ID: 903952802

1. (5.8.2-1)Write a sequence of PUSH and POP instructions to exchange the values of EAX and EBX.

```
PUSH EAX
PUSH EBX
POP EAX
POP EBX2
```

2. (5.8.2-2) Suppose you wanted a subroutine to return to an address that was 3 bytes higher in memory than the return address currently on the stack. Write a sequence of instructions that would be inserted just before the subroutine's RET instruction that accomplish this task.

```
POP EAX
ADD EAX, 3
PUSH EAX

or

ADD [ESP], 3    ; (not allowed)
```

3. (4.9.2-10)Write a sequence of instructions that set both the Carry and the Overflow flags at the same time.

```
MOV EAX, -128
ADD EAX,-1     ;(any two negative numbers summing < -128)
```

4. (4.9.2-4)Write a code using byte operands that adds two negative integers and causes the Overflow flag to be set.

```
MOV EAX,-2
ADD EAX, -127     ;(any two negative numbers summing < -128)
```

5. (3.9.2-13) Declarea string variablexcontaining the word "TEST" repeated 500 times.

```
x BYTE 500 DUP("TEST")
```

6. (1.7.1-25) Create a truth table to show all possible inputs and outputs for the Boolean function described by NOT (A OR B)

Truth Table

A	B	A OR B	¬(A OR B)
T	T	T	F
T	F	T	F
F	T	T	F
F	F	F	T

7. (1.7.1-15) What is the decimal representation of each of the following signed binary numbers?

```
a)  10110111
   -01001001 = -49h = -64-9 = -73

b) 00111010
   3Ah = 58

c)  11111000
   -00001000 = -8
```

8. (4.10-5)Write a program that compiles anduses a loop to calculate the first seven values, Fib(1) to Fib(7),of the Fibonacci number sequence described by the following formula: Fib(1) = 1, Fib(2)=1, Fib(n)=Fib(n-1)+Fib(n-2).

```
INCLUDE Irvine32.inc
.data
FIB BYTE8DUP(?)
.code
main PROC
mov [FIB + TYPE FIB],1
mov [FIB + 2 * TYPE FIB],1
mov ecx,5
mov esi,3
next:
mov al, [FIB + esi - TYPE FIB]
add al, [FIB + esi - 2 * TYPE FIB]
mov [FIB + esi], al
add esi, TYPE FIB
loop next
main ENDP
END main
```