Green University of Bangladesh

Department of Computer Science and Engineering(CSE)

Faculty of Sciences and Engineering

Semester: (Spring, Year: 2021), B.Sc. in CSE (Day)

LAB REPORT NO : 03

Course Title: Microprocessors and Microcontrollers Lab

Course Code: CSE 304 **Section:** PC-DD

Lab Experiment Name: Introduction of understanding the use of loop.

Student Details

Name	Id
Md. Romzan Alom	201902144

Lab Date: 14.11. 2021

Submission Date: 28.11.2021

Course Teacher's Name: Rusmita Halim Chaity

[For Teachers use only: Don't Write Anything inside this box]

Lab Report Status	
Marks: ……	Signature:.....
Comments:.....	Date:.....

Title of the Lab Experiment: Introduction of understanding the use of loop.

Objectives / Aim:

We learn about loop from this experiment. We can take user input and we can perform various operations repeatedly on this input and show the result as output.

Procedure / Analysis / Design:

Instruction MOV AX, BX is used moves data BX to AX and data is stored in AX.

Instruction DIV BX is used division. Here, AX is divided by BX and Quotient is stored in AL and Remainder is stored in AH.

Instruction MUL BX is used multiplication. Here, BX and AX are multiplied and this data is stored in AX.

Instruction ADD AX, BX is used addition. Here, add to BX and AX and this data is stored in AX.

Instruction SUB AX, BX is used subtraction. Here, BX and AX are subtracted and this data is stored in AX.

Instruction MOV AH, 1 and INT 21H are used for user input and it stored AX.

Instruction MOV DX, AX and MOV AH, 2 and INT 21H are used for show output. Here, DX store AX data and compiler understand DX data then it show DX data as output.

Instruction LEA DX, R and MOV AH, 9 and INT 21H are used for call variable and show output the variable data.

Instruction MOV AH,2 and MOV DL,0DH and INT 21H and MOV DL,0AH and INT 21H are used for newline.

Instruction MOV AH,2 and MOV DL,0DH and INT 21H and MOV DL,0AH and INT 21H are used for newline.

Instruction RESULT_1: are created a level.

Instruction LOOP START are used for looping. It goes to START level and this level decrease with CX.

Instruction V DB 0AH, 0DH, “The result is: $" are used for newline and take a message. Here, 0AH, 0DH, is used newline.

Problem-01

Write an assembly code to take a number input from the user (number must be from 0-9). Print from 0 to that number.

Pseudo-code:

```
.MODEL SMALL

.STACK 100H

.DATA

 n db ?


 R1 DB "Enter the number of input: $"


 R2 DB 0AH,0DH,"The result of our pragram: $"


 R3 DB 0AH,0DH,"$"


.CODE

        MAIN PROC
```

```
mov ax, @DATA
 mov ds, ax
    lea dx,R1
    mov ah,9
    int 21H
    mov ah,1
    int 21H
    sub al,30H
    mov n,al
    lea dx,R3
    mov ah,9
    int 21H

    lea dx,R2
    mov ah,9
    int 21H
    xor cx,cx
    mov cl,n
    add cl,1

    mov n,0
```

```
    Loop_1:

    mov al,n
    xor dx,dx
    mov ah,02h
    mov dl,al
    add dl,30H
    int 21h
    inc n

    loop Loop_1

mov ah, 4ch
int 21h

MAIN ENDP
END MAIN
```

Problem-02

Take a number n from user. After that find out the factorial of that number n. (Suppose for n=5, you have to find out factorial= 1 × 2 × 3 × 4 × 5.

Pseudo-code:

```
.MODEL SMALL
.STACK 100H
.DATA

n db ?
Factorial db ?
R1 DB "Enter the number of input: $"
R2 DB 0AH,0DH,"The result of our pragram show on vars option! $"
R3 DB 0AH,0DH,"$"

.CODE
        MAIN PROC
mov ax, @DATA
```

```
mov ds, ax

   lea dx,R1
   mov ah,9
   int 21H

   mov ah,1
   int 21H
   sub al,30H
   mov n,al
   lea dx,R3
   mov ah,9
   int 21H

   lea dx,R2
   mov ah,9
   int 21H
   xor cx,cx
   mov cl,n
   mov bl,0
```

