Laboratory Manual for
Computer Organization and Assembly Language

Department of Computer Science
National University of Computer and Engineering Sciences
Chiniot-Faisalabad Campus

Lab 2 Section C

Objectives

- Assemble, Link and Run assembly source code using EMU8086
- Pracice Problems

INSTRUCTIONS

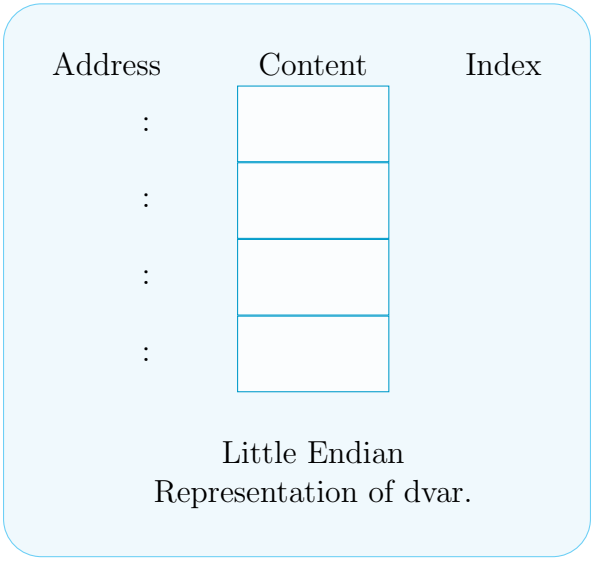
- Attach the `ONLY` word file containing the answers, codes, screenshots, and Roll No.
- 10% penalty for not following the naming rules mentioned in the manual.

Problems

Problem 1: Assemble the given program and give answers the questions given below.

```
 .model small
.stack 100h
.data
bvar db 40h
wvar dw 5246h
dvar dd 56781234h
y equ 8
count = 10h * 10h
num1 dw $
.code
mov ax, @data
mov ds, ax
mov ax, 5040h
mov bx, wvar
mov bl, byte ptr wvar
add ax, word ptr dvar
mov bvar, 25h
mov y, 30h
mov bx, 00h
add bx, y
mov cx, count
mov dx, num1
mov ax, offset dvar
mov bx, offset wvar
mov ah, 4ch
int 21h
```

Problem 1.1: Arrange the contents of memory location `dvar` in memory?



Little Endian
Representation of dvar.

Problem 1.2: What is the value of num1?

Problem 1.3: In Instruction `mov bl, byte ptr wvar`, why is there need of PTR operator?

Problem 1.4: Is there any error in instruction `mov y, 30h`?

Problem 1.5: What is the offset address of `wvar` and `count`?

Problem 2: Assemble the given program and give answers the questions given below.

```
 .model small
.stack 100h
.data
barr db 10h, 20h, 30h, 40h
warr dw 2465h, 5612h, 7869h
darr dd 12345678h, 78695421h
num1 db 2 dup (3)
num2 db 3 dup (?)
.code
mov ax, @data
mov ds, ax
mov ah, barr+1
mov al, barr+4
mov bx, warr
mov bx, warr+1
mov bx, warr+2
mov dx, offset num1+2
mov bl, num2+2
mov num2+2, 04h
add num2+2, 015h
mov ch, num2+2
add warr, 5612h
mov ah, 4ch
int 21h
```

Problem 2.1: After the execution of instruction `mov al, barr+4`, what actually moves in `AL` register?

Problem 2.2: What is the value of `BX` register after the instruction `mov bx, warr+1`?

Problem 2.3: After the exuction of whole program write down the updated value of array `num2`?

Problem 3: Write a program in assembly code that generate the first six Fibonacii number sequence `(1, 1, 2, 3, 5, 8)` by an array named Fibonacci using `DUP` directive. The rule to generate the sequence is

$$F_n = F_{n1} + F_{n2}$$

Hint: Initialize the array with `0` and then place 1 in first two places. Further use the above rule on the same array.

Problem 4: Insert the following variables in your program:

```
.data
Uarr WORD 1010h, 2020h, 3030h, 4040h
Sarr WORD -2, -3, -4, -5
```

Write instructions that moves the four values in `Uarr` to the `AX, BX, CX, DX` registers. It should be like this

```
AX = 1010
BX = 2020
CX = 3030
DX = 4040
```

Next, write instructions that moves the four values in `Sarr` to the `AX, BX, CX, DX` registers. It should be like this

```
AX = FFFE
BX = FFFD
CX = FFFC
DX = FFFB
```

You are done with your exercises, make your submission.