Mawlana Bhashani Science and Technology University

Santosh, Tangail-1902.

Lab Report

Department of Information and Communication Technology

Report No: 02

Report Name: Assembly language Program.

Course Title: Microprocessor and Assembly Language Lab

Course Code: ICT-3106

Submitted By	Submitted To
Name: **Zafrul Hasan Khan** ID: **IT-18003** Session: 2017-18 3rd Year 1st Semester Dept. of Information & Communication Technology, MBSTU.	**S.M. Shamim** **Lecturer,** Dept. of Information & Communication Technology, MBSTU.

1 . (a) Read a character and display it at the next position on the same line

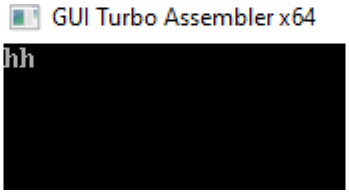
Algorithms :

1.Start the program.
2.Read a Character .
4.Display the character.
5.Stop the program.

Source Code :

```
.model small
.stack 100h
.data
.code
main proc

 mov ah,1
 int 21h

 mov ah,2
 mov dl,al ;charcter print
 int 21h

 main endp
 end main
```

Input & Output:



```
hh
```

1(b) Read an lowercase letter and display it at the next position on the same line in upper case.

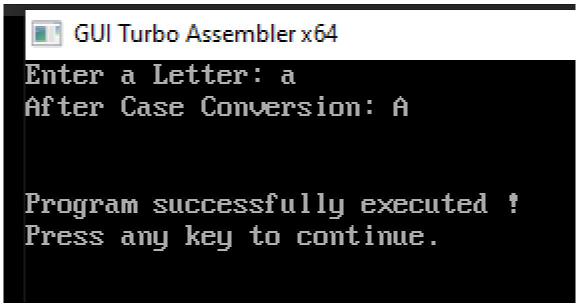
Algorithms :

1.Start the program.
2.Read a lowercase letter.
3.add 32 .
4. Then added ascii number find uppercase
5.Stop the program.

```
.MODEL SMALL
.STACK 100H
.DATA
MSG1 DB 'Enter a Letter: $'
MSG2 DB 'After Case Conversion: $'
.CODE

MAIN PROC
   MOV AX,@DATA    ;DATA SEGMENT
   MOV DS,AX

   LEA DX,MSG1
   MOV AH,9        ;MSG1
   INT 21H

   MOV AH,1
   INT 21H       ;INPUT
   MOV BL,AL

   MOV AH,2
   MOV DL,0AH
   INT 21H         ;NEW LINE
```

```
    MOV DL,0DH
    INT 21H

    LEA DX,MSG2
    MOV AH,9         ;MSG2
    INT 21H

    CMP BL,97
    JGE L1

    ADD BL,32
    MOV AH,2         ;UPPER TO LOWER
    MOV DL,BL
    INT 21H
    JMP EXIT

    L1:
    SUB BL,32
    MOV AH,2         ;LOWER TO UPPER
    MOV DL,BL
    INT 21H

    JMP EXIT

    EXIT:
    MOV AH,4CH        ;EXIT
    INT 21H
    MAIN ENDP
END MAIN
```

Output:



```
Enter a Letter: a
After Case Conversion: A


Program successfully executed !
Press any key to continue.
```

2. Write a program to a. display a "?" b. read two decimal digits whose sum is less than 10 c. display them and their sum in the next line with an appropriate message.

Algorithms :

1.Start the program.
2.Firstly display '?'.
3.Read two decimel number .
4. Then sum these number and add new line
5.Stop the program.

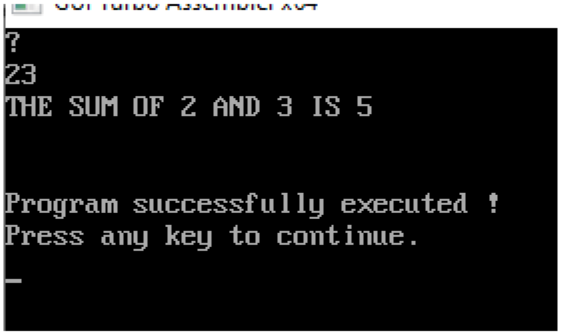
Source Code :

```
.MODEL SMALL

.STACK 100H

.DATA

   STR1 DB 0AH,0DH,'THE SUM OF '

   FIRSTNUM DB ?

   STR2 DB ' AND '

   SECONDNUM DB ?

   STR3 DB ' IS '

   ANS DB ?

   STR4 DB ' $'

.CODE

MAIN PROC
```

```
MOV AX,@DATA
MOV DS,AX

MOV AH,2
MOV DL,3FH
INT 21H



MOV AH,2
MOV DL,0AH
INT 21H
MOV DL,0DH
INT 21H

INT 21H

MOV AH,1
INT 21H
MOV BL,AL
MOV FIRSTNUM,AL
INT 21H
MOV SECONDNUM,AL

ADD BL,AL
SUB BL,30H
MOV ANS,BL

MOV AH,9
LEA DX,STR1
```

```
    INT 21H

    MOV AH,4CH
    INT 21H

    MAIN ENDP
  END MAIN
```

Output:



```
?
23
THE SUM OF 2 AND 3 IS 5


Program successfully executed !
Press any key to continue.

_
```

3. write a program to a. prompt the user b. Read first middle and last initials of a person’s name c. And display them down the left margin.

