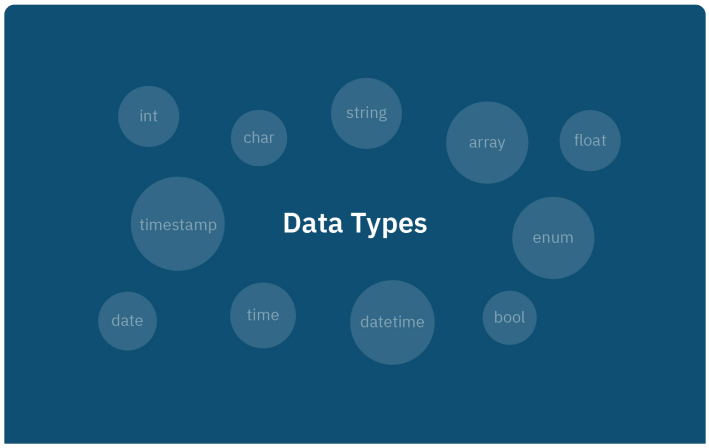
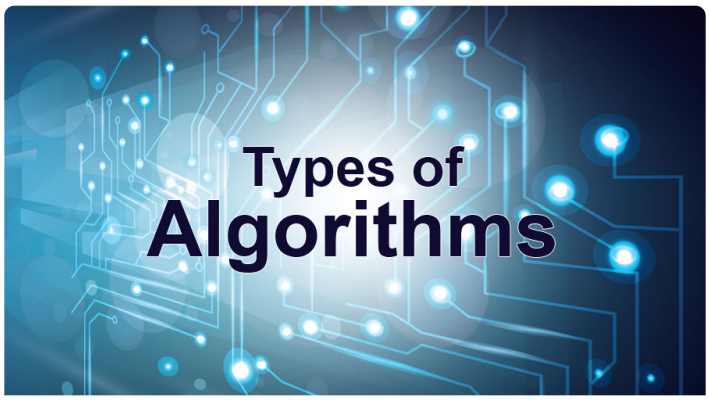
Get started with Datatypes in C



In computer science and computer programming, a data type (or simply type) is a set of possible values and a set of allowed operations on it. A data type tells the compiler or interpreter how the programmer intends to use the data. Most programming l... Read more

Get Premium access

Introduction to Algorithm in C

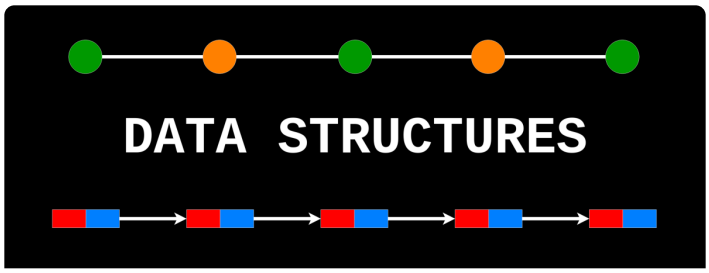


In mathematics and computer science, an algorithm is a finite sequence of rigorous instructions, typically used to solve a class of specific problems or to perform a computation.Algorithms are used as

specifications for performing calculations and d...
Read more

Get Premium access

Introduction to Data Structure in C



A data structure is a specialized format for organizing, processing, retrieving and storing data. There are several basic and advanced types of data structures, all designed to arrange data to suit a specific purpose. Data structures make it easy for... Read more

Get Premium access

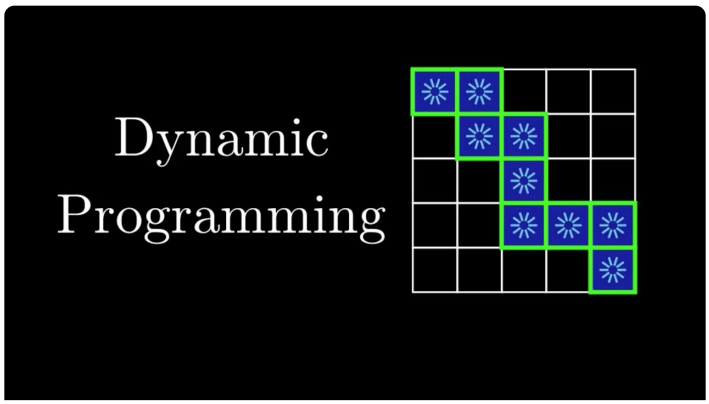
Introduction to Array in C

An array is a collection of items of same data type stored at contiguous memory locations.This makes it easier to calculate the position of each element by simply adding an offset to a base value, i.e., the memory location of the first

element of the... Read more

Get Premium access

Indroduction to Dynamic Programming



Dynamic Programming is mainly an optimization over plain recursion. Wherever we see a recursive solution that has repeated calls for same inputs, we can optimize it using Dynamic Programming. The idea is to simply store the results of subproblems, so... Read more