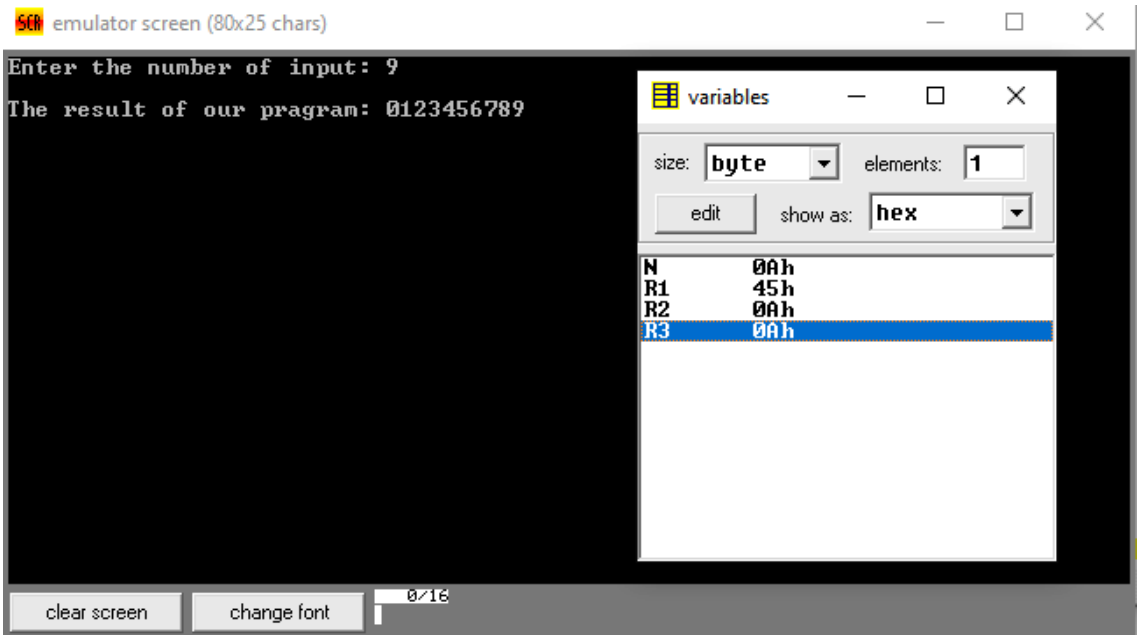
```
    mov al,1
    Loop_1:
    mul n
    dec n
    loop Loop_1
    mov Factorial , al

 mov ah, 4ch
 int 21h

 MAIN ENDP
 END MAIN
```

Calculation:

We will try to find the factorial of a user input and print 0 to a user number that take user as input. For find factorial and print 0 to a user number we can use loop and that loop has a level and that level operate that operation.

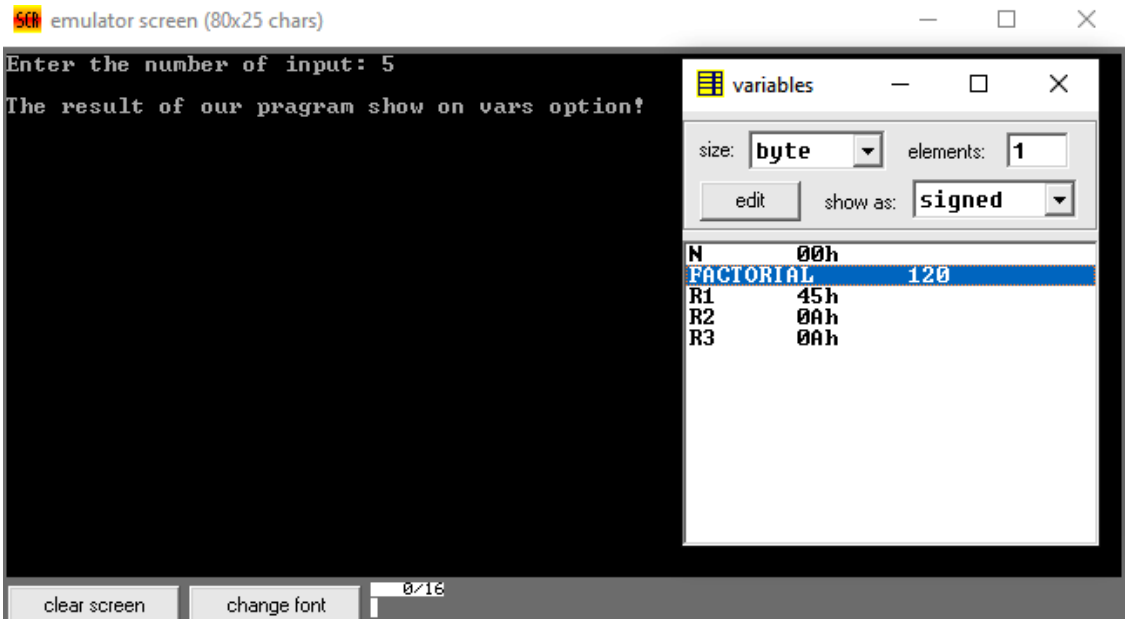
Test Result / Output:

Problem-1 output:



The input number is 9. So, it should have been = 0 1 2 3 4 5 6 7 8 9.
We see our output and calculation output are same. So, it is 100% right outputs.

Problem-2 output:



The input number is 5. So, the factorial of 5 is 120 and it should have been = (1*2*3*4*5) = 120.
We see our output and calculation output are same. So, it is 100% right outputs.

Analysis and Discussion:

1. Due to Covid-19 situation, we can't do this experiment directly. So, it is completely based on software.

2. Since, it is done with Software. So it may have some Software and Mechanical errors.

3. To emulate those codes I am facing so many problems for hexadecimal number.

4. To print 0 to a user number we have create a loop and a level that is very complex for me.

5. We can use loop but loop decreases with CX register. That is really confusing to think when loop will stop.

6. To find factorial number I face some problem to multiple each number.

7. From our compile show only char value of hexadecimal number. That why we face some problem to multiple some numbers.

8. Network problem. Cause of the ups and downs of the internet we could not attend the class properly. That why I saw the class recording but this video was very bad quality. This reason I have so many confusion.

Summary:

From those problems, we can take so many decimal numbers from user as input and print the 0 to a user number or perform various operations and find factorial on this input and show the result as output very easily which will help the user to change the program's output. Those are very important to complete this course.