Name _____ Student ID _____

CSc 252: Computer Organization

Spring 2025

Test 2: Feb. 20, 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:

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

Name _____ Student ID _____

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

1. (a) (6 points) Write the **sum-of-products** expression for each of the outputs X, Y, Z in the truth table below.

A	B	C	X	Y	Z
0	0	0	0	1	1
0	0	1	1	0	0
0	1	0	1	0	1
0	1	1	0	1	0
1	0	0	1	0	1
1	0	1	0	1	0
1	1	0	0	1	1
1	1	1	1	0	0

Place your final answers here.

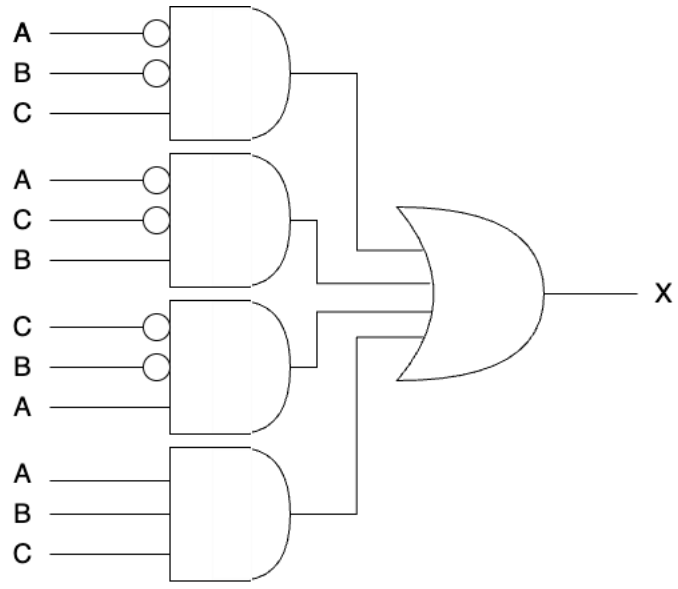
X = A'B'C + A'BC' + AB'C' + ABC

Y = A'B'C'+ A'BC + AB'C + ABC'

Z = A'B'C'+ A'BC' + AB'C' + ABC'

(b) (4 points) Now, draw a circuit diagram that implements **only output X**.

Place your final answers here.



Name _____ Student ID _____

2. (3 points): Consider a signed integer stored in `$s0`. Write the MIPS code to determine if `$s0` is odd **without branching**. Your code should assign the value of `1` to register `$s1` if `$s0` is odd and the value of `0` otherwise.

Place your final answer here.

```
andi $s1, $s0, 0x1
```

3. (2 points): Consider the Java classes given in Sim2, provide the output for this code snippet.

```
XOR mygate = new XOR();
mygate.a.set(false);
mygate.b.set(true);
mygate.execute();
System.out.println(mygate.out.get());
```

Place your final answer here.

```
true
```

4. (2 points) What is the value stored in `$t1` after executing the following MIPS code? Write your answer in hexadecimal.

```
addi  $t1, $zero, 15
sll   $t1, $t1, 28
addi  $t1, $t1, 15
```

Place your final answer here. One nibble on each box

$t1 = 0x | F | 0 | 0 | 0 | 0 | 0 | 0 | F |

Name _____ Student ID _____

5. (5 points): Assume that you have a string `str`, and its address is stored in `$s0`. You also have an index into the string `i`, which is currently stored in `$s1`. Give the MIPS assembly to copy the value at `str[i]` into `$s2`. Do not modify any other register.

Place your final answers here.

```
add $s0, $s0, $s1
lb $s2, 0($s0)
```

6. (8 points): What are the values stored in `$s0, $s1, $s2, and $s3` after executing the following MIPS code? Write your answer in hexadecimal.

```
.text
     addi $s0, $zero, 0
     addi $s1, $zero, 5
     addi $s2, $zero, -1
     addi $s3, $zero, 8
     slt  $t0, $s1, $s0
     bne  $t0, $zero, label1
     slt  $t1, $s2, $s3
     beq  $t1, $zero, label1
     sra  $s0, $s2, 2
     sub  $s2, $s3, $s1
     j end
label1:
     sll  $s0, $s2, 4
     add  $s2, $s3, $s1
end:
```

Place your final answer here. One nibble on each box

$s0 = 0x	F	F	F	F	F	F	F	F
$s1 = 0x	0	0	0	0	0	0	0	5
$s2 = 0x	0	0	0	0	0	0	0	3
$s3 = 0x	0	0	0	0	0	0	0	8

groups

Name _____ Student ID _____

7. (10 points) Give MIPS assembly for this common `for()` loop. Include all the necessary labels; you don't need to fill in the body with anything but indicate where the body would go.
   In this problem, comments are mandatory. Use comments to make it clear which registers are representing which C variables.

```
for (int i=6; i<100; i+=4)
... body ...
```

Place your final answers here.

```
.text   # it's okay id students do not write this

     # for (int i=6; i<100; i+=4)
     # ... body ...
     # I'll use $t0 to keep track of the value of i

     addi  $t0, $zero, 6  # t0 = i=6

     #check for condition i<100
     addi  $t1, $zero, 100  # store 100 in a register so we
can compare
for:
     slt   $t2, $t0, $t1    #  t2 = i < 100
     beq   $t2, $zero, end  # if i is not less than 100 end

     # ... body


     addi  $t0, $t0, 4      #i+=4
     j for   # check condition

end:
```

Name _____ Student ID _____

8. (5 points) Write the MIPS code to print out the message "CSC252 Rocks!"

Place your final answers here.

```
.data

MSG:    .asciiz "CSC252 Rocks!"

.text
    addi $v0, $zero, 4
    la $a0, MSG
    syscall
```

Name _____ Student ID _____

9. (5 points) Given the following MIPS code:

```
.data
something:  .word   0
            .word   0
            …               # more lines like the previous one
            .word   0

i:          .word   xx    # index (assume a positive number)
```

Give the MIPS assembly to copy the value at `something[i]` into `$s2`. In this problem, comments are mandatory. Use comments to make it clear which registers are representing which variables.

Place your final answers here.

```
.text   # it's okay id students do not write this

    la $t1, i                   #t1 = &i
    lw $t1, 0($t1)              #t1 = i
    la $t2, something           #t2 = something

    #compute the address of element i
    sll $t1, $t1, 2       # t1 = i * 4
    add $t1, $t1, $t2     # t1 = base + i * 4

    lw $s2, 0($t1)              #s2 = something[i]
```