

Nombre: _____

¡Anota tu nombre y número de sección en todas las hojas del examen AHORA! (penalidad de 5 puntos)

Tienes 2 horas para completar todos los problemas. Lee cuidadosamente todo el examen antes de empezar a trabajar. Muestra todo el trabajo conducente a tu contestación. Podrás recibir crédito parcial por contestaciones parciales siempre y cuando muestres tu trabajo por escrito. Usa tu tiempo inteligentemente. Exitó!

INEL 4206 Staff

Nombre: _____

1	30
2	40
3	20
4	10
Total	100

Nombre: _____

Problema 1. (30 puntos) High Level Language Control Structures in Assembly Language

(a) (10 points) Write a segment of **MIPS** assembly code equivalent to the following high level language code.

```
int a = 3;
int b = 6;
int res = 0;
main() {
    if (a > b) {
        res = 1;
    }
    else if (a < b) {
        res = -1;
    }
    else {
        res = 0;
    }
}
```

```
                .data                // Comments
a:              .word                3
b:              .word                6
res:            .word                0
                .globl               main
                .text
main:           la                   $t0, a
                lw                    $s1, 0($t0)
                lw                    $s2, 4($t0)
                lw                    $s3, 8($t0)
                blt                   $s1, $s2, then
                bgt                   $s1, $s2, elseif
else:           li                   $s3, 0
                b                     end
elseif:        li                   $s3, 1
                b                     end
then:           li                   $s3, -1
end:           la                   $t0, a
                sw                    $s1, 0($t0)
                sw                    $s2, 4($t0)
                sw                    $s3, 8($t0)
                jr                    $ra
```

Nombre: _____

(b) (10 points) Repeat problem 1(a) but this time generate **EASY I** assembly code.

```
// Assume a stored at address 1000
// Assume b stored at address 1002
// Assume res stored at address 1004
```

```

        ANDi      0
        ADDi      3
        STOREi    1000
        ANDi      0
        ADDi      6
        STOREi    1002
        ANDi      0
        STOREi    1004
        LOADi     1000
        COMP
        ADDi      1
        ADD       1002
        BrNi      then
        COMP
        ADDi      1
        BrN       elseif
else:    ANDi      0
        JUMP      end
elseif:  ANDi      0
        ADDi      1
        COMP
        ADDi      1
        JUMP      end
then:    ANDi      0
        ADDi      1
end:     STOREi    1004
```

Nombre: _____

(c) (10 points) Write a segment of MIPS assembly code equivalent to the following high level language code.

```
int a = 0;
int b = 6;
int res = 0;
main() {
    while(a < b) {
        res = res + a;
        a = a + 1;
    }
}
```

```

                .data                // Comments
a:              .word    0
b:              .word    6
res:            .word    0
                .globl    main
                .text
main:           la        $t0, a
                lw         $s1, 0($t0)
                lw         $s2, 4($t0)
                lw         $s3, 8($t0)
loop:           bge       $s1, $s2, end
                add        $s3, $s3, $s1
                b          loop
end:            la        $t0, a
                sw         $s1, 0($t0)
                sw         $s2, 4($t0)
                sw         $s3, 8($t0)
                jr         $ra
```

Nombre: _____

Problema 2. (40 points) Easy I Microprocessor Implementation

In this exercise, you will extend the instruction set of the Easy I processor discussed in class (refer to Appendix A for information about the Easy I) with a new instruction described in the following table:

Symbolic Name and Description	Assembler Example	Action
BrNri – Branch on Negative Relative Immediate	BrNri loop	$AC < 0: PC \leftarrow PC + 2 + X$

- a) (10 points) Show any changes to the Easy I datapaths necessary to implement the new instruction. Show all you changes on the diagram in the next page. Provide a concise textual description of your proposed modifications in the box below:

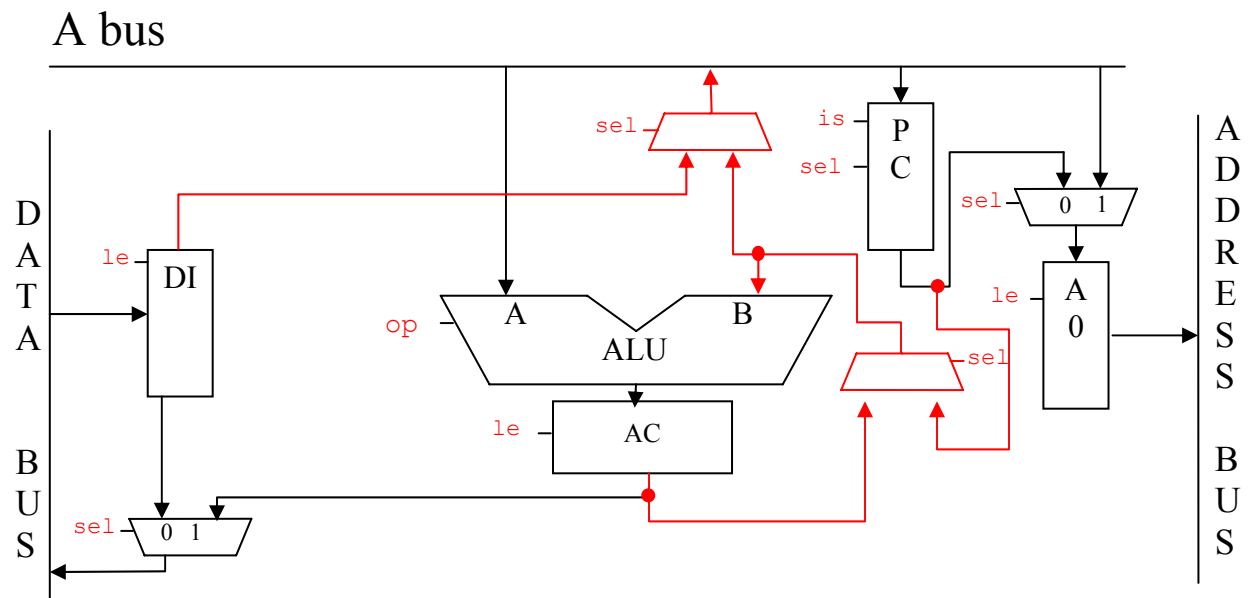
One possible approach is to modify the Datapaths in two ways:

1. Add path from PC to ALUB and corresponding MUX to select input B to ALU
2. Add path from AC to ABUS and corresponding MUX to select input ABUS

See next page.

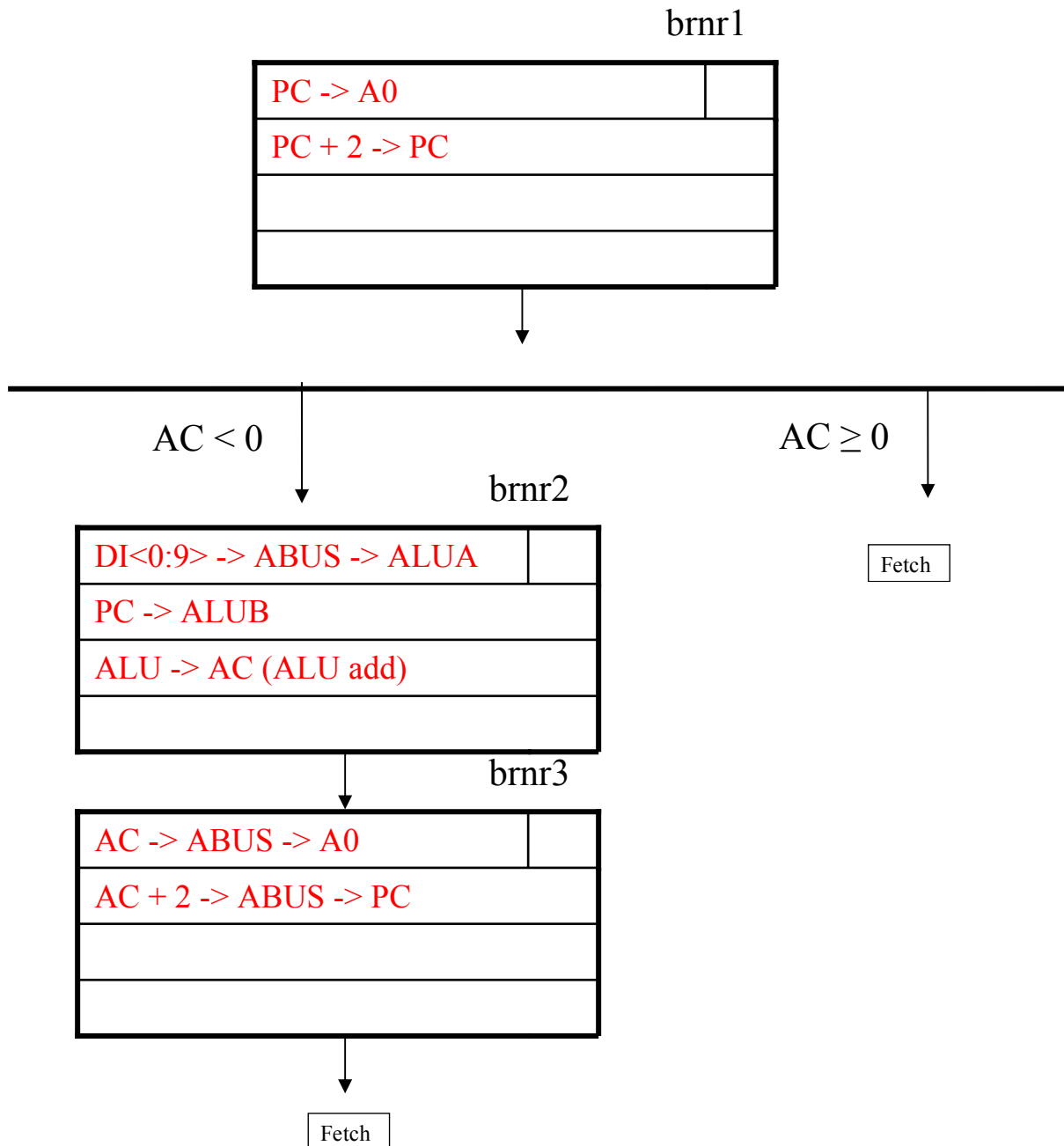
Nombre: _____

Easy I Datapaths



Nombre: _____

- b) (15 points) Show a level 2 flowchart implementing the BrNr instruction. We provide enough room for your flowchart below, but you may not need all the states depicted below.



Nombre: _____

- c) (**15 points**) Show any modifications to the Easy I state transition table described in class. Show your changes on the table in the following page. Also, provide a brief textual description of your changes in the box below.

Two modifications:

1. Add two columns for two new control points. The selector for the input to the ABUS and the selector for the input to ALUB
2. Add one row for the new brnr states