Algorithms :

1.Start the program.
2. Read three intitials input .
3. These three initials stored into FIRST, SECOND ,THIRD register respectively.
4. Then break line and show these intital left margin
5.Stop the program.

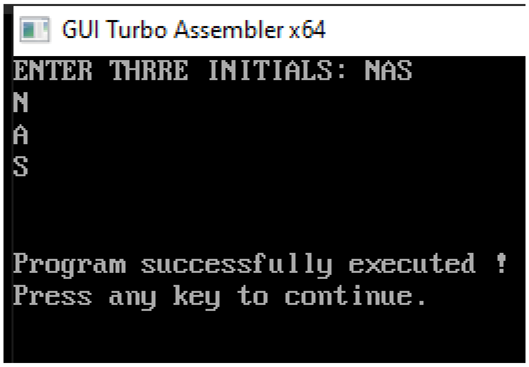
Source Code :

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

```
    STR DB 'ENTER THRRE INITIALS: $'
    STR1 DB '',0AH,0DH
    FIRST DB ?
    STR2 DB '',0AH,0DH
    SECOND DB ?
    STR3 DB '',0AH,0DH
    THIRD DB ?
    STR4 DB '$'
.CODE
MAIN PROC

    MOV AX,@DATA
    MOV DS,AX

    MOV AH,9
    LEA DX,STR
    INT 21H

    MOV AH,1
    INT 21H
    MOV FIRST,AL
    INT 21H
    MOV SECOND,AL
    INT 21H
    MOV THIRD,AL

    MOV AH,9
    LEA DX,STR1
    INT 21H
```

```
     MOV AH,4CH

     INT 21H


     MAIN ENDP

   END MAIN
```

Output:



```
ENTER THRRE INITIALS: NAS
N
A
S


Program successfully executed !
Press any key to continue.
```

4. Write an assembly program to enter one of the hex digits A-F, and display it on the next line in decimal.

Algorithms :

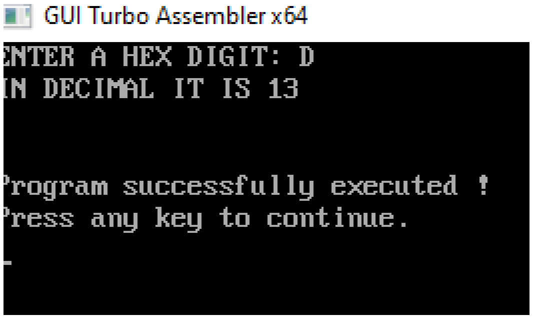
1.Start the program.
2. Input one hex digits (A-F) .
3. And subtraction 11H from these input .
4. Then its convert into binary form.
5.Stop the program.

Source Code :

```
  .MODEL SMALL

 .STACK 100H

.DATA
```

```
    STR1 DB 'ENTER A HEX DIGIT: $'
    STR2 DB 0AH,0DH,'IN DECIMAL IT IS 1'
    ANS DB ?
    STR3 DB '$'
.CODE
MAIN PROC

    MOV AX,@DATA
    MOV DS,AX

    MOV AH,9
    LEA DX,STR1
    INT 21H

    MOV AH,1
    INT 21H

    SUB AL,11H
    MOV ANS,AL

    MOV AH,9
    LEA DX,STR2
    INT 21H

    MOV AH,4CH
    INT 21H
    MAIN ENDP
END MAIN
```

Output:



```
ENTER A HEX DIGIT: D
IN DECIMAL IT IS 13


Program successfully executed !
Press any key to continue.
_
```

5. Write an assembly program to display asterisks (***********) ten times with new line.

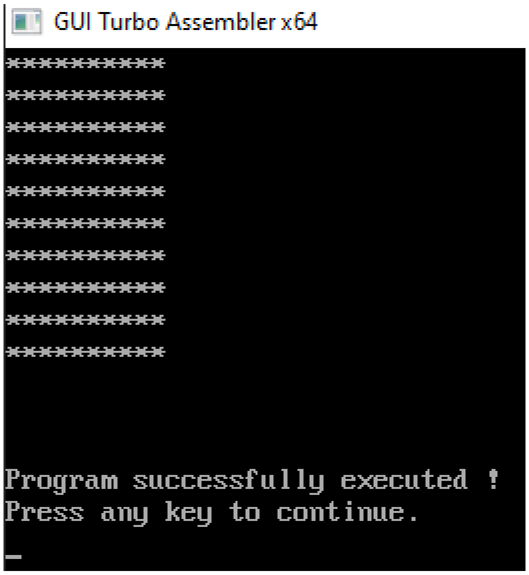
Algorithms :

1.Start the program.
2. In data segment take a string looks like (***********).
3.Then write ‘int 21h’ ten times.
4.Stop the program.

