Name: _____   Roll Number: _____   Section: _____

National University of Computer and Emerging Sciences, Lahore Campus

	Course:		Course	
		Computer Organization and Assembly Language	Code:	EE2003
			Semester:	Sp'2022
	Program:	BS (CS)	Total Marks:	30
	Duration:	60 Minutes	Weightage:	15
	Paper Date:	May-2022	Page(s):	8
	Section(s):	All	Section:	_____
	Exam:	Midterm II	Roll No:	_____

Instruction/Notes:

- Exam is Open book, Open notes.

- Properly comment your code.

- You **CANNOT** use an instruction **NOT** taught in class.

- If there is any ambiguity, make a reasonable assumption. Questions during the exam are not allowed.

- Write your answer in the space provided. You **can take extra sheets BUT they WON'T BE ATTACHED WITH THE QUESTION PAPER OR MARKED.**

- All other rules pertaining to examinations as per NUCES policy apply.

Question 1 [15 Marks]: Short Questions

i. **[3 marks]** Consider a subroutine fibo that uses the stack to pass the parameters, and to return the output values. The pseudocode for the subroutine is as follows:

int fibo(int x){
if (x==1 || x==0)
     return x;
else
     return fib(x-1)+fib(x-2);
}

In order to call the subroutine, you have to pass a parameter (size = 1 word), and return a result (size = 1 word). Write some assembly language statements that are required before and after calling this subroutine.

sub sp, 2 ; create space for result on stack

push bx; assume that bx contains the input parameter

call fibo ; calling the subroutine

pop ax ; store the result in ax register

ii.   **[4 marks]** Suppose that AX=0x1122, BX=0x0100, CX= 0x2341, and SP=0xffff. Give the contents of AX, BX, and SP after executing the following instructions:

	AX	BX	SP
push cx	0x1122	0x0100	0xfffd
push bx	0x1122	0x0100	0xfffb
xor ax, bx	0x1022	0x0100	0xfffb
pop bx	0x1022	0x0100	0xfffd
pop ax	0x2341	0x0100	0xffff
push cx	0x2341	0x0100	0xfffd

iii.   **[3 Marks]** Consider the code given below and information on when keyboard was pressed during the execution of this code, write out the sequence in which the instructions are executed. Each executable instruction in code is numbered so your answer should be as follows:

Sample answer:

Instructions executed in following order

I11

I6

….

You also have to briefly explain the working of the program.

			Order of execution:
	[org 0x0100]		I1
I1		jmp start	I19
			INT9 handler is invoked
I2	kbisr:	push ax	I20-I27 (kbisr is hooked
I3		push es	for INT9)
I4		mov ax, 0xb800	
I5		mov es, ax	
I6		in al, 0x60	
I7		cmp al, 0x2a	
I8		jne nextcmp	
I9		mov byte [es:0], 'L'	
I10		jmp nomatch	
I11	nextcmp:	cmp al, 0x36	
I12		jne nomatch	
I13		mov byte [es:0], 'R'	
I14	nomatch:	mov al, 0x20	
I15		out 0x20, al	
I16		pop es	
I17		pop ax	
I18		iret	
I19	start:	xor ax, ax	
		;---assume that keyboard was pressed by user at this point	
I20		mov es, ax	
I21		cli	
I22		mov word [es:9*4], kbisr	
I23		mov [es:9*4+2], cs	
I24		sti	
I26		mov ax, 0x4c00	
I27		int 0x21	

iv.   **[5 Marks]** In the code given below, we are copying the data of video memory from one location to another using string instructions. As a result of the execution of this code, what will be the changes on the screen?

```
[org 0x0100]
        jmp start

movepixels:
        push ax
        push bx
        push cx
        push si
        push di
        push es
        push ds

        mov ax, 0xb800
        mov es, ax
        mov ds, ax
        mov si, 0
        mov di, 20
        mov bx, 0

        ; (code is continued in the second column)
```

```
loop1:
        mov cx, 20
        cld
        rep movsb
        add si, 140
        add di, 140

        add bx, 1
        cmp bx, 5
        jne loop1

        pop ds
        pop es
        pop di
        pop si
        pop cx
        pop bx
        pop ax
        ret

start:
        call movepixels

        mov ax, 0x4c00
        int 0x21
```

==Solution: In the first five rows, the data displayed on the first 10 video memory cells is copied onto the next 10 video memory cells.==

Question 2 [15 Marks]: Draw a rectangle with the help of two given points i.e. A (xA, yA) and B (xB, yB).

   i.   **[4 Marks]** Before printing the rectangle, check that yA must be less than yB and xA must be less than xB.

   ii.   **[3 Marks]** Clear screen with white background.

iii.   **[8 Marks]** Write a subroutine to print the boundary of the rectangle with blue dollar characters (ASCII= 24-Hex,36-Decimal), and to fill red color inside of rectangle. Please note that the parameters should be passed
through the stack.

Example 1:
Input: A (7, 8) and B (10, 11)

Output :   *(7,8)*
```
          $       $       $       $
          $                       $
          $                       $
          $       $       $       $
                              (10,11)
```

Example 2:
Input: A (10, 11) and B (7, 8)
Output: No printing on screen