National University of Computer & Emerging Sciences, Karachi
FAST School of Computing,
Assignment_02

Due Date: Thursday October 31,2024

Course Code: EE2003	**Course Name:** Computer Organization and Assembly Language
Instructor: Shoaib Rauf, Kashan, Aashir Mahboob, Atiya, Muhammad Kariz,Muhammad Usman, Nauraiz Subhan	
Student's Roll No:	**Section:**

Instructions:

- Attempt all questions and return the question paper with the answer sheet
- 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(a): Given that **EAX = 0000 000Dh**, **ECX = 0000 00DDh**, **EDX = 0000 ABCDh**, and **ESP = 0000 0FFFh**, Draw the runtime stack (diagram) for the given instruction. Also display the values of EAX, EBX, ECX and EDX at the end of the program. **[CLO: 1]** [**4 Points**]

```
Main PROC
      ADD   AX, 1
      INC   DH
      PUSH  EAX
      PUSH  ECX
      CMP   CH,0
      JNZ   L1
      PUSH  ECX
      POP   EBX
L1:   NOT     DL
      PUSH  EDX
      POP     EAX
Main ENDP
END Main
```

Question No 1(b): You need to draw the stack diagram for the following code, specifically mention IP (Instruction Pointer) value and the stack status whenever a Procedure is CALL or RET. **[CLO: 1]** [**4 Points**]

```
.DATA
array DWORD 1, 2, 3, 4, 5
INTEGER_COUNT equ 5

.CODE
Main PROC
00401000   call   Clrscr
00401005   mov    esi,OFFSET array
0040100A   mov    ecx,5
0040100F   mov    eax,0
00401014   call   ArraySum
00401019   push   0

Main ENDP
```

```
Clrscr PROC
00401023  ;Logic for clearing the screen
00401028  ret

Clrscr ENDP

ArraySum PROC USES esi ecx
0040102B   mov   eax,0
L1:
00401031   add   eax,dword ptr [esi]
00401037   add   esi,4
00401031   loop  L1
0040103B   ret
ArraySum ENDP
```

	END Main

Question No 2(a): Suppose, you have a BYTE array of characters or string. You need to take a character from the user and search it in the array. If that particular character found in the array just replace it with the @.
If the character not found then display a message. Sorry, number not found in the list or array otherwise display the updated array or string. **[7 Points]**

.data
list byte "computer organization and assembly language",0
toReplace byte ? ; input a character that need to be replace by a space
msgFound byte "Here is the updated array or string",0
msgNFound byte "Sorry, character not found in the array or string",0

Question No 2(b): Convert the following Code in x86 assembly. **[7 Points]**

```
int myarray[100];
while(j>=0 && j<=100)
{
    myarray[j+1] = myarray[j];
    j = j-1;
}
```

Question No 3(a): **[4 Points]**
For the given Instructions, Write the contents of AL, BL and CL Registers along with carry flag.

```
MOV  CL,2
MOV  AL,8Ch
MOV  BL,C8h
SHL  AL,CL
SHR  BL,CL
INC  CL
SAR  BL,CL
ROL  AL,CL
CLC
DEC  CL
RCL  AL,CL
STC
RCR  BL,CL
SHRD AL,BL,2
SHLD BL,AL,2
```

Question No 3(b): **[4 Points]**
The time stamp field of a file directory entry uses bits 0 through 4 for the seconds, bits 5 through 10 for the minutes, and bits 11 through 15 for the hours. Write instructions that extract the seconds, minutes and hours and copy the values to a WORD variable named Seconds, Minutes and Hours Respectively.

GOOD LUCK!