Source Code:

```
.MODEL SMALL

.STACK 100H


 .DATA

   SQUARE  DB  '**********',0DH,0AH,'$'


 .CODE

  MAIN PROC

    MOV AX, @DATA

    MOV DS, AX


    LEA DX, SQUARE              ; load the string

    MOV AH, 9
    INT 21H
    INT 21H                  ; display the string 10 times
    INT 21H
    INT 21H
    INT 21H
    INT 21H
    INT 21H
    INT 21H
    INT 21H
    INT 21H

    MOV AH, 4CH              ; return control to DOS
    INT 21H
  MAIN ENDP
END MAIN
```

__Output :__



```
**********
**********
**********
**********
**********
**********
**********
**********
**********
**********



Program successfully executed !
Press any key to continue.
_
```

6. Write an assembly program to display to (a) display"?", (b) read three initials,(z,a,f) display them in the middle of an 11 x 11 box of asterisk.

Algorithms :

1. Start the program.
2. Display ‘?’ character.
3. Read three initials (z,a,f).
4. Take asterisks in data segment

5. Load the string asterisks and Loop it 11*11 times

6.stop the program.

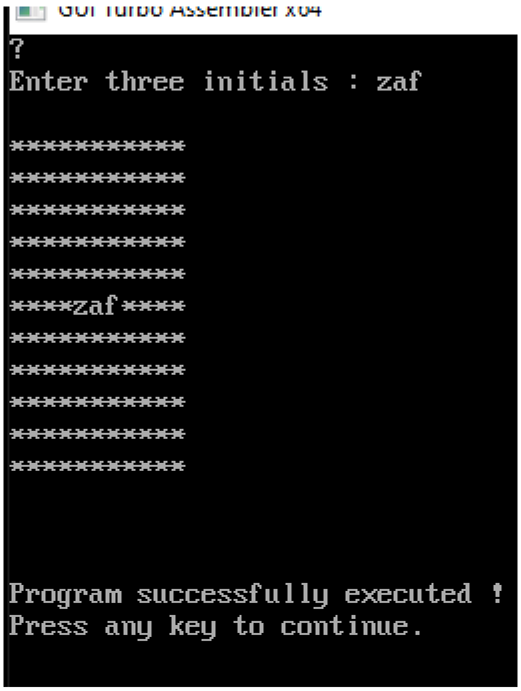
Source Code :

```
.MODEL SMALL

.STACK 100H


 .DATA

   PROMPT     DB  0DH,0AH,'Enter three initials : $'

   ASTERISKS  DB  '***********',0DH,0AH,'$'

   NEXT_LINE  DB  0DH,0AH,"$"


 .CODE

  MAIN PROC

    MOV AX, @DATA              ; initialize DS

    MOV DS, AX


    MOV AH, 2                  ; display "?"

    MOV DL, "?"

    INT 21H


    LEA DX, PROMPT             ; load and display the string PROMPT

    MOV AH, 9
```

```
            INT 21H


            MOV AH, 1
            INT 21H


            MOV BL, AL


            INT 21H


            MOV BH, AL


            INT 21H


            MOV CL, AL
            LEA DX, NEXT_LINE
            MOV AH, 9
            INT 21H
            INT 21H


            LEA DX, ASTERISKS       ; load the string ASTERISKS
            MOV AH, 9


            INT 21H                 ; display the string ASTERISKS 5 times
            INT 21H
INT 21H
INT 21H
INT 21H
```

```
    MOV ASTERISKS+4, BL        ; place the three initials in the position
    MOV ASTERISKS+5, BH        ; of middle asterisks i.e. 4,5,6.
    MOV ASTERISKS+6, CL

    INT 21H                    ; display the modified string ASTERISKS

    MOV ASTERISKS+4, "*"       ; place the "*" back in their original
    MOV ASTERISKS+5, "*"       ; position
    MOV ASTERISKS+6, "*"

    INT 21H                    ; print the string ASTERISKS 5 times
    INT 21H
    INT 21H
    INT 21H
    INT 21H

    MOV AH, 2
    MOV DL, 7H
    INT 21H

    MOV AH, 4CH
    INT 21H
   MAIN ENDP
  END MAIN
```

Output:



```
?
Enter three initials : zaf

***********
***********
***********
***********
***********
****zaf****
***********
***********
***********
***********
***********



Program successfully executed !
Press any key to continue.
```