Name _____ Student ID _____

CSc 252: Computer Organization

Spring 2025

Test 6: Apr. 24, 2025

Time: 30 min

DO NOT OPEN THE EXAM UNTIL INSTRUCTED TO DO SO

Please read this page and follow the directions before proceeding with the rest of the exam.

- To give all students the same amount of time to do the exam, please DO **NOT** OPEN THIS EXAM UNTIL INSTRUCTED TO DO SO.
- You are not allowed to use any external resources such as cellphones, notes, headphones, watch, neighbors, calculator, etc. If you have not done so yet, turn your cellphone off and place it in your backpack.
- The cellphone cannot be on you during the exam. If your cellphone is in your pocket, it will be considered cheating even if you are not looking at it (same for headphones and watches). We will collect your exam and ask you to leave.
- Place your final answers in the given boxes. You can show your work on any blank spaces.
- We recommend skimming the entire exam before completing any problems.
- Read carefully every question before answering and raise your hand if the question is unclear.
- DO NOT SPEAK TO ANYONE AT YOUR TABLE.

****** Good Luck! ******

Allowable MIPS Instructions

When writing MIPS assembly, the only instructions that you are allowed to use (so far) are:

- `and, andi, or, ori, nor, nori, xor, xori`
- `add, addi, sub, addu, addiu`
- `beq, bne, j, jal, jr`
- `slt, slti, sll, srl, sra`
- `lw, lh, lb, sw, sh, sb`
- `lui, la`
- `syscall`

Write your name and student ID on all the exam pages for one extra credit point.

Name _____ Student ID _____

1. (6 pts) Convert 5.75 to binary (e.g. -10.111 is -2. 875 in binary)

5 ->101

0.75 x 2 = 1.5

0.5 x 2 = 1

Place your final answer here. 101.11

2. (6 pts) Write the number -1011.101010 using normalized scientific notation = $(-1)^S \times (1.\mathrm{M})_2 \times 2^e$.

Sign = 1 (negative)

1011.101010 = 1.011101010 x 2^3

Place your final answer in the boxes.

$$(-1)^{\boxed{1}} \times \boxed{1.011101010}_2 \times 2^{\boxed{3}}$$

Name _____ Student ID _____

3. (6 pts) Write the number $(-1)^0 \times (100.010101)_2 \times 2^{56}$ using IEEE 754 single precision encoding (1 bit for sign, 8 bits for biased exponent, and 23 bits for mantissa; bias = 127; e = E – bias)

Sign = 0
100.010101 = 1.00010101 x 2^2
2^2 x 2^56 = 2^58

e = 58
E = 58 +127 = 185

185 to binary:
128 = 2^7 = 1 => 185 – 128 = 57
64 = 2^6 = 0
32 = 2^5 = 1 => 57 – 32 = 25
16 = 2^4 = 1 => 25 – 16 = 9
9 in binary = 1001

Place your final answer here.

0	1	0	1	1	1	0	0	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

4. (6 pts) Step 3. Write the number $(-1)^1 \times (11.011)_2 \times 2^{-130}$ Using IEEE 754 single precision encoding (32-bit floating point, 1 bit for sign, 8 bits for biased exponent, and 23 bits for mantissa).

Sign = 1
11.011 x 2^-130 => does not fit in the normalized format, so we need to de-normalize it.
0.0011011 x 2^-126

Place your final answer here.

1	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Name _____ Student ID _____

5. (10 pts): Fill out the following table to draw the pipeline diagram for the below MIPS program on a pipelined machine that forwards when a data hazard occurs and flushes two instructions when a branch is taken. Do not forget to draw the arrows. You only need to write the line number in the "# Inst" column.

```
1     addi $t1, $zero, 4
2     addi $t3, $zero, 0
3     lw   $t4, 0($t1)
4     add  $t2, $t2, $t4
5     slt  $t7, $t1, $t3
6     beq  $t7, $zero, END
7     addi $t0, $sp, 16
8     addi $t1, $zero, 4
9     addi $t0, $t0, 4
10    addi $t3, $t3, 1
11    addi $t5, $a0, 0
    END:
12    addi $v0, $zero, 24
```

	# Inst	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	1	F	D	E	M	W																
2	2		F	D	E	M	W															
3	3			F	D	E	M	W														
4	4				F	D	D	E	M	W												
5	5					F	F	D	E	M	W											
6	6							F	D	E	M	W										
7	~~7~~								~~F~~	~~D~~												
8	~~8~~									~~F~~												
9	12										F	D	E	M	W							
10																						
11																						
12																						
13																						
14																						
15																						

Name _____ Student ID _____

6. (6 pts): For a given program, 45% of the branches are taken, and 66% of the values loaded are immediately used. The CPI = 1 + lp + mp + fp. Assuming a mp penalty of 2.

Instruction type	**Percentage of instruction mix**
Alu operations (add, sub, . . .)	58 %
Load from memory operations	10 %
Store from memory operations	10 %
Conditional branches	13 %
jal instructions	4 %
jr instructions	5 %

a. What is the load and use penalty (lp) for this program?

O	1 * 58% * 45%	O	2 * 58% * 45%
O	1 * 10% * 45%	O	2 * 10% * 45%
O	1 * 13% * 45%	O	2 * 13% * 45%
O	1 * 4% * 45%	O	2 * 4% * 45%
O	1 * 9% * 45%	O	2* 9% * 45%
O	1 * 5% * 45%	O	2 * 5% * 45%
O	1 * 58% * 66%	O	2 * 58% * 66%
●	1 * 10% * 66%	O	2 * 10% * 66%
O	1 * 13% * 66%	O	2 * 13% * 66%
O	1 * 4% * 66%	O	2 * 4% * 66%
O	1 * 9% * 66%	O	2 * 9% * 66%
O	1 * 5% * 66%	O	Other

b. What is the mispredicted branch penalty (mp) for this program?

O	1 * 58% * 45%	O	2 * 58% * 45%
O	1 * 10% * 45%	O	2 * 10% * 45%
O	1 * 13% * 45%	●	2 * 13% * 45%
O	1 * 4% * 45%	O	2 * 4% * 45%
O	1 * 9% * 45%	O	2* 9% * 45%
O	1 * 5% * 45%	O	2 * 5% * 45%
O	1 * 58% * 66%	O	2 * 58% * 66%
O	1 * 10% * 66%	O	2 * 10% * 66%
O	1 * 13% * 66%	O	2 * 13% * 66%
O	1 * 4% * 66%	O	2 * 4% * 66%
O	1 * 9% * 66%	O	2 * 9% * 66%
O	1 * 5% * 66%	O	Other

c. What is the function call penalty (fp) for this program?

Place your final answer here. Use the same format as the previous 2 questions.

1*9%

Name _____ Student ID _____

7. (10 pts): Assume the following C struct to answer the following MIPS questions:

```
struct Turtle {
    char  x;        // X coordinate.
    char  y;        // Y coordinate.
    char  dir;      // direction.
    char  age;      // Turtle's age
};
```

The fields of the struct are arranged as follows:

```
           +--------+
 byte  0   |   x    |   <----- char x
           +--------+
 byte  1   |   y    |   <----- char y
           +--------+
 byte  2   |  dir   |   <----- char dir
           +--------+
 byte  3   |  age   |   <----- char age
           +--------+
```

Let's say you are implementing the function `turtle_ages(Turtle *obj)` in MIPS, which has one parameter: the address of an object of type Turtle.

7.1. Write the MIPS code to store the Turtle's X coordinate into register s0

```
lb    $s0, 0($a0)
```

7.2. Write the MIPS code to store the Turtle's Y coordinate into register s1

```
lb    $s1, 1($a0)
```

7.3. Write the MIPS code to increment the Turtle's age by one. This change must be reflected in memory, not only in the registers.

```
lb    $s2, 3($a0)
addi  $s2, $s2, 1
sb    $s2, 3($a0)
```