Assignment 2

Individual/independent
Due Date: APR 2, 2023
Assignment 1: **15** points of total mark

Assignment 1 consist of two sections:
PART1: Answering questions [40%]
PART2: Coding [60%]

For the coding section, you may work with TAs. Please consider that TAs will assist with any problems and errors; TA will not be assistant on answering the solutions.

How submit your solution for assignment 2:
PART1:

- For the first part submit solution as a single PDF file and upload it on mycourselink.
- Solutions MUST be typed; no other format will be accepted.
- Save your file as per the following example of naming pattern:
  Name_FamilyName_StudentID_PART1.pdf

PART2:

- For the second part submit the solution as a single PDF file and upload it on mycourselink.
- Solutions MUST be typed; no other format will be accepted.
- Sections that asked to analyze the code just add your explanation in your solution file.
- Section that requires writing a code take screenshots of the outputs and add them to your solution file.
- Save your solution file as per the following example of naming pattern:
  Name_FamilyName_StudentID_PART2.pdf
- Upload the source code of each program with the following pattern:
  Name_FamilyName_StudentID_PROGRAMX.c
  Ex: AMIN_SAFAEI_123456_PROGRAM1.

Assignment 2 - PART1

1. What will be the value in EDX after each of the lines marked (a) and (b) execute? **(4 points)**

```
    .data
    one WORD 8002h
    two WORD 4321h
    .code
    mov   edx,21348041h
    movsx edx,one                      ;  (a)
    movsx edx,two                      ;  (b)
```

2. What will be the value in EAX after the following lines execute? **(2 points)**

```
mov eax,1002FFFFh
inc ax
```

3. What will be the value in EAX after the following lines execute? (**2 points**)

```
mov eax,30020000h
dec ax
```

4. What will be the value of the Parity flag after the following lines execute? (**2 points**)

```
mov al,1
add al,3
```

5. What will be the value of EAX and the Sign flag after the following lines execute? (**2 points**)

```
mov eax,5
sub eax,6
```

6. In the following code, the value in AL is intended to be a signed byte. Explain how the Overflow flag helps, or does not help you, to determine whether the final value in AL falls within a valid signed range. **(2 points)**

```
mov al,-1
add al,130
```

7. What value will RAX contain after the following instruction executes? **(2 points)**

```
mov rax,44445555h
```

8. (Yes/No): Is it possible to set the Overflow flag if you add a positive integer to a negative integer? **(2 points)**

9. (Yes/No): Will the Overflow flag be set if you add a negative integer to a negative integer and produce a positive result? **(2 Points)**

10. (Yes/No): Is it possible for the NEG instruction to set the Overflow flag? **(2 points)**

11. (Yes/No): Is it possible for both the Sign and Zero flags to be set at the same time? **(2 points)**

Use the following variable definitions for Questions 16–19:

```
.data
var1 SBYTE -4,-2,3,1
var2 WORD 1000h,2000h,3000h,4000h
var3 SWORD -16,-42
var4 DWORD 1,2,3,4,5
```

12. For each of the following statements, state whether or not the instruction is valid: **(8 points)**

```
a. mov ax,var1
b. mov ax,var2
c. mov eax,var3
d. mov var2,var3
e. movzx ax,var2
f. movzx var2,al
g. mov ds,ax
h. mov ds,1000h
```

13. What will be the hexadecimal value of the destination operand after each of the following instructions execute in sequence? **(2 points)**

```
mov al,var1        ; a.
mov ah,[var1+3]    ; b.
```

14. What will be the value of the destination operand after each of the following instructions execute in sequence? **(4 points)**

```
mov ax,var2        ; a.
mov ax,[var2+4]    ; b.
mov ax,var3        ; c.
mov ax,[var3-2]    ; d.
```

15. What will be the value of the destination operand after each of the following instructions execute in sequence? **(4 points)**

```
mov    edx,var4      ; a.
movzx  edx,var2      ; b.
mov    edx,[var4+4]  ; c.
movsx  edx,var1      ; d.
```

16. Using the XCHG instruction no more than three times, reorder the values in four 8-bit registers from the order A,B,C,D to B,C,D,A. **(4 points)**

17. Transmitted messages often include a parity bit whose value is combined with a data byte to produce an even number of 1 bits. Suppose a message byte in the AL register contains 01110101. Show how you could use the Parity flag combined with an arithmetic instruction to determine if this message byte has even or odd parity. **(4 points)**

