Experiment Name: Input two number and add them using assembly language with proper prompt messages.

Objective: The objective of adding two numbers in assembly language programming is to perform the arithmetic operation of addition on two given values and obtain their sum. This fundamental operation is at the core of many computational tasks and algorithms.

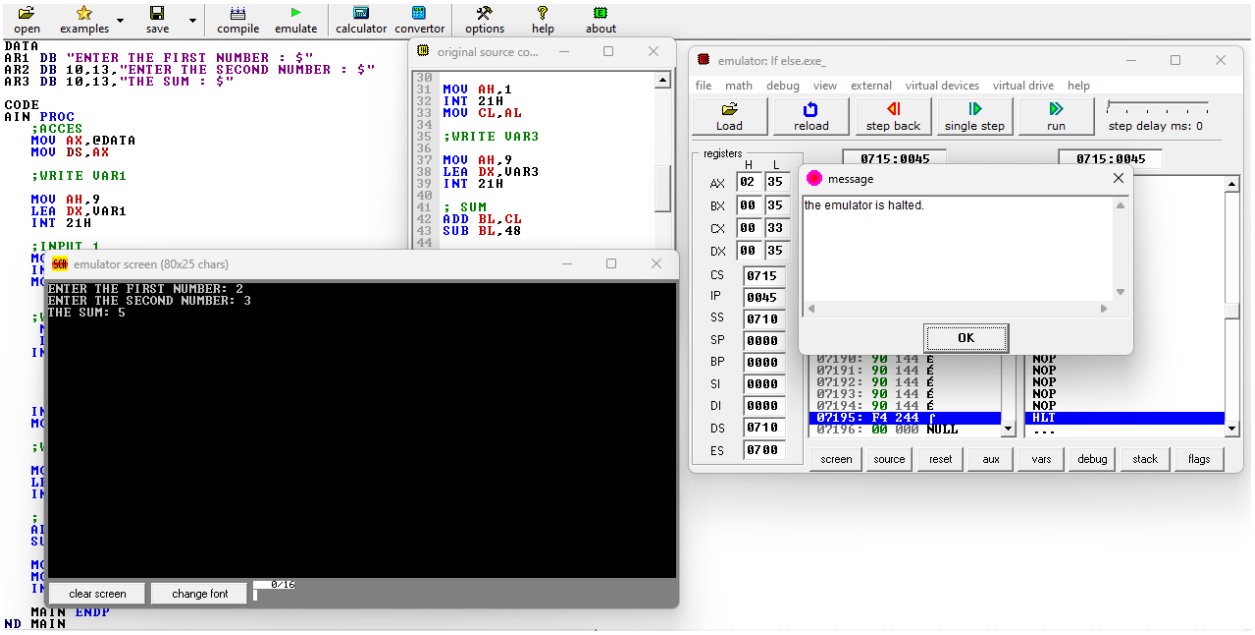
Theory: Adding two numbers is a fundamental arithmetic operation performed in computer programming to compute the sum of two numeric values. In assembly language programming, this operation is carried out at the hardware level using the arithmetic and logical unit (ALU) of the central processing unit (CPU). The ALU processes binary representations of numbers and performs addition.

Source code:

```
.DATA
VAR1 DB "ENTER THE FIRST NUMBER : $"
VAR2 DB 10,13,"ENTER THE SECOND NUMBER : $"
VAR3 DB 10,13,"THE SUM : $"

.CODE
MAIN PROC
   MOV AX,@DATA
   MOV DS,AX

   MOV AH,9
   LEA DX,VAR1
   INT 21H

   MOV AH,1
   INT 21H
```

```
    MOV BL,AL



    MOV AH,9
    LEA DX,VAR2
    INT 21H

     MOV AH,1
    INT 21H
    MOV CL,AL

    MOV AH,9
    LEA DX,VAR3
    INT 21H

    ADD BL,CL
    SUB BL,48

    MOV AH,2
    MOV DL,BL
    INT 21H

    MAIN ENDP
END MAIN
```

Emulator and output:



Experiment Name: Input a number and check whether it is greater than the given number or not using assembly language .

Objectives: The objective of a program written in assembly language to check whether a given number is greater than another given number is to implement a basic comparison operation that evaluates the relationship between the two numbers. This operation can serve various purposes, including decision-making, control flow, and data filtering.

Theory: The goal of writing an assembly language program to check whether a given number is greater than another number involves utilizing the hardware-level capabilities of a computer's central processing unit (CPU) to perform a comparison operation. Assembly language provides a low-level interface to the CPU, allowing precise control over operations and data manipulation.

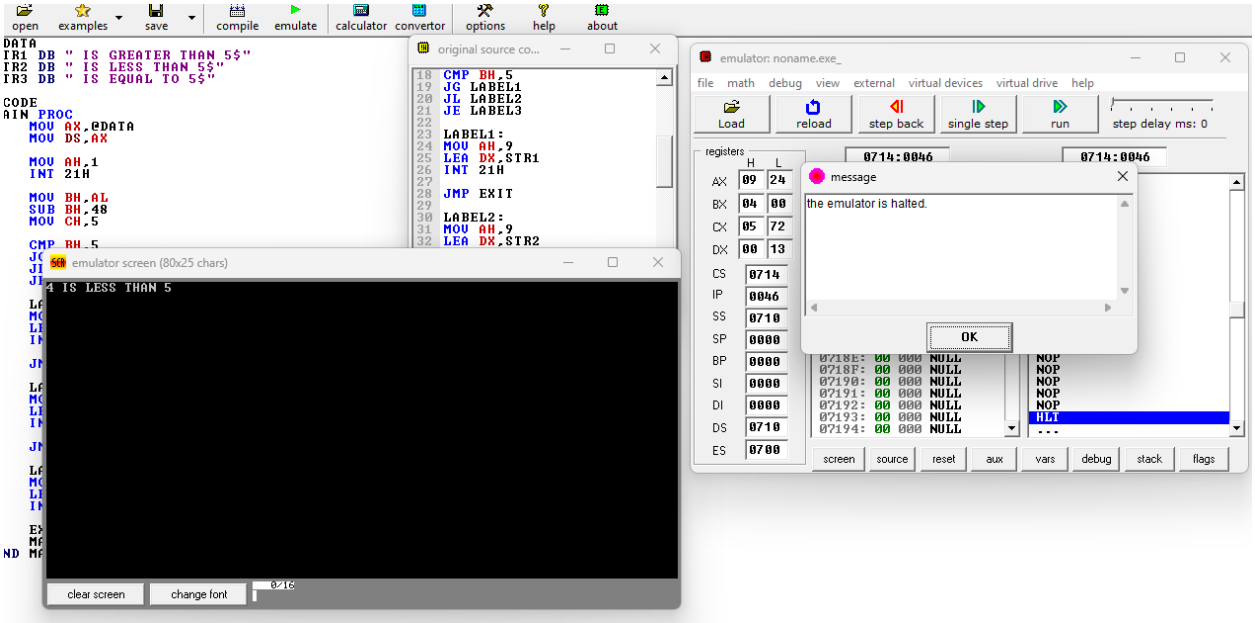
Source code:

```
.DATA

STR1 DB " IS GREATER THAN 5$"

STR2 DB " IS LESS THAN 5$"

STR3 DB " IS EQUAL TO 5$"


.CODE

MAIN PROC

    MOV AX,@DATA

    MOV DS,AX


    MOV AH,1

    INT 21H


    MOV BH,AL
```

```
SUB BH,48
MOV CH,5

CMP BH,5
JG LABEL1
JL LABEL2
JE LABEL3

LABEL1:
MOV AH,9
LEA DX,STR1
INT 21H

JMP EXIT

LABEL2:
MOV AH,9
LEA DX,STR2
INT 21H

JMP EXIT

LABEL3:
MOV AH,9
```

```
    LEA DX,STR3

    INT 21H


    EXIT:

    MAIN ENDP

END MAIN
```

Emulator and output:



Bangladesh University Of Business And Technology

Lab Report

Course Title : Microprocessor and Interfacing Lab

Course Code : CSE - 316

Experiment No : 02

Experiment Name : Input two number and add them using assembly language with proper prompt messages.

Submitted To:	**Submitted By :**
Ms. Nowrin khandaker	**Name :** Md Masud Rana
Assistant Professor,	**Intake :** 48
Department Of Computer	**Section** : 02
Science And Engineering,	**Id :** 21224103162
BUBT	Department Of CSE

Submitted On : 09/08/2023

Bangladesh University Of Business And Technology

Lab Report

Course Title : Microprocessor and Interfacing Lab

Course Code : CSE - 316

Experiment No : 03

Experiment Name : Input a number and check whether it is greater than the given number or not using assembly language .

Submitted To:	**Submitted By :**
Ms. Nourin Khandaker	**Name :** Md Masud Rana
Lecturer,	**Intake :** 48
Department Of Computer	**Section** : 02
Science And Engineering,	**Id :** 21224103162
BUBT	Department Of CSE

Submitted On : 09/08/2023