Mawlana Bhashani Science and Technology University

Santosh, Tangail-1902.

Lab Report

Department of Information and Communication Technology

Report No: 04

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. Write an assembly program to find larger number between two numbers.

Algorithms :

1.Start the program.
2. Read the two integer number
3.Then use CMP using compare two numbers .
4. use JG for jumping to greatest number.

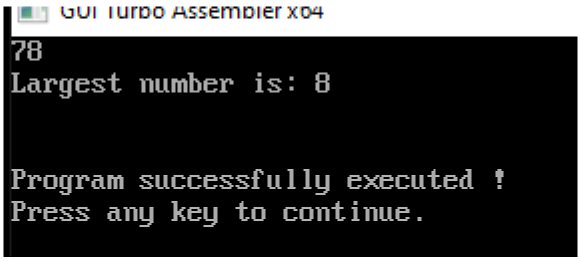
5. Stop the program.

Source Code:

```
.model small
.stack 100h
.DATA
MSG1 DB 'Largest number is: $'

.code
main proc
   MOV AX,@DATA    ;DATA SEGMENT
   MOV DS,AX

   mov ah,1
   int 21h
   mov bl,al

   mov ah,1
   int 21h
   mov bh,al

   lp:
   cmp bl,bh
   MOV AH,2
   MOV DL,0AH
   INT 21H
   MOV DL,0DH
   INT 21H

   LEA DX,MSG1
```

```
        MOV AH,9        ;MSG1
        INT 21H
        jg greater



        mov ah,2
        mov dl,bh
        int 21h
        jmp exit

        greater:

        mov ah,2
        mov dl,bl
        int 21h

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

Output:



```
78
Largest number is: 8


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

2. Write an assembly program to find small number between two numbers

Algorithms :

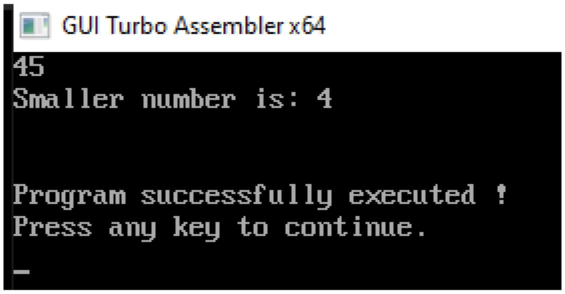
1.Start the program.
2. Read the two integer number
3.Then use CMP using compare two numbers .
4. use JL for jumping to smallest number.

5. Stop the program.

Source Code:

```
    .model small
    .stack 100h
    .DATA
    MSG1 DB 'Smaller number is: $'

    .code
    main proc
       MOV AX,@DATA    ;DATA SEGMENT
       MOV DS,AX

       mov ah,1
       int 21h
       mov bl,al

       mov ah,1
       int 21h
       mov bh,al

       lp:
       cmp bl,bh
      MOV AH,2
       MOV DL,0AH
       INT 21H
       MOV DL,0DH
    INT 21H


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



    jg smaller


    mov ah,2
    mov dl,bl
    int 21h
    jmp exit


    smaller:
    mov ah,2
    mov dl,bh
    int 21h


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

Output :



```
45
Smaller number is: 4


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

3.Write an assembly program to enter value of Al If Al contains a negative number, put -1 In Bl; if Al contains 0, put O In Bl; if Al contains a positive number, put 1 In Bl.·

Algorithms :

1.Start the program.
2. Input the one number
3.Then use CMP using check numbers by zero .
4. if it is JG, then its positive and put 1 in BL.

4. if it is JL, then its negative and put -1 in Al.

4. otherwise its zero.