Get Premium access

Jahangirnagar University (JU)

Institute of Information Technology

Lab Report-4

Assembly Language

Name: Abier Farzana Hoque **Class Roll:** 1980

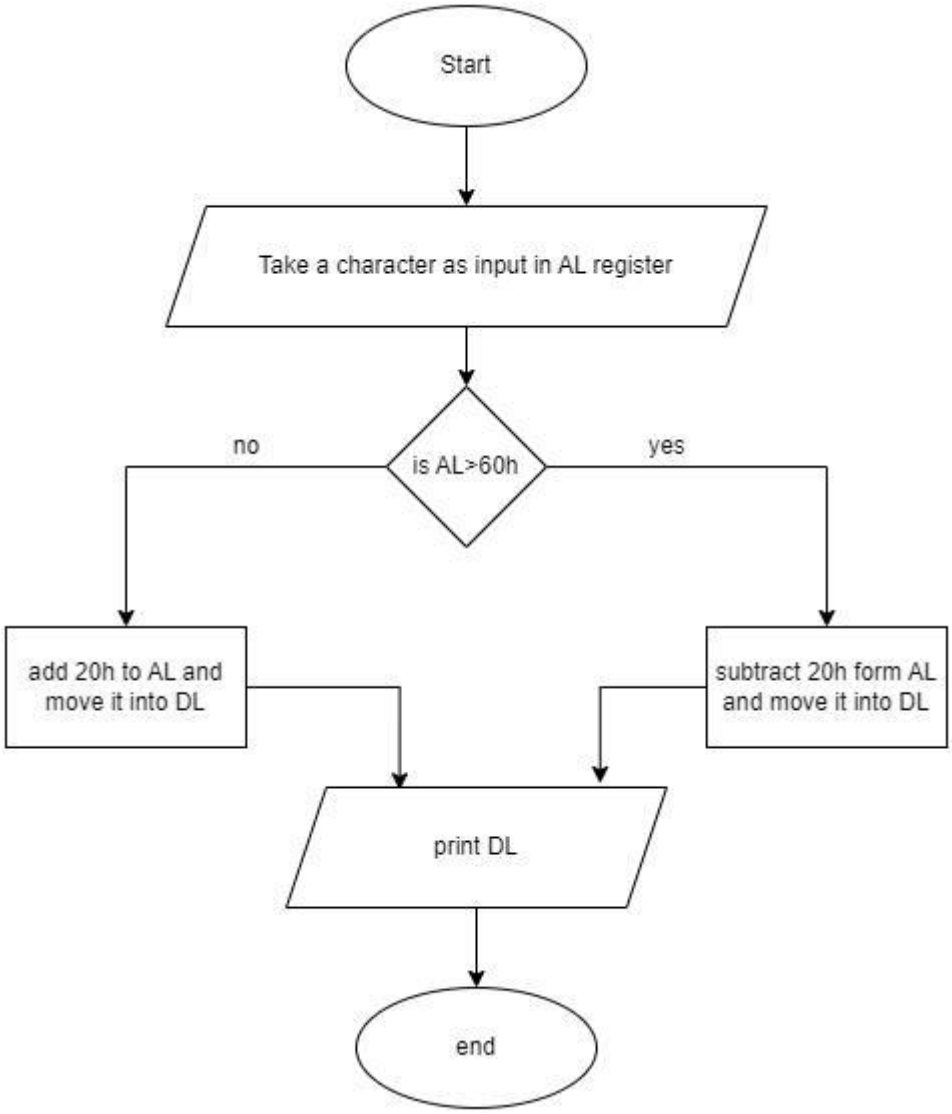
Experiment 1.

Title: Case conversion (upper case to lower case and vice versa Using an assembly language
program).

Algorithm:

Step1: Start
Step2: take an input character in AL register.
Step3: if AL>60h then subtract 20h from AL.
Step4: otherwise add 20h to AL.
Step5: print the content of AL register.
Step6: end.

Flow chart:



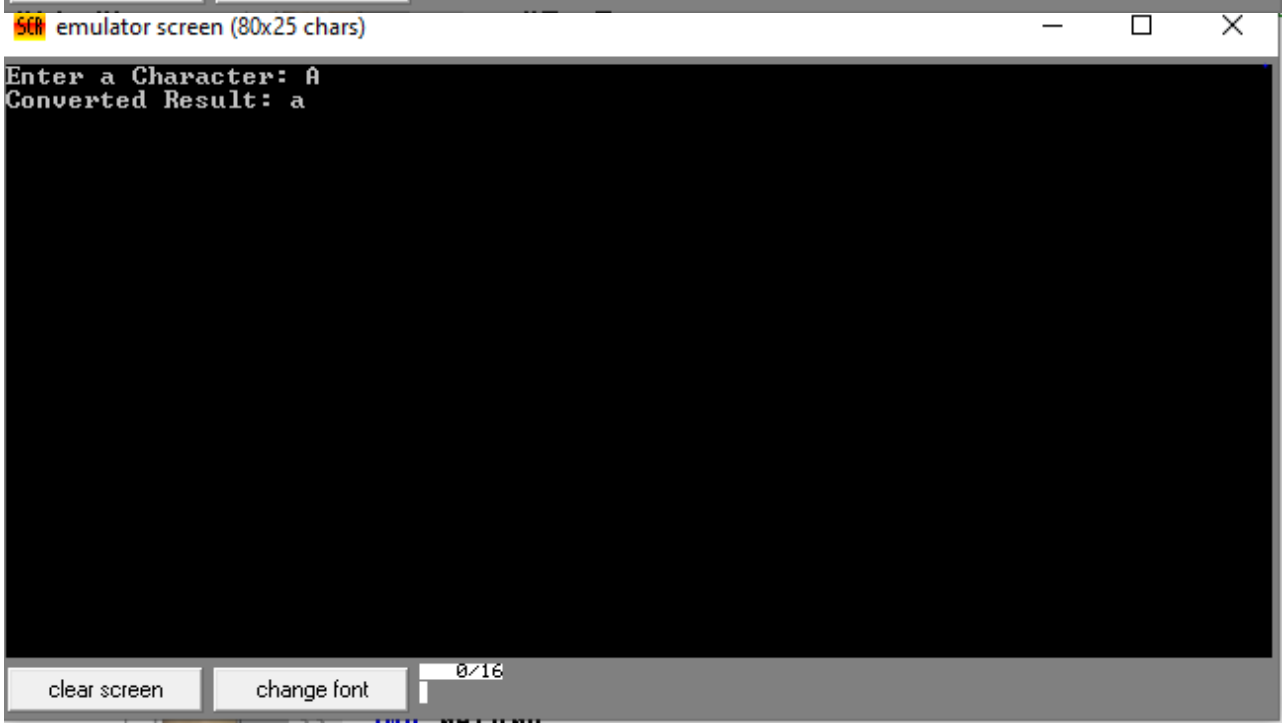
Source Code:

```
.model small
.stack 100h
.data

.code
 input db 'Take Your Input: $'
 res db 10,13,'Your Result is: $'

 main proc
 mov ax,@data
 mov ds,ax

 mov ah,9
 lea dx,input
 int 21h

mov ah,01
int 21h



mov bl,al
 cmp bl,90
 jle big_to_small
 jmp small_to_big

 big_to_small:
 add bl,32
 jmp return

mov bl,al
small_to_big:

sub bl,32

  return:

 mov ah,9
 lea dx,res
 int 21h


 mov ah,02
 mov dl,bl
 int 21h
 exit:
 mov ah,4ch
 int 21h
```

```
 main endp
end main
```

Input: Firstly take a input in lower case latter a and take another input in upper case latter A.

Output:



