===================================================================================================

**Question No. 1**

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

1. Briefly discuss two types of applications that would be better suited to assembly language than a high-level language.
2. At which level does assembly language appear in the virtual machine level?
3. Why does memory access take more machine cycles than register access?
4. How do we override the declare size of an operand while moving larger values into smaller destinations? Give an Example.
5. Give one example instruction for each of the following addressing modes:
6. Indirect Addressing
7. Base indexed

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

1. 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, \_\_\_, \_\_\_, \_\_\_, \_\_\_, \_\_\_.

1. 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(?)

**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 |  | P1 PROC |
| 00FFC10F | MOV EAX, 0 | 0000727C | PUSH EAX |
| 00FFC113 | MOV AL, [arr1+1] | 0000727F | MOV AL, arr1 |
| 00FFC117 | MOV EDX,[arr2+8] | 00007382 | INC AL |
| 00FFC11A | ADD AL, AL | 00007386 | XCHG DL, DH |
| 00FFC11F | MOV ECX, 0Ch | 00007389 | XCHG DX, WORD PTR [arr2+8] |
| 00FFC123 | PUSH ECX | 0000738B | POP EAX |
| 00FFC125 | CALL P1 | 0000738E | RET |
| 00FFC126 | JMP L2 |  | P1 ENDP |
| 00FFC127 | L1: ADD DL, 1 |  | END main |
| 00FFC12A | ADD AL, 2 |  |  |
| 00FFC12C | LOOP L2 |  |  |
| 00FFC129 | L2: POP ECX |  |  |
|  | main ENDP |  |  |
|  |  |  |  |

1. Discuss the instruction execution cycle for the instruction located at memory address 0000727C.
2. Draw the stack diagram, when the instruction located at address 0000727C is executed.
3. Write down the values of EAX, ECX and EDX registers after the above code is executed?
4. What will be the value of Status Flag ZF, CF, SF and PF after the instruction located at address 007382 is executed?
5. Draw the memory map (byte by byte) for array **arr3** afterthe above code is executed.

**\*\*\*STAY BRIGHT\*\*\***