EE213 Computer Organization and Assembly Language
Quiz III – FALL 2018
December 7th, 2018
Section: F
Paper-A

Student Name: _____ Roll#_____

MOD=11			Effective Address Calculation			
R/M	W = 0	W = 1	R/M	MOD = 00	MOD = 01	MOD = 10
000	AL	AX	000	(BX) + (SI)	(BX) + (SI) + D8	(BX) + (SI) + D16
001	CL	CX	001	(BX) + (DI)	(BX) + (DI) + D8	(BX) + (DI) + D16
010	DL	DX	010	(BP) + (SI)	(BP) + (SI) + D8	(BP) + (SI) + D16
011	BL	BX	011	(BP) + (DI)	(BP) + (DI) + D8	(BP) + (DI) + D16
100	AH	SP	100	(SI)	(SI) + D8	(SI) + D16
101	CH	BP	101	(DI)	(DI) + D8	(DI) + D16
110	DH	SI	110	DIRECT ADDRESS	(BP) + D8	(BP) + D16
111	BH	DI	111	(BX)	(BX) + D8	(BX) + D16

ADD	0000 00DW
ADD reg/mem,imm	1000 000W (Ext 000)
MOV	1000 100DW
MOV reg/mem,imm	1100 011W (Ext 000)
MUL	1111 011W (Ext 100)
SUB	0010 10DW
SUB reg/mem,imm	1000 100W (Ext 101)
POP reg16	0101 1000
POP mem16	1000 1111 (Ext 000)
PUSH reg16	0101 0000
PUSH imm	0110 1000
PUSH mem16	1111 1111 (Ext 110)

1. Provide machine language (in hex-decimal) for the following x86 instructions **[14 Points]**

a. `ADD CH,AH`
0000 0000 11 100 101
00E5h

b. `MOV BYTE PTR[BP+108h],0Ah`
1100 0110 10 000 110 ← 08 01 ← 0A
C6 86 08 01 0Ah

c. `MUL DX`
1111 0111 + 010
F7 + 2
F9h

d. `SUB BYTE PTR [BX+DI+1709h], 1h`
1000 1000 10 101 001 ← 09 17 ← 01
88 A9 09 17 01h

e. `POP WORD PTR [DI+1CEh]`
1000 1111 10 000 101 ← CE 01
8F 85 CE 01h

f. `SUB mem16, 1709h`
1000 1001 00 101 110 ← 09 17
89 2E 09 17h

g. `PUSH SI`
0101 0000 + 110
50 + 6
56h

2. What is the role of IRQ when discussing interrupts? [2 Points]

Answer: IRQ defines priority of an interrupt.

3. Calculate the square of average of third column of following 2D array in EDX, assuming the given array is a byte array [4 Points]

45	32	33	3	19	45
01	12	76	12	23	43
20	100	18	81	98	33
190	11	43	67	13	15

```
Rowlength = 6
col_length = 4
col_index = 2

.CODE
       MOV    ebx, OFFSET array
       ADD    ebx, (TYPE array*col_index)
       MOV    ecx, col_length-1
       MOV    eax, [ebx]

       L1:
       ADD    ebx, TYPE array*Rowlength
       ADD    eax, [ebx]
       LOOP L1

       MOV    DX, 0
       MOV    CX, col_length
       DIV    CX
       MOVZX  EDX, AL
```

4. Write a procedure that should calculate and replace each of the following elements with their mathematical thrice (x3) without using LOOP, make use of string primitive instructions: **[4 Points]**

SQUARES SBYTE 4,9,-16,25,36,-49,64, 81,-100,121

```
P1 PROC
       MOV    EDI, OFFSET squares
       MOV    ESI, EDI
       MOV    EBX, 3
       MOV    ECX, LENGTHOF squares
       L1:
       LODSD                ;      MOV EAX,[esi]
       MUL    EBX
       STOSD                ;      MOV [edi],EAX
       DEC    ECX
       CMP    ECX,0
       JA     L1
       Ret
P1 ENDP
```