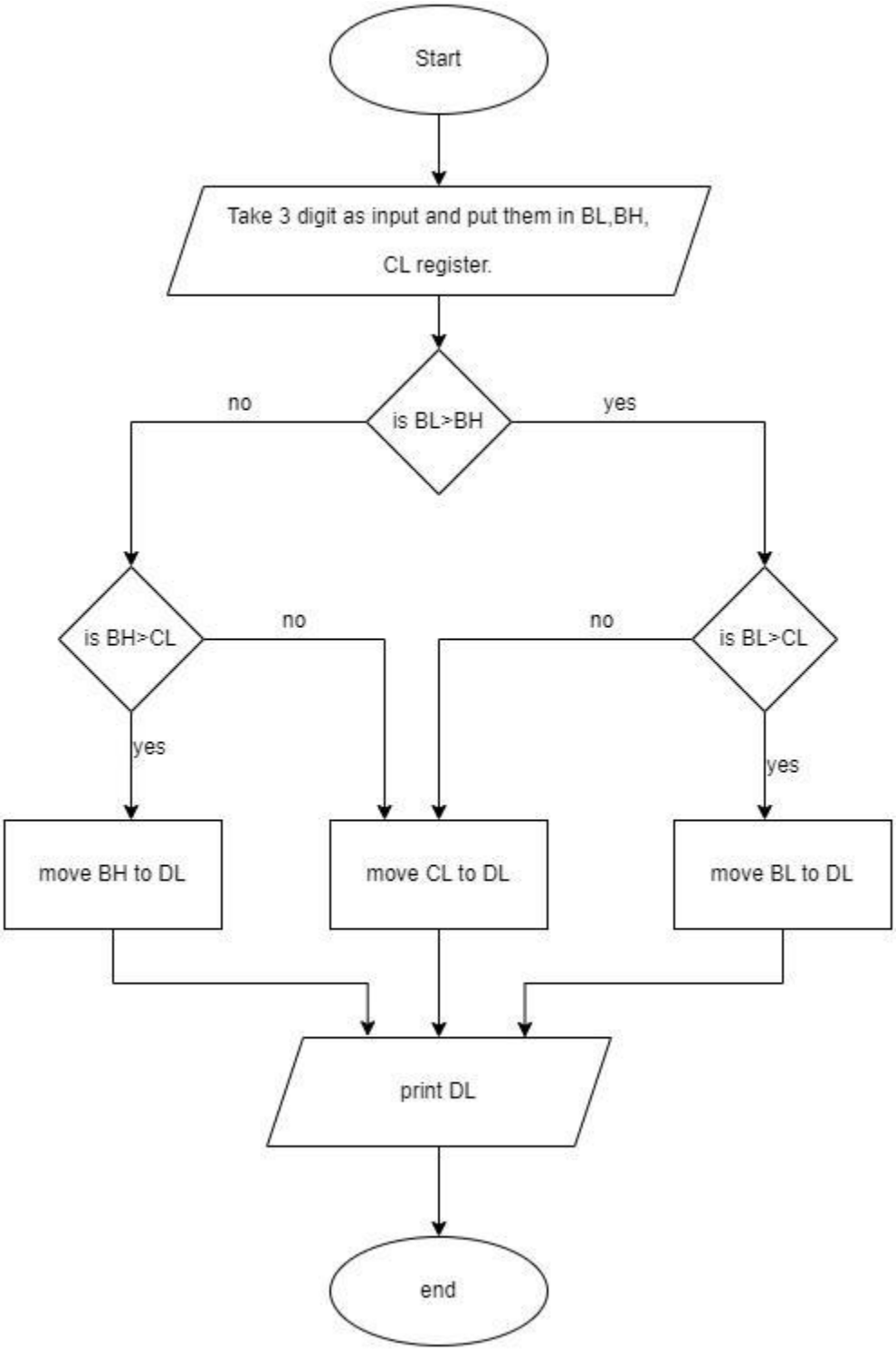
Experiment 2.

Title: Compare three digits and find the biggest number (Using an assembly language
program).

Algorithm:

Step1: Start
Step2: Take 3 digit as input and put them in BL,BH,CL register.
Step3: if BL>BH then
Step4:if BL<CL move CL to DL. otherwise move BL to DL.
Step5: otherwise,if BH<CL move CL to DL. otherwise move BH to DL.
Step6: print the content of DL.
Step7: end.

Flow chart:


Start
Take 3 digit as input and put them in BL,BH, CL register.
is BL>BH
no
yes
is BH>CL
no
no
is BL>CL
yes
yes
move BH to DL
move CL to DL
move BL to DL
print DL
end

Source Code:

```
data segment
     a db 0dh,0ah,"Enter first number $"
     b db 0dh,0ah,"Enter second number $"
     c db 0dh,0ah,"Enter third number $"
    d db 0dh,0ah,"Biggest out of three number is: $"
    data ends
code segment
    assume CS:code,DS:Data

    start:

    mov ax,Data
    mov DS,ax

    mov dx,offset a
    mov ah,09
    int 21h
    mov ah,01
    int 21h
    mov bl,al
```

```
mov dx,offset b
mov ah,09
int 21h
mov ah,01
int 21h
mov bh,al


mov dx,offset c
mov ah,09
int 21h
mov ah,01
int 21h
mov cl,al


cmp bl,bh
JNG secondbig

firstbig:

cmp bl,cl
JNG thirdbig1
```

```
mov dx,offset d
mov ah,09
int 21h

mov dl,bl
mov ah,02
int 21h
ret


thirdbig1:

mov dx,offset d
mov ah,09
int 21h

mov dl,cl
mov ah,02
int 21h
ret



secondbig:
```

```
cmp bh,cl
JNG thirdbig2
mov cl,bh

mov dx,offset d
mov ah,09
int 21h

mov dl,cl
mov ah,02
int 21h
ret


thirdbig2:
mov dx,offset d
mov ah,09
int 21h

mov dl,cl
mov ah,02
int 21h
ret
```

