COEN 311: Computer Organizations and Software

Assignment 4- 8086 Instructions
Deadline: Refer to Moodle
Submission instructions:

- You have to demonstrate your work for each question, writing final answers is not acceptable.
- Only one **pdf** file is acceptable.
- Submit through Moodle.
- Emails are not accepted. In case you face technical issue with Moodle, you must send a screenshot of the issue and email your assignment **before** the deadline.

Note: You must include a statement that you are submitting your original work.

1- What operation is performed by the instruction sequence that follows? **(10 points)**
Assume (DS) = 0100, (SI) = 0200 H

```
MOV AH, [SI]
SAHF
```

2- Assume the status of registers and memory are given as below:

(AX) = 8001 H, (SI)= 0200 H, (DI) = 0300 H, (DS:100H) = F0 H, (DS:200H) = F0 H, (DS:201H) = 01 H, (DS: 300) =34H, (DS:301) = 12 H

Describe the operation of the following instructions: **(10 points)**

Explain what is the result of each instruction.

a) CMP [0200H], AL
b) CMP WORD PTR [DI], 1234H

3- Assume the values are unsigned, what happens to ZF and CF status flags after the following instructions are executed. Assume that they are both initially cleared. **(10 points)**

```
MOV BX, 1111H
MOV AX, BBBB H
CMP BX, AX
```

4- Identify the type of the jump and the type of the operand in the following instructions: **(18 points)**

a) JMP 10H
b) JMP 1000H
c) JMP WORD PTR [SI]
d) JNC 20 H

e) JNP 1000H
f) JO DWORD PTR [BX]

5- Assume:
(CS) = 1075 H, (IP) = 0300 H, (SI) = 0100H , (DS:100H)= 00H, (DS:101H) = 10H

To what address is program control passed in each of the following instructions: (**12 points**)

a) JMP 10H
b) JMP 1000H
c) JMP WORD PTR [SI]

6- Encode the following instructions using the formats given in class notes. Note that there may be special cases that do not follow the general format: **(10 points)**

a) INC CX
b) MOV [SI], BX
c) MOV CL, 20
d) ADD AX, [DI] +1234H
e) MOV AX, [0100H]

7- Write a program that compares two arrays, A and B. Each array contains 100 8-bit signed numbers. Compare corresponding elements of the two arrays until either two elements are found to be unequal or all elements of the arrays have been compared and found to be equal. Update the data segment to the new DATASEGADDR value, and assume the arrays start at offsets A000 H and B000 H in this segment, respectively. If the two arrays are found to be unequal save the address of the first unequal element of A array in the memory location with the offset address FOUND in this data segment. If the two arrays are equal, write 0 in this memory location FOUND. **(30 points)**