



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

Nome: _____ Matrícula: /

Prova 1

1) (5.0) Dada a função: $f(x) = \frac{a \cdot x^2 + b \cdot \sqrt{x}}{c \cdot x}$,

a)(3.0) Respeitando a convenção do uso dos registradores, escreva um procedimento (otimizado) em Assembly MIPS que receba os valores em ponto flutuante precisão simples dos parâmetros a , b , c em $\$f1$, $\$f2$, $\$f3$ respectivamente, e de x em $\$f0$, e retorne em $\$f12$ (Parte Real) e $\$f13$ (Parte Imaginária) o valor de $f(x)$ para qualquer $x \in \mathbb{R}$.

b)(1.0) Considerando que seu procedimento irá ser executado em um processador MIPS uniciclo de 533MHz, onde cada instrução que envolva o Coprocessador 1 (FPU) necessita 5 ciclos de *clock*. Qual o tempo de execução para $a=3.333$; $b=4.5$; $c=2.0$ e $x=1.1$?

c)(1.0) Qual o valor esperado dos registradores $\$f12$ e $\$f13$, em hexadecimal, se:

c.1)(0.5) $a=3.543$; $b=4.423$; $c=10.0$; $x=0.226208993683$

c.2)(0.5) $a=d_4.d_5$; $b=-8.732$; $c=-d_2.d_3$; $x=0.0$;

2) (3.5) Dado o código em Assembly MIPS ao lado.
Qual é o valor impresso na tela? Justifique.

```
INICIO: li $a0,2
        beq $a0,$zero,JUMP
        la $t1,0x10800004
        la $t2,JUMP
        sw $t1,0($t2)
JUMP:   bne $a0,$zero,FIM
        la $t1,0x208400d5d6
        sw $t1,16($t2)
        li $a0,0
FIM:    li $a1,3
        li $v0,1
        syscall
```

3) (1.5) A vida do programador em Assembly MIPS é bastante facilitada pelo programa montador, uma vez que o mesmo implementa várias pseudo-instruções que são bastante úteis.

Dado que BIG é uma constante imediata de 32 bits, SMALL uma constante de 16 bits, LABEL um endereço de 32 bits, implemente as seguintes pseudo-instruções:

a)(0.5) `subi $t0, $t1, BIG`

$\$t0 = \$t1 - \text{BIG}$

b)(0.25) `b LABEL`

`branch LABEL`

c)(0.5) `s.s $f0, LABEL($t0)`

`Memory[LABEL+$t0] = $f0`

d)(0.25) `negi $t0, SMALL`

$\$t0 = -\text{SMALL}$;

BOA SORTE!

1ª Prova
GABARITO

1) $f(x) = \frac{ax + b\sqrt{x}}{c}$ Economiza 1 multiplicação

a) Listagen von Anexo

b) O procedimento necessita 12 instruções do Coproc 1 e 1 instrução MIPS (para)

Logo: $n^{\circ} \text{Circos} = 12 \times 5 + 1 = 61 \text{ circos}$

$$\text{assim: } \tau = 61 \times \frac{1}{533 \text{ M}} = 1,1444 \times 10^{-7} \text{ s} = 114,44 \text{ ns}$$

c) c.1) $\$12 = 1,070101 \Rightarrow 0x3F814AFD$

$$9 \div 13 = 0,00 \Rightarrow 0 \times 0000 \ 0000$$

C.2) \$12 = NaN \Rightarrow 0x7FC00000 \rightarrow\$ Representação de NaN
 Expo = 255 Mant ≠ 0

$$f_{13} = 0,00 \Rightarrow 0 \times 000000000$$

2) O valor de \$q_0\$ será impresso na tela, assim:

$$Li q\theta, 2 \rightarrow a\theta = 2$$

beg \$a0, \$ZERO, JUMP → F9C50, NÖN RESVIZ

la \$t1, 0x10800004 beq \$a0, \$ZERO, FIM

la \$12, 7000

la \$t2, jump
sw \$t1, 0(\$t2) → Escreve a instrução no endereço jump

~~JUMP, [ONE, \$ONE, \$ZERO, FIM] BEQ \$ONE, \$ZERO, FIM~~ → FALSE NOT DESIGN

la \$x1, 0x208400 d5d6 → addi \$a0, \$a0, 0x d5d6 } wi 21: sinca

SW \$t1, 16(\$t2) → Escreve a informação → 16 bytes após JUMP

7. $[l, r, 0] \rightarrow \text{add } \text{gas}, \text{gas}, \text{dx}, \text{dx}$

