Time Allowed: 60 minutes. **Maximum Points**: 30 points

==========

Q No. 1 Briefly answer each of the following: [5 x 2 + 1 x 4 = 14 points]

(i) Differentiate between SAR and SHL instructions with one example each.

(ii) Using shift and add instructions multiply a decimal number X_{10} by 23_{10}. Assume that the result does not exceed the range of a16-bit register.

(iii) Why an assembly procedure cannot be implemented using JMP instructions?

(iv) Why stack parameters are considered more convenient than register parameters. Justify your answer.

(v) Provide the contents of registers where indicated, after execution of the following instructions:
Note: SOLVE THIS PART HERE.

```
MOV AL, 41h
MOV BL, 56h
TEST AL, 56h                    ; AL = _________
CMP AL,  BL
JNE L1
ADD AL, 32h                     ; AL = _________
L1:
MOV DX,0
MOV AX, 8003H
MOV CX, 100H
DIV CX                          ; AX = _________ , DX = _________
```

(vi) Give the contents of the following registers, along with the run-time stack, when the following instructions are executed. Initially, consider ESP = 00001FF8h.

Note: SOLVE THIS PART HERE. No Marks will be awarded without proper working using the stack diagrams.

```
X1 DWORD 25H
X2 DWORD 27H
MAIN PROC
                PUSH 6H
                PUSH 5H
                CALL P1
    11500000H   MOV RESULT, EAX              ; ESP:      _____________
MAIN ENDP
    P1 PROC
    115000A4H   PUSH EBP
                MOV EBP, ESP                 ; EBP:      _____________
                MOV EAX, [EBP+8]
                ADD EAX, [EBP+12]            ; EAX:      _____________
                PUSH OFFSET X1
                PUSH OFFSET X2               ; ESP:      _____________
                POP ESI
                POP EBX
                ADD [ESI], EAX               ; X2:       _____________
                ADD [EBX], EAX               ; X1:       _____________
                MOV ESP, EBP
                POP EBP
                RET 8                        ; EIP:      _____________
    P1 ENDP
```

Q. No. 2 (i) Suppose the following data is received from a wireless sensor node operating in a smart building and is stored in EAX register, as shown in Figure 1. You are required to write an assembly language program with the corresponding data definition directives that would extract the data items and store them at memory locations Sequence_Number, Revision_Count, Status, and Sensor_Data.

a) Bits 0 to 11 reflect an integer Sequence_Number of the packet being sent.
b) Bits 12 – 14 show an integer Revision_Count of the packet.
c) Bit 15 is the Status of the sensor flag (0 – Forwarded Data and 1 – Sensed Data)
d) Bits 16 – 31 contain the Sensor_Data. [04 points]

16 bits	1 bit	3 bits	12 bits
Sensor_Data	Status	Revision_ Count	Sequence_Number

Figure: 1

(ii) Write an assembly language code to implement the following high-level language code showing the use of LEA instruction and OFFSET assembler directive. [04 points]

```
char moon [20];
void star_array () {
      char cell[20];
      for (int i=19; i>=0; i--) {
         cell[i] = '*';
         moon[i] = 'x';
         }
}
```

Q. No. 3 (i) Write a recursive procedure in x86 assembly language that divides a number by another number and stops when dividend is less than or equal to 5h. Consider dividend = D4A4h and divisor = Ah. [04 points]

(ii) Implement the following pseudo-code in assembly language. Also, give the corresponding data definition directives: [04 points]

```
; All values are 32-bit unsigned integers


do {
       A++;
      if (C == B)
          X = Y + 2;
      else
          X = Y + 10;
} while(A < B)
```

STAY BRIGHT