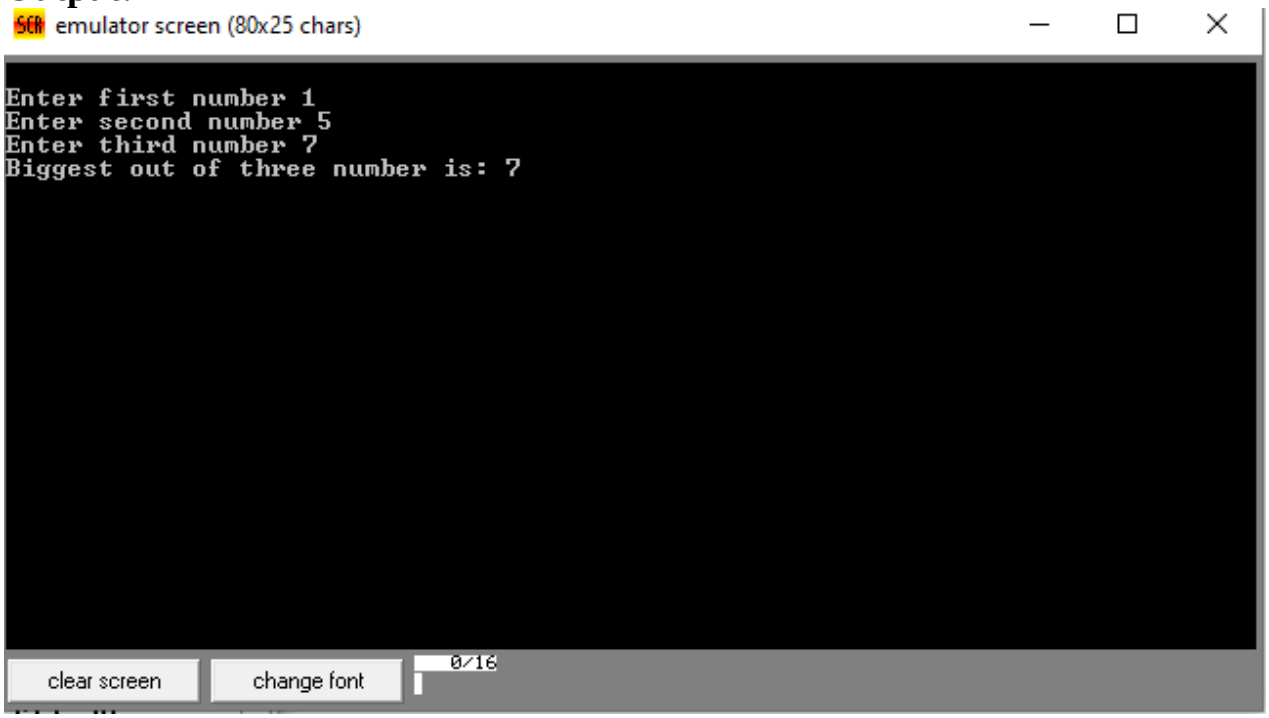
code ends

end start

Input:

Taking input 1,5,7.

Output:



Jahangirnagar University (JU)

Institute of Information Technology

Lab Report-4

Assembly Language

Name: Yeasmine Akter Mitu **Class Roll:** 1992

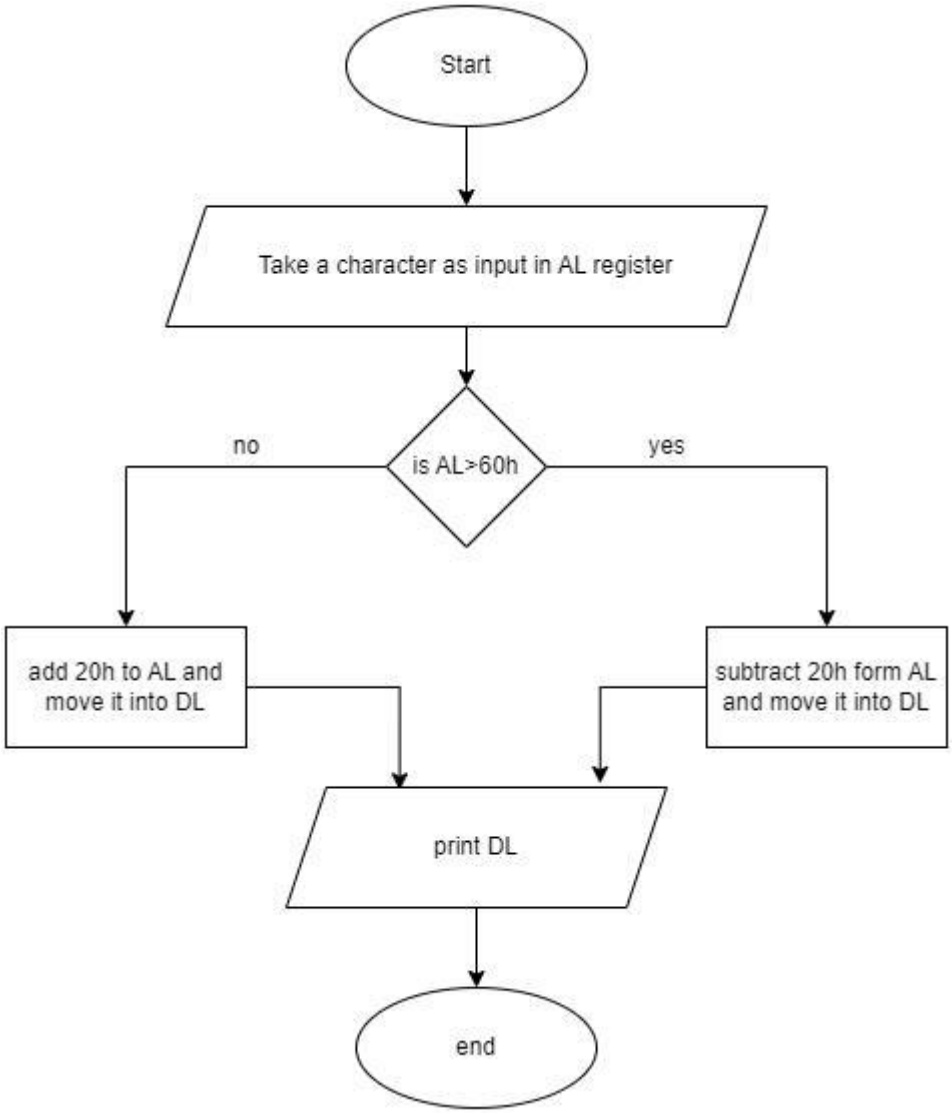
Experiment 1.

