FAST- National University of Computer & Emerging Sciences, Karachi
School of Computing,
Mid Term I Examinations, Fall 2020
21st October, 2020, 01:15 pm – 02:15 pm.

Course Code: EE 229	**Course Name:** Computer Organization and Assembly Language
Instructor: Dr. Muhammad Nouman Durrani, Muhammad Danish Khan, Shoaib Rauf	
Student's Roll No:	**Section:**

Instructions:

- Attempt all questions and return the question paper with the answer copy.
- Read each question completely before answering it. There are **3 questions on 2 pages.**
- In case of any ambiguity, you may make assumption. But your assumption should not contradict any statement in the question paper.
- All the answers must be solved according to the SEQUENCE given in the question paper, otherwise points will be deducted.
- Where asked for values, only provide the **hex-decimal** values.

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

==========

Question No. 1

Briefly answer each of the following questions, examples are necessary where asked. **[5 x 2 = 10 Points]**

(i) Briefly discuss two types of applications that would be better suited to assembly language than a high-level language.

(ii) At which level does assembly language appear in the virtual machine level?

(iii) Why does memory access take more machine cycles than register access?

(iv) How do we override the declare size of an operand while moving larger values into smaller destinations? Give an Example.

(v) Give one example instruction for each of the following addressing modes:

a. Indirect Addressing [ESI]

b. Base indexed ARR[ESI]

Question No. 2 **[2 X 5 = 10 Points]**

(i) Write an assembly language procedure to find the missing elements in the Fibonacci Series.

Hint: Fib(n) = Fib(n – 1) + Fib(n – 2)

Fibonacci Series: 1, 1, ___, ___, ___, ___, ___.

```
MOV ESI, OFFSET FIB
MOV EDI, OFFSET [FIB+4]
MOV ECX, 5
L1: MOV EAX, [ESI]
    MOV EDX, [EDI]
    ADD EAX, EDX
    MOV [EDI+4], EAX
    ADD  ESI, 4
    ADD EDI,4
LOOP L1
```

(ii) Given the following WORD sized arrays (Mid1 and Mid2) where every element represents the obtained marks in a particular course for that exam. Write an Assembly Language Program, that find sum of obtained marks in both exams of each course, and store them into a third DWORD sized array MidTermTotal.

```
Mid1          WORD 10h, 17h, 13h, 15h, 20h, 16h
Mid2          WORD 12h, 13h, 14h, 16h, 18h, 16h
MidTermTotal  DWORD 6 Dup(?)
```

```
MOV ESI, 0
MOV ECX,6
L1:   MOV AX, [MID1+ESI*TYPE Mid1]
      ADD AX, [MID2+ESI*Mid2]
      MOVZX [MidTermTotal+ESI*TYPE MidTermTotal], AX
      INC ESI
LOOP L1
```

Question No. 3

Consider the following data segment (starting from 00000000h) and code segment for the following questions. Also consider, ESP= 0FFF2020, and initially flags are cleared. **[5 x 2 = 10 Points]**

```
.data
arr1        BYTE        0FFh, 10h, 87h
arr2        WORD        2 DUP(?)
arr3        DWORD       $, 0F11970Ah
```

```
.code
            main PROC
00FFC10F     MOV EAX, 0
00FFC113     MOV AL, [arr1+1]
00FFC117     MOV     EDX,[arr2+8]
            ;EDX = 0F11970A
00FFC11A     ADD AL, AL; 20h
00FFC11F     MOV ECX, 0Ch

00FFC123     PUSH ECX
00FFC125     CALL P1
00FFC126     JMP L2
00FFC127     L1: ADD DL, 1
00FFC12A     ADD AL, 2
00FFC12C     LOOP L2
00FFC129     L2: POP ECX
            main ENDP
```

```
            P1 PROC
0000727C     PUSH EAX ; 0000 0020h pushed
0000727F     MOV AL, arr1; 0FF
00007382     INC AL  ; AL = 00 ZF=1, PF=1

00007386     XCHG DL, DH ; EDX =0F11 0A97
00007389     XCHG DX, WORD PTR [arr2+8]
            ;DX=970A,[arr2+8] = 0F110A97h
0000738B     POP EAX
0000738E     RET
            P1 ENDP
            END main
```

(i) Discuss the instruction execution cycle for the instruction located at memory address 0000727C.

The instruction (PUSH EAX) is fetched and decoded first, the operands (EAX and ESP) are fetched then where EAX provides with the data and ESP provides the location to write to. Instruction is executed and finally data is stored onto the stack pointed by ESP.

(ii) Draw the stack diagram, when the instruction located at address 0000727C is executed.

0FFF 2020	**0000 000C**	; Value of ECX
0FFF 201C	**00FFC126**	; Return Address
0FFF 2018	**0000 0020**	; Value of EAX

(iii) Write down the values of EAX, ECX and EDX registers after the above code is executed?

EAX = 20h

ECX = 0Ch

EDX = 0F11 970Ah

(iv) What will be the value of Status Flag ZF, CF, SF and PF after the instruction located at address 007382 is executed?

ZF = 1 CF = 0 SF= 0 PF = 1

(v) Draw the memory map (byte by byte) for array **arr3** after the above code is executed.

0000 0007	07
0000 0008	00
0000 0009	00
0000 000A	00
0000 000B	97
0000 000C	0A
0000 000D	11
0000 000E	0F

*****STAY BRIGHT*****