18. Which statement is true about what will happen when the example code runs? **(10 points)**

```
 1: main PROC
 2:      push 10
 3:      push 20
 4:      call Ex2Sub
 5:      pop  eax
 6:      INVOKE ExitProcess,0
 7: main ENDP
 8:
 9: Ex2Sub PROC
10:      pop eax
11:      ret
12: Ex2Sub ENDP
```

a. EAX will equal 10 on line 6
b. The program will halt with a runtime error on Line 10
c. EAX will equal 20 on line 6
d. The program will halt with a runtime error on Line 11

19. Which statement is true about what will happen when the example code runs? **(10 points)**

```
 1: main PROC
 2:      mov  eax,30
 3:      push eax
 4:      push 40
 5:      call Ex3Sub
 6:      INVOKE ExitProcess,0
 7: main ENDP
 8:
 9: Ex3Sub PROC
10:      pusha
11:      mov eax,80
12:      popa
13:    ret
14: Ex3Sub ENDP
```

a. EAX will equal 40 on line 6
b. The program will halt with a runtime error on Line 6
c. EAX will equal 30 on line 6
d. The program will halt with a runtime error on Line 13

20. Which statement is true about what will happen when the example code runs? **(10 points)**

```
 1: main PROC
 2:      mov eax,40
 3:      push offset Here
 4:      jmp Ex4Sub
 5:    Here:
 6:      mov eax,30
 7:      INVOKE ExitProcess,0
 8: main ENDP
 9:
10: Ex4Sub PROC
11:       ret
12: Ex4Sub ENDP
```

a. EAX will equal 30 on line 7
b. The program will halt with a runtime error on Line 4
c. EAX will equal 30 on line 6
d. The program will halt with a runtime error on Line 11

21. Which statement is true about what will happen when the example code runs?
     (10 points)

```
1: main PROC
 2:      mov  edx,0
 3:      mov  eax,40
 4:      push eax
 5:      call Ex5Sub
 6:      INVOKE ExitProcess,0
 7: main ENDP
 8:
 9: Ex5Sub PROC
10:      pop  eax
11:      pop  edx
12:      push eax
13:      ret
14: Ex5Sub ENDP
```

a. EDX will equal 40 on line 6
b. The program will halt with a runtime error on Line 13
c. EDX will equal 0 on line 6
d. The program will halt with a runtime error on Line 11

22. What values will be written to the array when the following code executes?
     (10 points)

```
.data
array DWORD 4 DUP(0)
.code
main PROC
     mov  eax,10
     mov  esi,0
     call proc_1
     add  esi,4
     add  eax,10
     mov  array[esi],eax
     INVOKE ExitProcess,0
main ENDP
proc_1 PROC
     call proc_2
     add  esi,4
     add  eax,10
     mov  array[esi],eax
     ret
proc_1 ENDP
proc_2 PROC
```

```
        call  proc_3
        add   esi,4
        add   eax,10
        mov   array[esi],eax
        ret
proc_2 ENDP
proc_3 PROC
        mov   array[esi],eax
        ret
proc_3 ENDP
```

Assignment 2 – PART2

1. Write a program with a loop and indirect addressing that copies a string from source to target, reversing the character order in the process. Use the following variables: **(25 points)**

```
source BYTE "This is the source string",0
target BYTE SIZEOF source DUP('#')
```

2. Using a loop and indexed addressing, write code that rotates the members of a 32-bit integer array forward one position. The value at the end of the array must wrap around to the first position. For example, the array [10,20,30,40] would be transformed into [40,10,20,30]. **(25 points)**

3. Create a procedure that generates a random string of length L, containing all capital letters. When calling the procedure, pass the value of L in EAX, and pass a pointer to an array of byte that will hold the random string. Write a test program that calls your procedure 20 times and displays the strings in the console window. **(25 points)**

4. Write a procedure that produces N values in the Fibonacci number series and stores them in an array of doubleword. Input parameters should be a pointer to an array of doubleword, a counter of the number of values to generate. Write a test program that calls your procedure, passing The first value in the array will be 1, and the last value will be 2,971,215,073. Use the Visual Studio debugger to open and inspect the array contents. **(25 points)**