text{ MHz}$$

25

$$\text{b) CPI} = \frac{65}{1+1+(1+1+1) \times 5 + 1+1} = 2,728 // 1,0$$

nº instruções

$$\text{c) Núclos} = R + I + (I + M + I + J) \times 5 + I + R \\ = 2R + 12I + 5M + 5J$$

Logo

um decomscimo de 1 ciclo na instrução I 0,5

causa a maior redução.

$$r = \text{ciclos} = 2 + 2 + (2 + 4 + 2 + 1) \times 5 + 2 + 2 \\ = 53 //$$

$$t = \text{nº núcleos} \times T = 53 \times \frac{1}{65n} = 815,38 \text{ ns} // 0,5$$

3)

a) seqi \$t0, \$t1, IIM

addi \$t0, \$zero, 1

addi \$at, \$zero, 0x1234

1,0

beq \$at, \$t1, FIM

addi \$t0, \$zero, 0

FIM:

b) addv END3, END1, END2, N

addi \$SP, \$SP, -28

sw \$T0, 0(\$SP)

sw \$T1, 4(\$SP)

sw \$T2, 8(\$SP)

sw \$T3, 12(\$SP)

sw \$S1, 16(\$SP)

sw \$S2, 20(\$SP)

sw \$S3, 24(\$SP)

addi \$T0, \$ZERO, N

lui \$T1, END1[31:16]

ori \$T1, END1[15:0]

lui \$T2, END2[31:16]

ori \$T2, END2[15:0]

lui \$T3, END3[31:16]

ori \$T3, END3[15:0]

add \$AT, \$ZERO, \$ZERO

LOOP: beq \$AT, \$T0, Fin

lw \$S1, 0(\$AT)

lw \$S2, 0(\$T2)

add \$S3, \$S2, \$S1

sw \$S3, 0(\$T3)

addi \$AT, \$AT, 1

loop

Fin:

lw \$T0, 0(\$SP)

lw \$T1, 4(\$SP)

lw \$T2, 8(\$SP)

lw \$T3, 12(\$SP)

lw \$S1, 16(\$SP)

lw \$S2, 20(\$SP)

lw \$S3, 24(\$SP)

addi \$SP, \$SP, 28

c) MULADD.D \$T0,\$T2,\$T4

addi \$SP, \$SP, -8 # DOUBLE

SDC1 \$T12, 0(\$SP)

MUL.D \$T12, \$T2, \$T4

ADD.D \$T0, \$T0, \$T12

LDC1 \$T12, 0(\$SP)

addi \$SP, \$SP, 8

4) Benchmark São

Categorias de Software

que tem prece. o

WORKLOAD DE USUÁRIO.

WORKLOAD IDEAL O,5

O conjunto de programas

que São periodicamente

usados por um usuário.

→ mais EFICIENTE.

mfc1 \$at, \$T12

MUL.D \$T12, \$T2, \$T1

ADD.D \$T0, \$T0, \$T12

mtc1 \$at, \$T12

exit 0

→ addi \$T1, \$T1, 4

addi \$T2, \$T2, 4

addi \$T3, \$T3, 4