EE213 Computer Organization and Assembly Language
Quiz II_**Section-G**_Fall 2019
October 31, 2019
Paper-A
Student Name: _____Roll#_____

1. Given the following array write a recursive procedure to replace each of the negative elements in the array with its mathematical positive value. [4 Points]

```
array       WORD  1,-2,-3,4,-5,6,-7,8,-9,-10
```

2. In the following instructions sequence, show the resulting value of **AL/AX** where indicated, in hexadecimal. [**4 Points**]

```
mov     al, 0Fh
not     al
rol     al, 3                   ; al = _____

mov     al, 31h
and     al, 74h
stc
rcr     al, 4                   ; al = _____

mov     al, 18h
test    al, 0ffh
xor     al, 0ffh
sar     al, 4                   ; al = _____


mov     ax, 1E71h
mov     cx, 4B08h
shrd    ax, cx, cl              ; ax = _____
```

3. Write the equivalent assembly code for following procedure and draw out the stack frame. Do not use ENTER/LEAVE, USES, and LOCAL directives. (Assume ESP = 11FF CD40h, and EBP = 7FED 7FEDh, initially). [4 Points]

```
void main(){
      char a[]={1,2,3,4,5};
      cubes(a);
}
```

```
void cubes(char arr[]){
      for(int i=0;i<5;i++)
            arr[i] = arr[i] * arr[i] * arr[i];
      }
```

```
main   proc
       push   ebp
       mov    ebp,esp
       sub    esp, 8

       mov    [ebp-4], 1
       mov    [ebp-5], 2
       mov    [ebp-6], 3
       mov    [ebp-7], 4
       mov    [ebp-8], 5
       INVOKE cubes, offset [ebp-4]

       add    esp, 4       ;cleaning passed arguments
       mov    esp, ebp     ;cleaning local data
       pop    ebp
       ret
main   endp

squares PROC, p:ptr byte
       push   ebp
       mov    ebp,esp
       mov    esi, p       ;pointer to x[]
       mov    cx, 5

       L1:    mov    ax,0
              mov    al, [esi]
              movzx  bx, al
              mul    al
              mul    bx
              mov    [esi], al
              sub    esi, 1
       loop   L1
       pop    ebp
       ret
squares       ENDP
```

11FF CD3C	Ret address(system)	MAIN'S STACK FRAME
11FF CD38	7FED 7FED (ebp)	
11FF CD34	1	
11FF CD33	2	
11FF CD32	3	
11FF CD31	4	
11FF CD30	5	
11FF CD2F	11FF CD34 (Argument)	Squares' Stack Frame
11FF CD2B	Ret Address (main)	
11FF CD27	C101 00F8 (ebp)	