Changes on next page

Nombre: _____

Easy I State Transition Table

Curr State	opcode	I	AC:15	Next State	ALU op	Mem OP	PC sel	PC is	DI le	AC le	AO sel	AO le	EDB sel	ABUS sel	ALUB sel
reset1	xx xxx	x	x	reset2	XXX	NOP	01	X	0	0	X	0	X	0	X
reset2	xx xxx	x	x	fetch	XXX	NOP	10	1	0	0	0	1	X	0	X
fetch	00 00x	0	x	sopr	XXX	NOP	11	X	1	0	X	0	X	0	X
fetch	00 010	0	x	brn1	XXX	RD	11	X	1	0	X	0	X	0	X
fetch	00 011	0	x	jump	XXX	RD	11	X	1	0	X	0	X	0	X
fetch	00 100	0	x	store1	XXX	RD	11	X	1	0	X	0	X	0	X
fetch	00 101	0	x	load1	XXX	RD	11	X	1	0	X	0	X	0	X
fetch	00 11x	0	x	aopr	XXX	RD	11	X	1	0	X	0	X	0	X
aopr	00 110	x	x	fetch	AND	NOP	10	1	0	1	0	1	X	0	X
aopr	00 111	x	x	fetch	ADD	NOP	10	1	0	1	0	1	X	0	X
sopr	00 000	x	x	fetch	NOTB	NOP	10	1	0	1	0	1	X	0	X
sopr	00 001	x	x	fetch	SHRB	NOP	10	1	0	1	0	1	X	0	X
store1	xx xxx	x	x	store2	XXX	NOP	11	X	0	0	1	1	X	0	X
store2	xx xxx	x	x	store3	XXX	WR	10	1	0	0	0	1	1	0	X
load1	xx xxx	x	x	load2	XXX	NOP	11	X	0	0	1	1	X	0	X
load2	xx xxx	x	x	load3	XXX	RD	11	X	1	0	X	0	X	0	X
load3	xx xxx	x	x	fetch	XXX	NOP	10	1	0	1	0	1	X	0	X
brn1	xx xxx	x	0	fetch	XXX	NOP	10	1	0	0	0	1	X	0	X
brn1	xx xxx	x	1	brn2	XXX	NOP	10	1	0	0	0	1	X	0	X
brn2	xx xxx	x	x	fetch	XXX	NOP	10	0	0	0	1	1	X	0	X
jump	xx xxx	x	x	fetch	XXX	NOP	10	0	0	0	1	1	X	0	X
fetch	01000 ¹	x	x	brnr1	XXX	RD	11	X	1	0	X	0	X	0	X
brnr1	xx xxx	x	0	fetch	XXX	NOP	10	1	0	0	0	1	X	0	X
brnr1	xx xxx	x	1	brnr2	XXX	NOP	10	1	0	0	0	1	X	0	X
brnr2	xx xxx	x	x	brnr3	ADD	NOP	11	X	0	1	X	0	X	0	1
brnr3	xx xxx	x	x	fetch	XXX	NOP	10	0	0	0	1	1	X	1	X

¹ Assumes opcode for new instruction is 01000

Nombre: _____

Problem 3 (20 points) MIPS Assembly Code

- (a) **(10 points)** Add comments to the following MIPS code and describe in one sentence what it computes. Assume that \$a0 is used for the input and initially contains n, a positive integer. Assume that \$v0 is used for the output.

```
begin:      addi $t0, $zero, 0      # t0 = 0
            addi $t1, $zero, 0      # t1 = 0
loop:       slt  $t2, $a0, $t1      # while (t1 <= N){
            bne  $t2, $zero, finish #
            add  $t0, $t0, $t1      #     t0 += t1
            addi $t1, $t1, 3        #     t1 += 3
            j    loop              # }
finish:     add  $v0, $t0, $zero    # output = $t0
```

Computes the sum all all the multiples of 3 which are less or equal to N

- (b) **(10 points)** Provide the best equivalent sequence of MIPS instructions that could be used to implement the pseudo-instruction “set on greater than or equal”:

sgt \$rx, \$ry, \$rz

You may use register \$at for temporary results.

```
nequal:    bne  $rz, $ry, nequal
end:       ori  $rx, $r0, 1
           beq  $r0, $r0, end
           slt  $rx, $rz, $ry
```

Nombre: _____

Problema 4 (10 points) Evaluación del curso

(a) Menciona los tres aspectos que mas te gustan de la clase INEL 4206 en orden decreciente de importancia.

a.

b.

c.

(b) Menciona los tres aspectos que menos te gustan de la clase INEL 4206 en orden decreciente de importancia.

a.

b.

c.

(c) En una escala de 0 (no mejoría) al 5 (mejoría excelente) como evaluarías la respuesta del professor a los comentarios que sugeriste durante el examen parcial 1. Explica brevemente.

1

2

3

4

5

Nombre: _____

Appendices