UNIVERSITY OF INFORMATION TECHNOLOGY AND SCIENCES

LAB Report

COURSE TITLE – Microprocessor Lab

COURSE CODE – CSE 360

Submitted To

Md. Ismail
Lecturer of UITS

Submitted By

*Md. Shoyaif Rahman (**0432310005101050**)*

Semester: *Autumn-2025 (6th)*
Department: *CSE*
Batch: *53*

Section:*6A2*

Date: 27/08/25

Assembly Language Lab Report

Lab report: 02

Experiment Name: addition and difference of two numbers Assembly Language

1. Process

Adding two numbers:

2. Read **first number** from user (single character).
3. Move cursor to a **new line**.
4. Read **second number** from user (single character).
5. Add the two numbers and adjust ASCII (sub 48).
6. Move cursor to a **new line**.
7. Print the **result**.

subtracting two numbers:

1. Read **first number** from user (single character).
2. Move cursor to a **new line**.
3. Read **second number** from user (single character).
4. Add the two numbers and adjust ASCII (sub 48).
5. Move cursor to a **new line**.
6. Print the **result**.

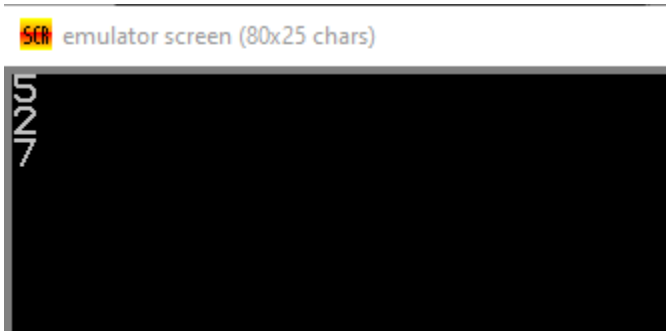
2. Implementation (Program Code – ASM)

1.Adding two numbers

```
org 100h


 mov ah,1

 int 21h

 mov bl,al
```

```
mov ah,2
mov dl,010
int 21h     ;new line
mov dl,013
int 21h

mov ah,1
int 21h
mov bh,al


add bl,bh
sub bl,48

mov ah,2
mov dl,010
int 21h    ;new line
mov dl,013
int 21h

mov ah,2
mov dl,bl
int 21h

ret
```

output:



2.Subtrating two numbers

```
org 100h

 mov ah,1
 int 21h
 mov bl,al

 mov ah,2
 mov dl,010
 int 21h     ;new line
 mov dl,013
 int 21h

 mov ah,1
 int 21h
 mov bh,al


 sub bl,bh
```

```
add bl,48

mov ah,2
mov dl,010
int 21h    ;new line
mov dl,013
int 21h

mov ah,2
mov dl,bl
int 21h

s

ret
```

output:

- The program correctly reads two single-digit numbers from the user.

3.Result

- For the **addition program**, the sum of the two input numbers is displayed as output.
    - Example: Input `4` and `3` → Output `7`.
- For the **subtraction program**, the difference between the two input numbers is displayed as output.
    - Example: Input `8` and `5` → Output `3`.
- Both programs successfully handle ASCII to numeric conversion and display results on a new line.

4. Conclusion

1. We successfully took two numbers as input from the user using INT 21h, AH=1.
2. In the **first program**, we performed **addition** of two ASCII digits and displayed the result using INT 21h, AH=2.
3. In the **second program**, we performed **subtraction** of two ASCII digits and displayed the result.
4. Learned how to handle **ASCII to numeric conversion** (sub 48) for addition and **numeric to ASCII conversion** (add 48) for subtraction.
5. Used **Line Feed (10h) and Carriage Return (13h)** to print outputs neatly on a new line.
6. This demonstrates basic arithmetic operations and output formatting in **8086 Assembly**.