1. **Bit Addition:**

Data segment

msg db 0dh,0ah,"Enter first number: $"

msg1 db 0dh,0ah,"Enter second number: $"

result db 0dh,0ah,"The Result is: $"

Data ends

Code segment

assume CS:Code,DS:Data

start:

mov ax,Data ; Move Data to Data Segment add8

mov DS,ax

mov dx,offset msg ; Display contents of variable msg

mov ah,09h

int 21h

mov ah,01h ; To accept input and store ASCII value into al

int 21h

sub al,30h ; Accept 10's place of the Number

mov bl,al

rol bl,4

mov ah,01h ; To accept input and store ASCII value into al

int 21h

sub al,30h ; Accept unit's place of Number

add bl,al ; Get the number by adding 10's and unit's place

mov dx,offset msg1 ; Display contents of variable msg1

mov ah,09h

int 21h

mov ah,01h ; To accept input and store ASCII value into al

int 21h

sub al,30h ; Accept 10's place of the Number

mov cl,al

rol cl,4

mov ah,01h ; To accept input and store ASCII value into al

int 21h

sub al,30h ; Accept unit's place of Number

add cl,al ; Get the number by adding 10's and unit's place

add bl,cl ; Add the two accepted Number's

mov dx,offset result ; Display contents of string result

mov ah,09h

int 21h

mov cl,bl ; Store the value of the Result

and bl,0f0h ; Isolate 10's place of Result

ror bl,4

call AsciiConv ; Convert to ASCII to display

mov dl,bl ; Display a Number/Alphabet

mov ah,02h

int 21h

mov bl,cl ; Retrieve original Result

and bl,0fh ; Isolate unit's place of Result

call AsciiConv ; Convert to ASCII to display

mov dl,bl ; Display a Number/Alphabet

mov ah,02h

int 21h

mov ah,4ch ; Terminate the program

int 21h

AsciiConv proc ; Compare to 0a if it is less than A then we need to add only 30

cmp bl,0ah ; If it is greater than or equal to 0a then we also need to add 07

jc skip

add bl,07h

skip: add bl,30h

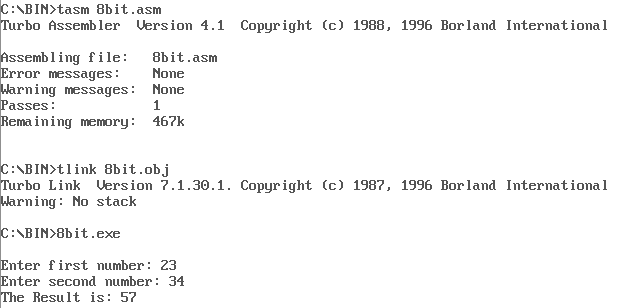
ret

endp

Code ends

end start

**OUTPUT:**



**8 Bit Subtraction:**

Data segment

msg db 0dh,0ah,"Enter first number: $"

msg1 db 0dh,0ah,"Enter second number: $"

result db 0dh,0ah,"The Result is: $"

Data ends

Code segment

assume CS:Code,DS:Data

start:

mov ax,Data ; Move Data to Data Segment

mov DS,ax

mov dx,offset msg ; Display contents of variable msg sub8

mov ah,09h

int 21h

mov ah,01h ; To accept input and store ASCII value into al

int 21h

sub al,30h ; Accept 10's place of the Number

mov bl,al

rol bl,4

mov ah,01h ; To accept input and store ASCII value into al

int 21h

sub al,30h ; Accept unit's place of Number

add bl,al ; Get the number by adding 10's and unit's place

mov dx,offset msg1 ; Display contents of variable msg1

mov ah,09h

int 21h

mov ah,01h ; To accept input and store ASCII value into al

int 21h

sub al,30h ; Accept 10's place of the Number

mov cl,al

rol cl,4

mov ah,01h ; To accept input and store ASCII value into al

int 21h

sub al,30h ; Accept unit's place of Number

add cl,al ; Get the number by adding 10's and unit's place

sub bl,cl ; Subtract the two accepted Number's

mov dx,offset result ; Display contents of string result

mov ah,09h

int 21h

mov cl,bl ; Store the value of the Result

and bl,0f0h ; Isolate 10's place of Result

ror bl,4

call AsciiConv ; Convert to ASCII to display

mov dl,bl ; Display a Number/Alphabet

mov ah,02h

int 21h

mov bl,cl ; Retrieve original Result

and bl,0fh ; Isolate unit's place of Result

call AsciiConv ; Convert to ASCII to display

mov dl,bl ; Display a Number/Alphabet

mov ah,02h

int 21h

mov ah,4ch ; Terminate the program

int 21h

AsciiConv proc ; Compare to 0a if it is less than A then we need to add only 30

cmp bl,0ah ; If it is greater than or equal to 0a then we also need to add 07

jc skip

add bl,07h

skip: add bl,30h

ret

endp

Code ends

end start

**OUTPUT:**

