**Task 1: read two inputs as a number from the user and add them**

; You may customize this and other start-up templates;

; The location of this template is c:\emu8086\inc\0\_com\_template.txt

; add your code here

.model small

.stack 100h

.data

.code

main proc

;read two inputs as a number from the user and add them

mov ah,01

int 21h

mov bl,al

mov ah,01

int 21h

mov ch,al

mov bl,bl

add ch,bl

sub ch,48

mov ah,02

mov dl,ch

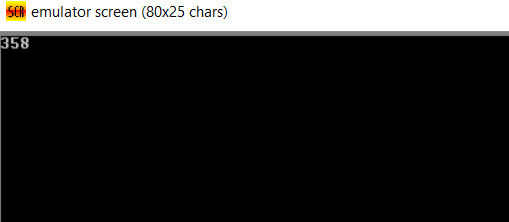
int 21h

mov ah,4ch

int 21h

main endp

end main



**Task 2: display string**

; You may customize this and other start-up templates;

; The location of this template is c:\emu8086\inc\0\_com\_template.txt

; add your code here

.model small

.stack 100h

.data

str db "Sukkur IBA Uniersity$"

str1 db "Ahmed Ali Shah$"

.code

main proc

;display string

mov ax,@data

mov ds,ax

mov ah,09

lea dx,str

int 21h

mov ah,09

lea dx,str1

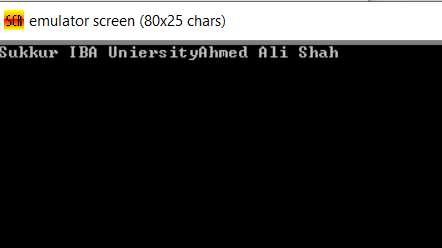
int 21h

mov ah,4ch

int 21h

main endp

end main



**Task 3: prompt user to input two number and then display their sum**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; add your code here**

**.model small**

**.stack 100h**

**.data**

**str db "enter num 1: $"**

**str1 db "enter num 2: $"**

**.code**

**main proc**

**;**

**mov ax,@data**

**mov ds,ax**

**mov ah,09**

**lea dx, str**

**int 21h;**

**mov ah,01**

**int 21h**

**mov bl,al**

**mov ah,02**

**mov dl,0dh**

**int 21h**

**mov dl,0ah**

**int 21h**

**mov ah,09**

**lea dx,str1**

**int 21h**

**mov ah,01**

**int 21h**

**add bl,al**

**mov ah,02**

**mov dl,0dh**

**int 21h**

**mov dl,0ah**

**int 21h**

**sub bl,48**

**mov ah,02**

**mov dl,bl**

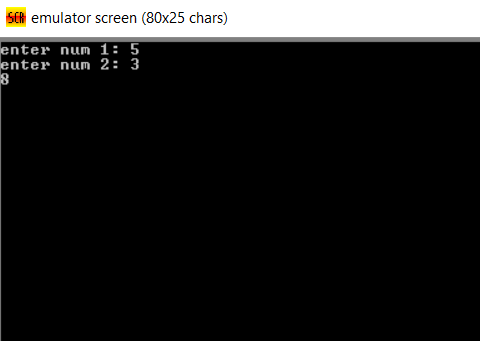
**int 21h**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**



**Task 4: user enter small character and program displays capital character**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; write a code in assembly language to perform the following operations**

**;1. read a character in upper case and display the same character in lower case**

**;2 read a small letter and display in upper case**

**.model small**

**.stack 100h**

**.data**

**.code**

**main proc**

**;**

**mov ax,@data**

**mov ds,ax**

**mov ah,01**

**int 21h**

**mov bl,al**

**add bl,32**

**mov ah,02**

**mov dl,bl**

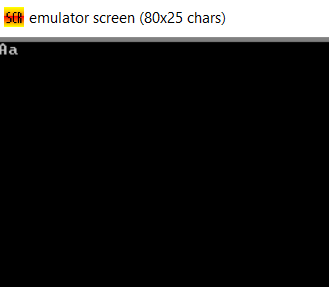
**int 21h**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**



**Task 5: user enters small character and then program displays capital character**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; write a code in assembly language to perform the following operations**

**;1. read a character in upper case and display the same character in lower case**

**;2 read a small letter and display in upper case**

**.model small**

**.stack 100h**

**.data**

**.code**

**main proc**

**;**

**mov ax,@data**

**mov ds,ax**

**mov ah,01**

**int 21h**

**mov bl,al**

**sub bl,32**

**mov ah,02**

**mov dl,bl**

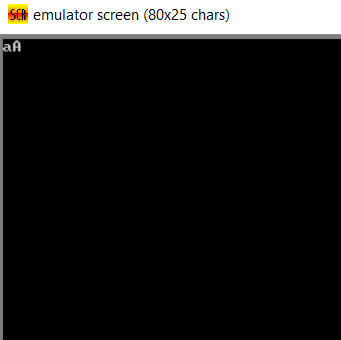
**int 21h**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**



**Task 6: loop**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; write a code in assembly language to perform the following operations**

**;1. read a character in upper case and display the same character in lower case**

**;2 read a small letter and display in upper case**

**.model small**

**.stack 100h**

**.data**

**str1 db "Sukkur IBA$"**

**str2 db "University$"**

**.code**

**main proc**

**;**

**mov ax,@data**

**mov ds,ax**

**mov cx,10**

**label1:**

**mov ah,09**

**lea dx,str1**

**int 21h**

**mov ah,02**

**mov dl,0dh**

**int 21h**

**mov dl,0ah**

**int 21h**

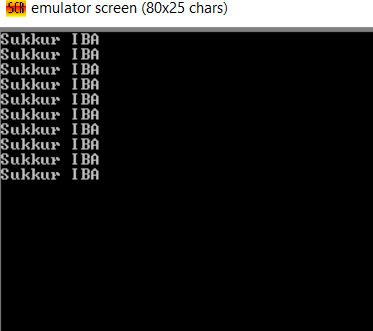
**loop label1**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**



**Task 7 : display a to z using loop**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; write a code in assembly language to perform the following operations**

**;1.display a series of alphabets from a to z and z to a**

**; 2. write a program to display numbers from 0 to 9 and 9 to 0**

**;3. write a code to display the odd number series and display the even series forward and backward both**

**.model small**

**.stack 100h**

**.data**

**str1 db "Sukkur IBA$"**

**str2 db "University$"**

**.code**

**main proc**

**mov ax,@data**

**mov ds,ax**

**mov bl,'a'**

**mov cx,26**

**label1:**

**mov ah,02**

**mov dl,bl**

**int 21h**

**inc bl**

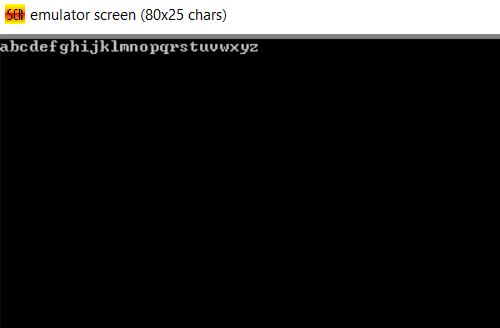
**loop label1**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**



**Task 8: display z to a using loop**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; write a code in assembly language to perform the following operations**

**;1.display a series of alphabets from a to z and z to a**

**; 2. write a program to display numbers from 0 to 9 and 9 to 0**

**;3. write a code to display the odd number series and display the even series forward and backward both**

**.model small**

**.stack 100h**

**.data**

**str1 db "Sukkur IBA$"**

**str2 db "University$"**

**.code**

**main proc**

**mov ax,@data**

**mov ds,ax**

**mov bl,'z'**

**mov cx,26**

**label1:**

**mov ah,02**

**mov dl,bl**

**int 21h**

**dec bl**

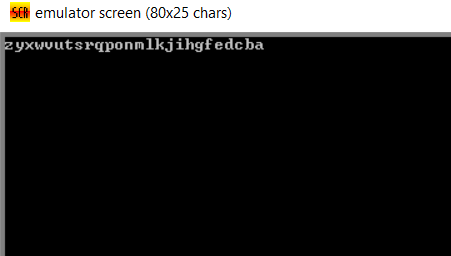
**loop label1**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**



**Task 9 : display 0 to 9 using loop**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; write a code in assembly language to perform the following operations**

**;1.display a series of alphabets from a to z and z to a**

**; 2. write a program to display numbers from 0 to 9 and 9 to 0**

**;3. write a code to display the odd number series and display the even series forward and backward both**

**.model small**

**.stack 100h**

**.data**

**str1 db "Sukkur IBA$"**

**str2 db "University$"**

**.code**

**main proc**

**mov ax,@data**

**mov ds,ax**

**mov bl,'0'**

**mov cx,10**

**label1:**

**mov ah,02**

**mov dl,bl**

**int 21h**

**inc bl**

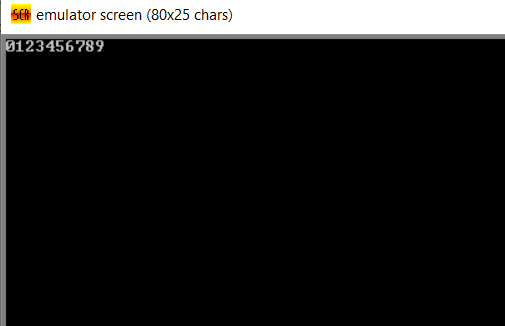
**loop label1**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**



**Task 10: display 9 to 0 using loop**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; write a code in assembly language to perform the following operations**

**;1.display a series of alphabets from a to z and z to a**

**; 2. write a program to display numbers from 0 to 9 and 9 to 0**

**;3. write a code to display the odd number series and display the even series forward and backward both**

**.model small**

**.stack 100h**

**.data**

**str1 db "Sukkur IBA$"**

**str2 db "University$"**

**.code**

**main proc**

**mov ax,@data**

**mov ds,ax**

**mov bl,'9'**

**mov cx,10**

**label1:**

**mov ah,02**

**mov dl,bl**

**int 21h**

**dec bl**

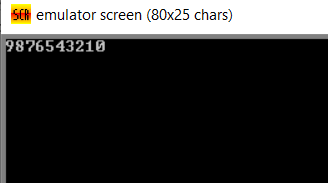
**loop label1**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**



**Task 11: display 1 to 9 odd using loop**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**; write a code in assembly language to perform the following operations**

**;1.display a series of alphabets from a to z and z to a**

**; 2. write a program to display numbers from 0 to 9 and 9 to 0**

**;3. write a code to display the odd number series and display the even series forward and backward both**

**.model small**

**.stack 100h**

**.data**

**str1 db "Sukkur IBA$"**

**str2 db "University$"**

**.code**

**main proc**

**mov ax,@data**

**mov ds,ax**

**mov bl,'1'**

**mov cx,5**

**label1:**

**mov ah,02**

**mov dl,bl**

**int 21h**

**add bl,2**

**loop label1**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**