5. Stop the program.

<u>Source Code :</u>

```
.model small

.stack 100h


.code

main proc

   mov ah,1

   int 21h


   cmp ax,0

   jl negative
   je zero
   jg positive

   negative:
   mov bx,-1
   jmp exit

   positive:
   mov bx,1
   jmp exit

   zero:
   mov bx,0
   jmp exit

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

3. **Write an assembly program to enter value of Al If AL contains 1 or 3, display "o"; if AL contains 2 or 4, display "e".**

Algorithms :

1.Start the program.
2. Input the one value that is entered into Al

3. If the value match with 1 and 3 then its display ‘o’ .
4. If the value match with 2 and 4 then its display ‘e’ .

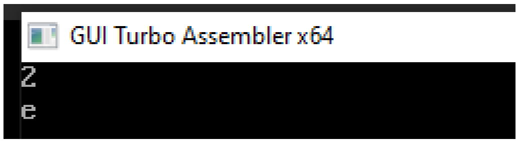
5. Stop the program.

Source Code :

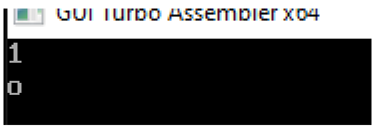
```
.model small
.stack 100h

.code
main proc
   mov ah,1
   int 21h

   cmp al,"1"
   je odd
   cmp al,"3"
   je odd
   cmp al,"2"
   je even
   cmp al,"4"
   je even

   odd:
   MOV AH,2
   MOV DL,0AH
   INT 21H
   MOV DL,0DH
```

```
    INT 21H

    mov ah,2
    mov dl,"o"
    int 21h
    jmp exit;

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

    mov ah,2
    mov dl,"e"
    int 21h
    jmp exit;

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

Input & Output :



```
2
e
```



```
1
0
```

4. Write an assembly program to enter a character if it’s an uppercase latter, display it otherwise terminate.

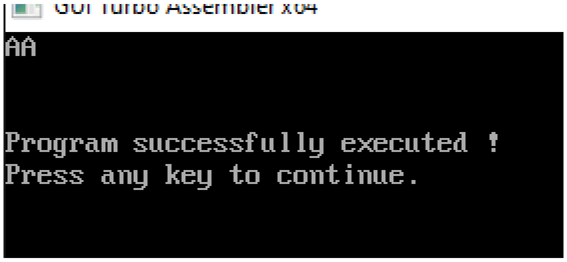
Algorithms :

1.Start the program.
2. Input a character
3. Then check the character inside the uppercase letters .
4. if it is yes , then show that .
5. Stop the program.

Source Code :

```
.model small
.stack 100h

.code
main proc
   mov ah,1
   int 21h

   lp:
   cmp al,"A"
   jnge exit
   cmp al,"Z"
   jnle exit

   mov ah,2
   mov dl,al
   int 21h

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

Output:



```
AA


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

5.Write an assembly program to enter a character if it's y or Y, display it. Otherwise terminate.

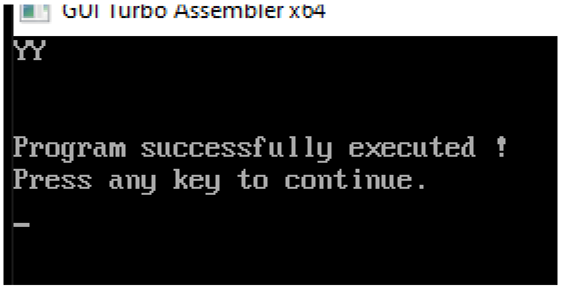
<u>Algorithms :</u>

1.Start the program.
2. Input a character
3. Then check the character , if it is 'y' or 'Y' .
4. if it is yes , then show that .
4. Otherwise terminate the program
5. Stop the program.

<u>Source Code :</u>

```
.model small
.stack 100h

.code
main proc
   mov ah,1
   int 21h

   lp:
   cmp al,"y"
   je eq
   cmp al,"Y"
   je eq
   jmp exit

   eq:
   mov ah,2
   mov dl,al
   int 21h

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

Output:



```
YY


Program successfully executed !
Press any key to continue.

_
```