1) 4 points (2 points each). Convert 8-bit (2-digit) 2’s complement hex values:

41

-29

2) 4 points. Convert 8-bit 2’s complement hex to decimal:

0x3A

0xE5

3) 4 points. Give the truth table for this transistor-level diagram

4) 4 points. Create a truth table for the following circuit.

5) 4 points. Prof. Madden likes Sesame bagels, but does not like Raisin bagels. He will be Happy if the coffee shop has Sesame bagels, and does not have Raisin bagels. Or, if it’s the Weekend, he’s always happy no matter what. Given inputs of S, R, W, sketch a small circuit using Boolean logic gates to determine if he is H.
6) 4 points. Simplify the Boolean equation \( Y = (ABC + A!BC)(B+C)(A) \)

7) 4 points. You need a 2-input OR gate, but you only have an AND gate and some inverters. Use your AND gate and inverters to make an OR gate.

8) 4 points. Show the truth table for the following finite state machine. \( S1 \ S0 \) is the current state, \( NS1 \ NS0 \) is the next state. The input is \( T \)

9) 6 points. Construct a Karnaugh map, and show the simplified circuit for the following truth table.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>x</td>
</tr>
</tbody>
</table>
For the following questions, assume that the data segment starts at 0x10010000, and any code starts at 0x40000000. The MIPS assembly language sometimes converts a single machine instruction into two simpler instructions; go ahead and assume that this does not happen, and each instruction is only 32 bits.

10) 4 points. What is in $ra, when you get to the beq?

    addi $a0, $0, 55
    addi $a1, $a0, 23
    jal cosine
    beq $a3, $a4, skip

11) 2 points. What is in a1, after this instruction?

    lw $a1, b

12) 2 points. What is in a1, after this instruction?

    la $a1, b

13) 10 points. Write a MIPS assembly subroutine “maxArray”. You are passed a pointer to an integer array (that has all positive numbers, with zero used to terminate the array). Your subroutine should find the maximum (largest positive) number in the array, and return that.
14) 4 points. Write MIPS assembly to add 8 to a value (passed as a pointer). The C code looks like this. This is a subroutine.

```c
add_eight(int *x)
{
    *x = *x + 8;
}
```

15) 8 points. Write MIPS assembly for sub1, a subroutine that calls two other subroutines. The C code looks like this. This is also a subroutine. sub2 and sub3 are in a library.

```c
int sub1()
{
    sub2(10);
    sub3(30);
    return 6;
}
```

16) 4 points. You have a pointer in register a0 to the C structure below. Write MIPS code for the line of C.

```c
struct
{
    int x;
    int y;
    int z;
} examObj;
```
17) 2 points. Name two types of hazards you might see on a pipelined microprocessor

18) 4 points. Modify the code below so that it could run faster on a pipelined microprocessor.

```
add $a0, $a3, $a4
add $a0, $a1, $a0
lui $a3, 58
```

19) 2 points. Suppose you have a pipelined machine. Each cycle is 300ps, and there are five stages in the pipeline. How long does it take to complete one instruction?

20) 2 points. Other than rearranging lines of code, what's a way that you can fix a hazard?

21) 4 points. You have a direct mapped cache with 4k entries. Your program makes 2000 memory access, with 1200 of them being hits. The processor is pipelined, has 6 stages, and each cycle is 50ps. Sesame bagels are delicious. What is the miss rate?
22) 8 points. You have a simple direct-mapped cache with 8 blocks, each block being one word (just like the example in the slides and textbook). You run the following code; what is the hit rate for the memory accesses?

```assembly
addi $t0, $0, 10
loop: beq $t0, $0, done
    lw $t1, 0x4($0)
    lw $t3, 0x8($0)
    addi $t0, $t0, -1
    j loop
done:
```

23) 4 points. We have this page table. What physical location does virtual memory address 0x5B44 map to?

24) 2 points. What does the abbreviation TLB stand for?

25) 4 points. Bonus question. You're writing a travel guide. What is the best thing about Binghamton?