Fin: li \$a0, 3

li \$v0, 1
syscall

} escape \$a0 = 0xdeadbeef + 2 // r9 rca

3)

a) subi \$t0, \$t1, BIG → lui \$at, BIG[31..16]
ori \$at, \$at, BIG[15..0]
sub \$t0, \$t1, \$at

b) b LABEL → beq \$zero, \$zero, LABEL or j LABEL

c) s.s \$t0, LABEL(\$t0) → lui \$at, BIG[31..16]
ori \$at, \$at, BIG[15..0]
add \$at, \$at, \$t0
swc1 \$t0, 0(\$at)

d) negi \$t0, small → addi \$t0, \$zero, -small or addi \$at, \$at, small
sub \$t0, \$zero, \$at

a)

FX: mul.s \$f12, \$f1, \$f0 # $a \cdot x$
div.s \$f12, \$f12, \$f3 # $a \cdot x / c$
abs.s \$f13, \$f0 # $|x|$
sqrt.s \$f13, \$f13 # $\sqrt{|x|}$
mul.s \$f13, \$f13, \$f2 # $\sqrt{x} \cdot b$
div.s \$f13, \$f13, \$f0 # $b \sqrt{x} / x$
div.s \$f13, \$f13, \$f3 # $b \sqrt{x} / x / c$
mtcr \$ZERO, \$f31
c.lx.s \$f0, \$f31 # $x < 0$?
bc1t COMPLEX
add.s \$f12, \$f12, \$f13 # $\text{Re}(z) \leftarrow \text{Re}(z) + \text{Im}(z)$
mov.s \$f13, \$f31 # $\text{Im}(z) = 0$

CONTEXT: jr \$ra

Wigolo

b) $N_{\text{CIRWS}} = 10 \times 5 + 1 = 51$

$$t = 51 \times \frac{1}{533 \times 10^6} = 9,568 \times 10^{-8} = 95,68 \text{ ns}$$

(1) $\$p_{12} = 1.010101$ $\$p_{13} = 0 = 0 \times 00000000$

$\Phi_{12} = 0X\ 3F\ 81\ 4A\ FD$

C.2) $\$f12 = \text{NaN}$ $\$f13 = 0 = 0x00000000$

~~0012~~ = 0x 7F C0 00 00

0111 1111 1100 0000 ... 0000
255 0

2)

li \$a0, 2 # \$a0 = 2

beg \$a0, \$zero, jump # não pula

la \$t1, 0x10800005

la \$t2, jump

sw \$t1, 0(\$t2) # coloca em jump o valor

0001 0000 1000 0000 0000 0000 0000 0100

JUMP: ~~beg \$a0, \$zero~~ beg \$a0 \$zero, PC+5

la \$t1, 0x20040037 → lui \$t1, 0x2004

ori \$t1, \$t1, 0x0037

sw \$t1, 12(\$t2) # coloca em jump+12 o valor

0010 0000 1000 0100 0000 0000 0011 0111

li ~~\$a0, 0~~

addi \$a0, \$a0, 55
c5d6=32

fim: li \$a1, 3

li \$v0, 1

→ Logo mostra 55+2=57

syscall

3)

a) subi \$t0, \$t1, Big → lui \$at, Big(31..16)

ori \$at, \$at, Big(15..0)

sub \$t0, \$t1, \$at

b) b LABEL → beg \$zero, \$zero, LABEL ou LABEL

c) s.s \$t0, LABEL(\$t0) → lui \$at, LABEL(31..16)

ori \$at, \$at, LABEL(15..0)

add \$at, \$at, \$t0

swc1 \$t0, 0(\$at)

d) neg \$t0, \$t1 → sub \$t0, \$zero, \$t1