Title: Case conversion (upper case to lower case and vice versa Using an assembly language program).

Algorithm:

Step1: Start
Step2: take an input character in AL register.
Step3: if AL>60h then subtract 20h from AL.
Step4: otherwise add 20h to AL.
Step5: print the content of AL register.
Step6: end.

Flow chart:



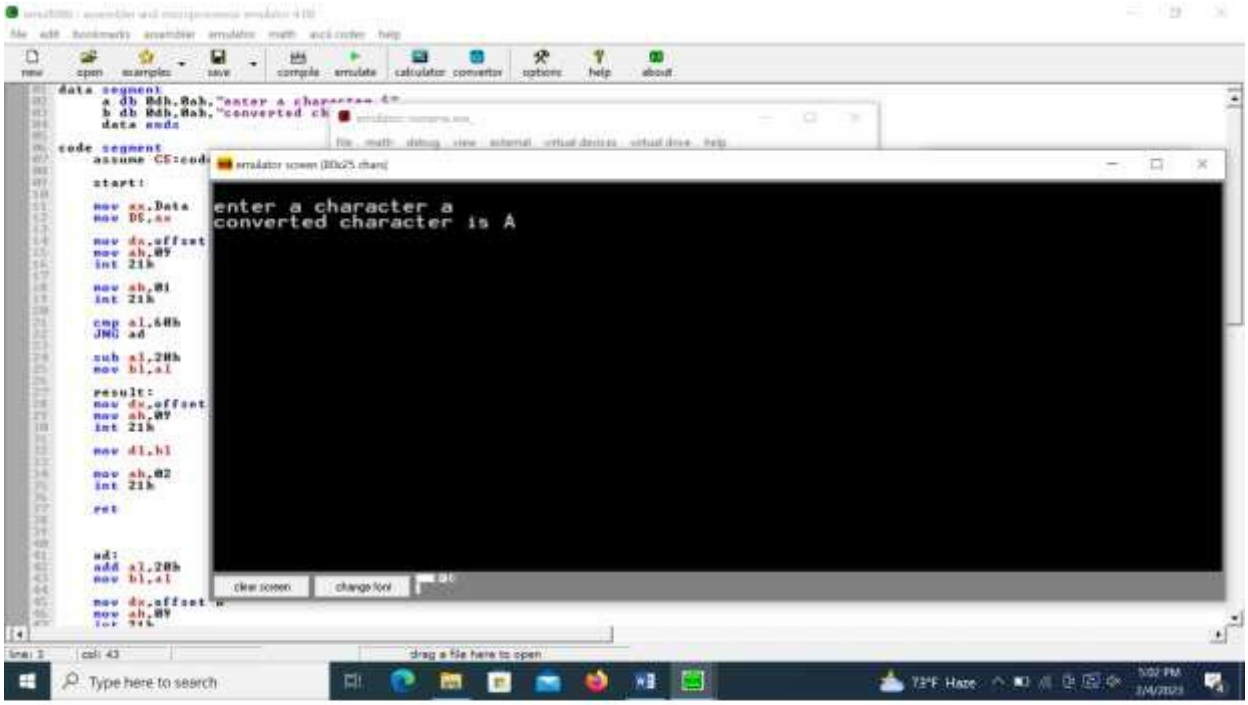
Source Code:

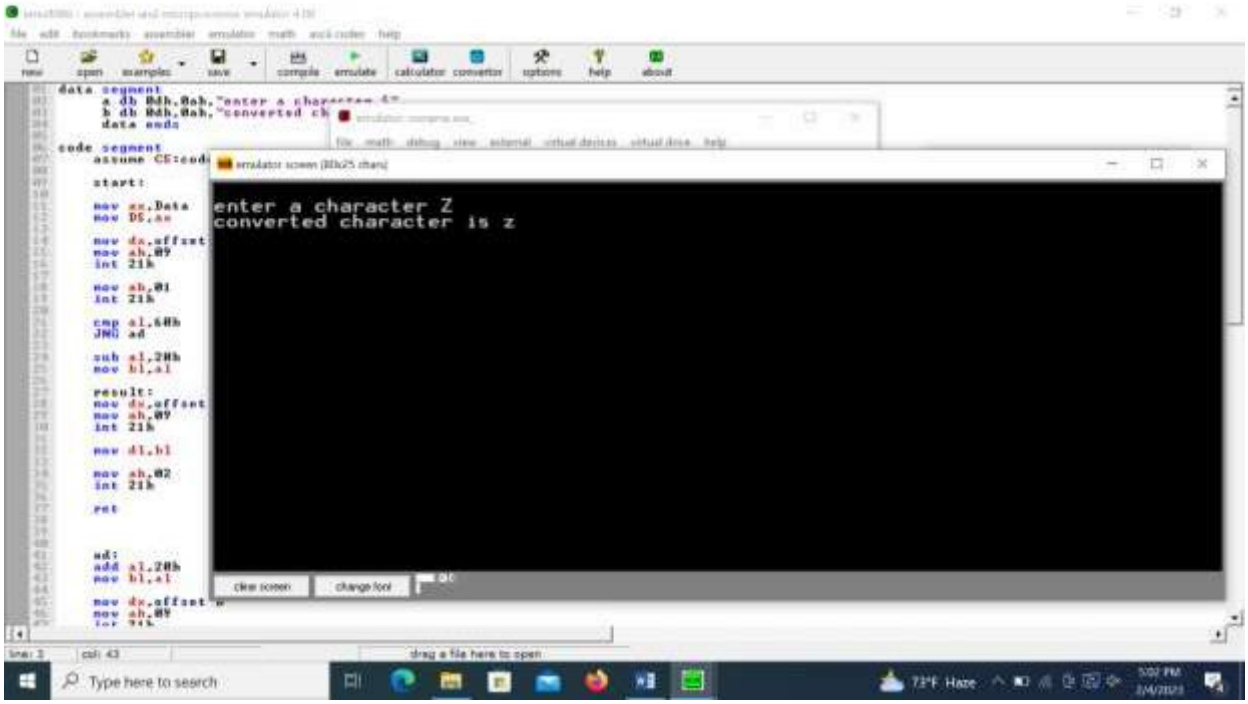
```
data segment
    a db 0dh,0ah,"enter a character $"
    b db 0dh,0ah,"converted character is $"
    data ends

code segment
   assume CS:code,DS:Data

   start:

   mov ax,Data
   mov DS,ax

   mov dx,offset a
   mov ah,09
   int 21h

   mov ah,01
   int 21h

   cmp al,60h
   JNG capital

   small:
   sub al,20h
   mov bl,al
```

```
    mov dx,offset b
    mov ah,09
    int 21h

    mov dl,bl
    mov ah,02
    int 21h
    ret



    capital:
    add al,20h
    mov bl,al

    mov dx,offset b
    mov ah,09
    int 21h

    mov dl,bl
    mov ah,02
    int 21h
    ret

code ends
end start
```

Input: firstly take a input in lower case latter a and take another input in upper case latter Z.

Output:





