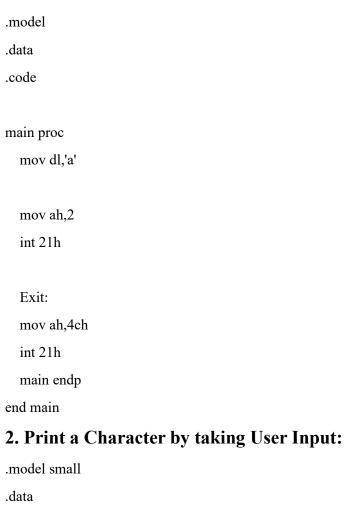
Ahmed Ur Razzak CSE 57th

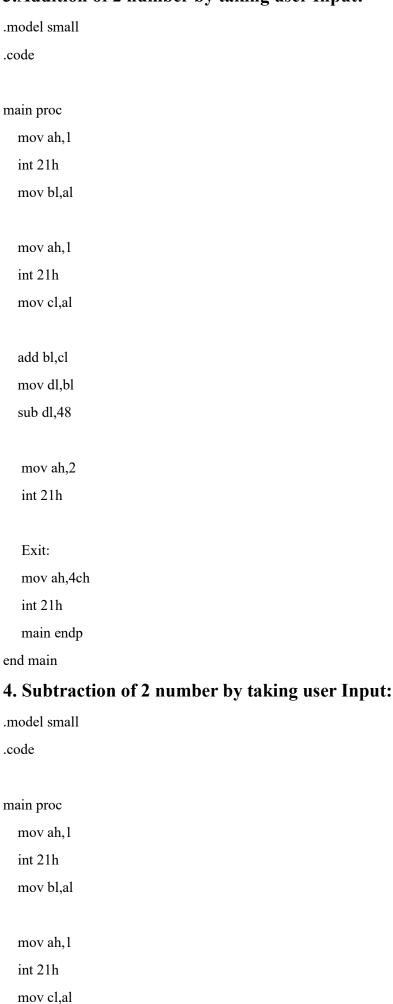
1.Print a Character:



```
.code
main proc
  mov ah,1
  int 21h
  mov dl,al
 mov ah,2
  int 21h
  Exit:
  mov ah,4ch
  int 21h
  main endp
```

end main

3.Addition of 2 number by taking user Input:



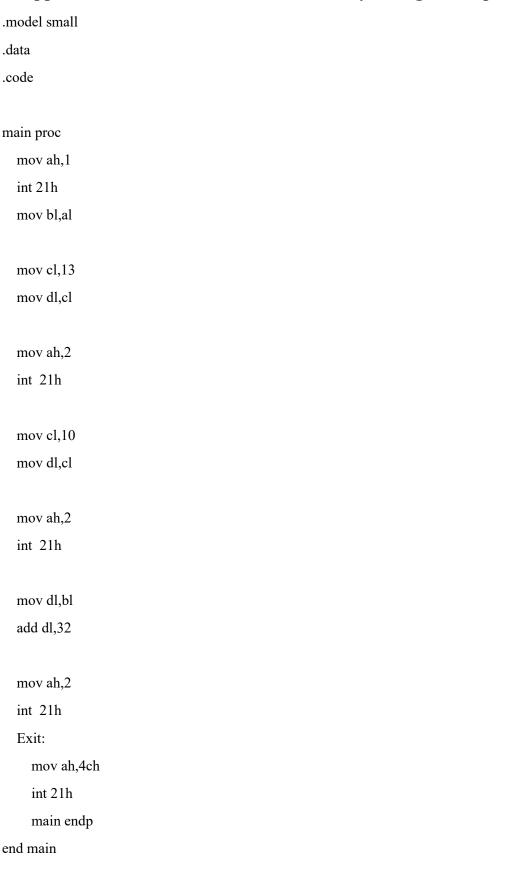
```
sub bl,cl
  mov dl,bl
  add dl,48
  mov ah,2
  int 21h
  Exit:
  mov ah,4ch
  int 21h
  main endp
end main
5. Show the output in a new line after taking user Input:
.model small
.data
.code
main proc
  mov ah,1
  int 21h
 mov dl,10
 mov ah,2
 int 21h
 mov dl,13
 mov ah,2
 int 21h
  mov dl,al
  mov ah,2
  int 21h
  Exit:
  mov ah,4ch
```

```
int 21h
main endp
end main
```

6.Lower Case to Upper Case with new line by taking user input:

```
.model small
.data
.code
main proc
  mov ah,1
  int 21h
  mov bl,al
  mov cl,13
  mov dl,cl
  mov ah,2
  int 21h
  mov cl,10
  mov dl,cl
  mov ah,2
  int 21h
  mov dl,bl
  sub dl,32
  mov ah,2
  int 21h
  Exit:
    mov ah,4ch
    int 21h
    main endp
end main
```

7. Upper Case to Lower Case with new line by taking user input:



8. Variable & String

-Take 2 user input by prompting message & do addition.

Example> I/P: Enter first number:

Enter second number:

O/P: Result:

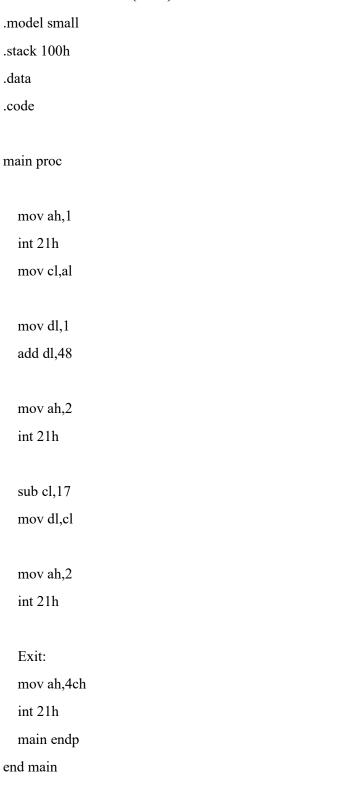
```
.model
.data
var1 db?
var2 db?
smg1 db 'Enter first number: $'
smg2 db 'Enter second number: $'
smg3 db 'Result: $'
.code
main proc
  mov ax,@data
  mov ds,ax
lea dx,smg1
  mov ah,9
  int 21h
mov ah,1
int 21h
mov var1,al
mov dl,10
mov ah,2
int 21h
mov dl,13
```

mov ah,2

int 21h

```
lea dx,smg2
  mov ah,9
  int 21h
mov ah,1
int 21h
mov var2,al
mov dl,10
mov ah,2
int 21h
mov dl,13
mov ah,2
int 21h
lea dx,smg3
mov ah,9
int 21h
mov dl,var1
add dl,var2
sub dl,48
mov ah,2
int 21h
Exit:
  mov ah,4ch
  int 21h
  main endp
end main
```

9. Hexa-Decimal (A-F) to Decimal Conversion:



10. Loop: .model small .code main proc mov cl,10 L1: mov dl,'a' mov ah,2 int 21h loop L1 Exit: mov ah,4ch int 21h main endp end main 11. Find the Largest number between 2 or 3 Digits: .model small

```
.code
main proc
  mov ah, 1
  int 21h
  mov bl,al
  mov ah, 1
  int 21h
  mov bh,al
  mov ah, 1
```

int 21h

mov cl,al

cmp bl,bh

```
cmp bl,cl
 jng L2
  mov dl,bl
  mov ah,2
  int 21h
  jmp exit
  L1:
  cmp bh,cl
 jng L3
  mov dl,bh
 mov ah,2
  int 21h
 jmp exit
  L2:
  mov dl,cl
 mov ah,2
  int 21h
 jmp exit
 L3:
  mov dl,cl
 mov ah,2
  int 21h
 jmp exit
 Exit:
  mov ah,4ch
  int 21h
  main endp
end main
```

jng L1