National University of Computer and Emerging Sciences

Laboratory Manual

for

Computer Organization and Assembly Language Programming

(EL 213)

Course Instructor	Ms. Aatira Anum
Lab Instructor(s)	Mr. Rasaal Ahmad Mr. Muhammad Salman Mubarik
Section	H
Semester	Fall 2023

Department of Computer Science

FAST-NU, Lahore, Pakistan

Exercise 1: [Bit Manipulation] Calculate the number of one bits in BX and complement an equal number of least significant bits in AX using MASK. HINT: Use the XOR instruction.

Sample Run:

Initial value of BX	Total No of 1 Bits in BX	Initial value of AX	AX after Complementing 7 least significant bits
1011 0001 1000 1001	7	1010 1011 1**010 0101**	1010 1011 1**101 1010**

Exercise 2: You need to perform bit by bit comparison of two words. If the two words are equal then dx =1 otherwise, dx =0.

Exercise 3: Declare a 32byte buffer containing random data. Consider for this problem that the bits in these 32 bytes are numbered from 0 to 255. Declare another byte that contains the starting bit number. Write a program to copy the byte starting at this starting bit number in the AX register. Be careful that the starting bit number may not be a multiple of 8 and therefore the bits of the desired byte will be split into two bytes.

Exercise 4: [Extended Subtraction] Write a program for subtracting 64 bits given below.

Initially				
num1:	**1000 1000 0000 0000**	**0110 0000 1111 1111**	**0100 0000 0000 0000**	**1111 1111 1111 1111**
num2:	**1000 1111 0000 1111**	**0000 0000 0000 0000**	**0100 0000 0000 0001**	**1000 0000 0000 0000**
result:	**0000 0000 0000 0000**	**0000 0000 0000 0000**	**0000 0000 0000 0000**	**0000 0000 0000 0000**

Exercise 5: [Extended Multiplication] Write a program to multiply two 32-bit numbers and store the answer in a 64-bit location.

Sample Run:

```
        a:      dq 0xABCDD4E1    ; dq allocates 64-bit memory space. a is 32-bit number but
        it has space allocation of 64 bits
        b:      dd 0xAB5C32      ; 32-bit space for multiplier
result: dq 0x0                   ; result should be 0x73005CB8FF6FF2 verify on calculator
programmer's view
```

Exercise 6:

Initialize AX with last 4 digits of your roll number (for example, if your roll number is 16L-1105 then AX should be initialized with 1105). Store AX in BX. Make a memory variable f, initialize it with 0 and compute

$$F = (A||B)\ \&\&\ (A \odot 0x1BCD)$$

|| is bitwise OR operation, && is bitwise AND operation whereas ⊙ is bitwise XOR operation.

Exercise 7: Fill the following table. These instructions are from same program and are not independent. Write the corresponding output for the given registers' and flags' values.
AX=0x5CAA DX=0x3729 CX=0x235A

Instructions	Updated value after executing the instruction			Flag values after the instruction execution		
	AL	DL	CL	CF	OF	SF
xor al, dl						
add dl, dl						
sub cl, dl						
sar al, cl						
adc al, dl						