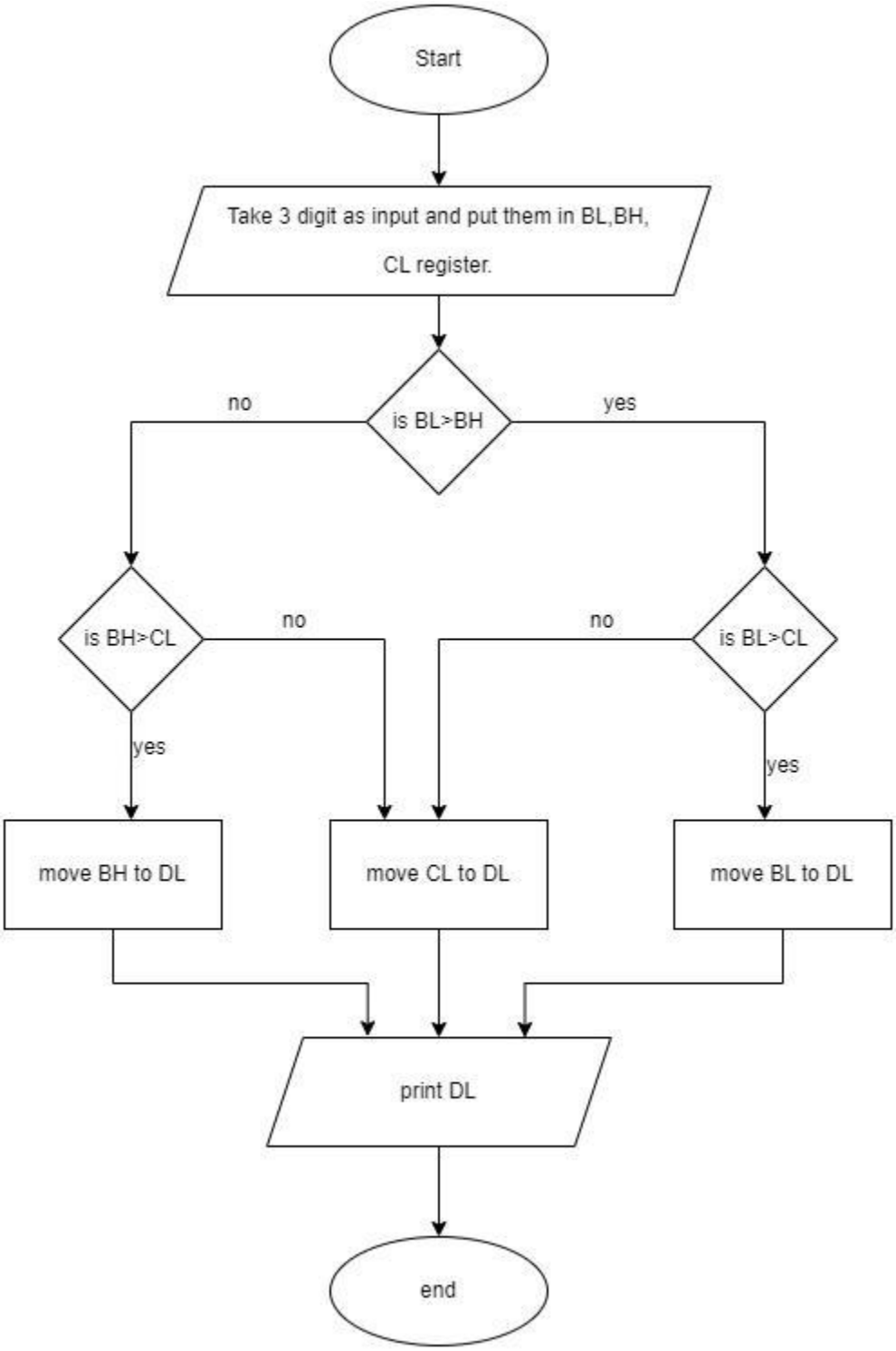
Experiment 2.

Title: Compare three digits and find the biggest number (Using an assembly language program).

Algorithm:

Step1: Start
Step2: Take 3 digit as input and put them in BL,BH,CL register.
Step3: if BL>BH then
Step4:if BL<CL move CL to DL. otherwise move BL to DL.
Step5: otherwise,if BH<CL move CL to DL. otherwise move BH to DL.
Step6: print the content of DL.
Step7: end.

Flow chart:


Start
Take 3 digit as input and put them in BL,BH, CL register.
is BL>BH
no
yes
is BH>CL
no
no
is BL>CL
yes
yes
move BH to DL
move CL to DL
move BL to DL
print DL
end

Source Code:

```
data segment
    a db 0dh,0ah,"enter first number $"
    b db 0dh,0ah,"enter second number $"
    c db 0dh,0ah,"enter third number $"
   d db 0dh,0ah,"biggest number is: $"
   data ends
code segment
   assume CS:code,DS:Data

   start:

   mov ax,Data
   mov DS,ax

   mov dx,offset a
   mov ah,09
   int 21h
   mov ah,01
   int 21h
   mov bl,al
```

```
mov dx,offset b
mov ah,09
int 21h
mov ah,01
int 21h
mov bh,al


mov dx,offset c
mov ah,09
int 21h
mov ah,01
int 21h
mov cl,al


cmp bl,bh
JNG secondbig

firstbig:

cmp bl,cl
JNG thirdbig1
```

```
mov dx,offset d
mov ah,09
int 21h

mov dl,bl
mov ah,02
int 21h
ret


thirdbig1:

mov dx,offset d
mov ah,09
int 21h

mov dl,cl
mov ah,02
int 21h
ret



secondbig:
```

```
cmp bh,cl
JNG thirdbig2
mov cl,bh

mov dx,offset d
mov ah,09
int 21h

mov dl,cl
mov ah,02
int 21h
ret


thirdbig2:
mov dx,offset d
mov ah,09
int 21h

mov dl,cl
mov ah,02
int 21h
ret
```

code ends

end start

Input:

Taking input 3,6,9.

Output: