MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Course Title: Microprocessor and Embedded System Lab

Course Code: ICT-2204

Experiment Name: Problem Solving with Assembly Language - IV

Lab Report No: 05

Submitted By	Submitted To
Name: Kuldip Saha Mugdha ID: IT-22018 2nd Year, 2nd Semester Session: 2021-2022 Dept. of ICT, MBSTU	Dr. Md. Abir Hossain Associate Professor DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

Date of Performance: 11/12/2024 Date of Submission: 18/12/24

Experiment no: 05

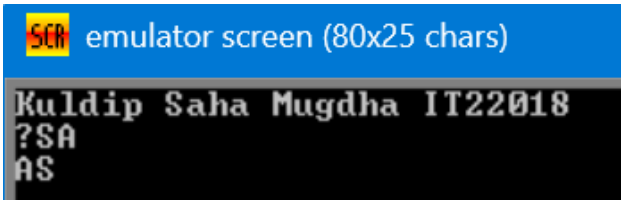
Experiment name: Problem Solving with Assembly Language – IV

Program 1: A program to display a "?", read two capital letters, and display them on the next line in alphabetical order.

Code:

```
.model small
.stack 100h
.data
m1 db 'Kuldip Saha Mugdha IT22018',0ah,0dh,'$'
ch1 db ?
.code
main proc
mov ax,@data
mov ds,ax

mov ah,9
lea dx,m1
int 21h

mov dl,'?'
mov ah,2
int 21h

mov ah,1
int 21h
mov ch1,al
int 21h
mov bl,al

mov ah,2
mov dl,0ah
int 21h
mov dl,0dh
int 21h
cmp ch1,bl
jle display

change:
xchg ch1,bl

display:
mov ah,2
mov dl,ch1
int 21h
mov dl,bl
int 21h
main endp
end main
```

Output:



Program 2: A program to display the extended ASCII characters (ASCII codes 80h to FFh). Display 10 characters per line, separated by blanks. Stop after the extended characters have been displayed once.

Code:

```
.model small
.stack 100h
.data
m1 db 'Kuldip Saha Mugdha IT22018',0ah,0dh,'$'
.code
main proc
mov ax,@data
mov ds,ax

mov ah,9
lea dx,m1
int 21h

mov cx,7fh
mov bl,80h
mov bh,0

loop:
mov dl,bl
mov ah,2
int 21h
inc bl
dec cx
inc bh

mov dl,20h
int 21h
cmp bh,10
je newline
cmp cx,0
je exit
jmp loop

newline:
mov bh,0
mov dl,0ah
int 21h
mov dl,0dh
int 21h
jmp loop

exit:
mov ah,4ch
int 21h
main endp
end main
```

Output:

Program 3: A program that will prompt the user to enter a hex digit character ("0"· ... "9" or "A" ... "F"), display it on the next line in decimal, and ask the user if he or she wants to do it again. If the user types "y" or "Y", the program repeats; If the user types anything else, the program terminates. If the user enters an illegal character, prompt the user to try again.

Sample execution:

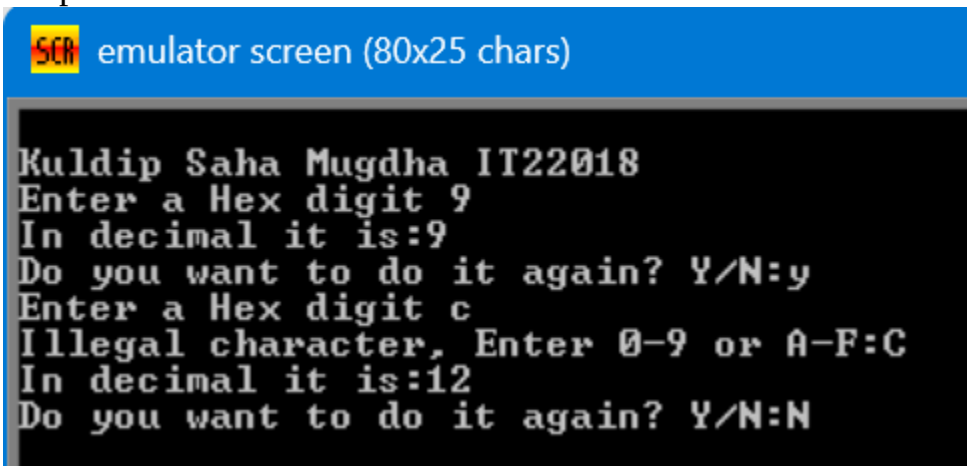
Enter a hex digit: 9
In decimal it is: 9
Do you want to do it again?: y
Enter a hex digit: c
Illegal character-Enter 0..9 or A..F: C
In decimal it is: 10
Do you want to do it again ?: n

Code:

```
.model small
.data
m db 0ah,0dh,"Kuldip Saha Mugdha IT22018 $"
m1 db 0ah,0dh,"Enter a Hex digit $"
m2 db 0ah,0dh,"In decimal it is:"
c1 db ?,"$"
m3 db 0ah,0dh,"Do you want to do it again? ","Y/N:$"
m4 db 0ah,0dh,"Illegal character, Enter 0-9 or A-F:$"
m5 db 0ah,0dh,"In decimal it is:1"
c2 db ?,"$"
.code
main proc
    mov ax,@data
    mov ds,ax

    lea dx,m
    mov ah,9
    int 21h

    Begin:
    lea dx,m1
    mov ah,9
    int 21h

    input:
    mov ah,1
    int 21h

    cmp al,'0'
    jl illegal_check
    cmp al,'9'
    jg illegal_check

    mov c1,al
    lea dx,m2
    mov ah,9
    int 21h

    msg_show:
    lea dx,m3
    mov ah,9
    int 21h

    mov ah,1    ; take input y or n
    int 21h
```

```
    cmp al,'Y'
    je Begin
    cmp al,'y'
    je Begin
    jmp end_

    illegal_check:
    cmp al,'A'
    jl illegal
    cmp al,'F'
    jg illegal

    sub al,11h
    mov c2,al
    mov ah,9
    lea dx,m5
    int 21h

    jmp msg_show

    illegal:
    mov ah,9
    lea dx,m4
    int 21h
    jmp input

    end_:
    mov ah,4ch
    int 21h
    main endp
end main
```

Output:



Program 4: Write a program using a stack to input a string and reverse it.

Code:

```
.model small
.stack 100h
.data
m db 0ah,0dh,'Kuldip Saha Mugdha IT22018',0ah,0dh,'$'
.code
main proc
    mov ax, @data
    mov ds, ax

    mov ah,9
    lea dx,m
    int 21h

    mov ah,2
    mov dl,'?'
    int 21h

    xor cx,cx

    mov ah,1
    int 21h

    while_:
        cmp al,0dh
        je end_while

        push ax
        inc cx

        int 21h
        jmp while_

     end_while:
        mov ah,2
        mov dl,0ah
        int 21h
        mov dl,0dh
        int 21h
        jcxz exit

     TOP:
        pop dx
        int 21h
        loop TOP


     exit:
        mov ah,4ch
        int 21h

    main endp
       end main
```

Output: