.model small

.stack

.data

num db 00h

sum dw 00h

result db 5 dup(0)

disp macro m1

lea dx,m1

mov ah,09H

int 21H

endm

msgm db 10,13,'Menu'

db 10,13,' Enter 1 for BCD to HEX conversion'

db 10,13,' Enter 2 for HEX to BCD conversion'

db 10,13,' Enter 3 to exit'

db 10,13,' Enter your choice : $'

msg1 db 10,13,'Enter BCD no.:$'

msg2 db 10,13,'The converted HEX no.is:$'

err\_msg db 10,13,'Invalid Input.$'

msg3 db 10,13, "Enter HEX no : $"

msg4 db 10,13, "The converted BCD no. : $"

.code

mov ax,@data ;initialisation of data segment

mov ds,ax

again: disp msgm

mov ah, 01h

int 21h

cmp al,31h

je b2h

cmp al,32h

je h2b

cmp al,33h

je exit

b2h: call bcdtohex

call displayhex

mov sum,00h

mov num,00h

jmp again

h2b: call ascii2hex ;call of procedure ascii2bcd

call hex2bcd ;call of procedure bcd2hex

mov sum,00h

mov num,00h

jmp again

exit: mov ah,4ch ;return to DOS

int 21h

bcdtohex proc near

disp msg1

upi:mov AH,01H

int 21H

cmp AL,0DH

je w1

cmp AL,39H

jbe w2

disp err\_msg

disp msg1

jmp upi

w2:sub AL,30H

mov num,AL

mov AX,sum

mov BX,000AH

mul bx

add AL,num

mov sum,AX

jmp upi

w1:ret

endp

displayhex proc near

disp msg2

mov bx,sum ;store sum in bx

mov ch,04h

mov cl,04h

up1:ROL bx,cl ;label1

mov dl,bl

AND dl,0fh

cmp dl,09h

jbe dw1

add dl,07h

dw1:add dl,30h

mov ah,02h

int 21h

dec ch

jnz up1

ret

endp

display\_no proc near

add dl,30h

cmp dl,39h

jbe next

add dl,07h

next: mov ah,02h ;display the result

int 21h

ret

display\_no endp

ascii2hex proc near

mov ah,09h ;display message

lea dx,msg3

int 21h

next\_dh: mov ah,01h ;read the character

int 21h

cmp al, 0dh ;comparison with enter key

je quit1

cmp al,30h

jb error1

cmp al,39h

jb mask1

cmp al,41h

jb error1

cmp al,46h

ja error1

sub al,07h

mask1: sub al,30h

mov num, al

mov ax, sum

mov bx,10h

mul bx

add al,num

mov sum,ax

jmp next\_dh

error1: mov ah,09h ;display of error message

lea dx,err\_msg

int 21h

jmp again

quit1: ret

ascii2hex endp

hex2bcd proc near

mov ah,09h ;display msg1

lea dx,msg4

int 21h

mov si,00h

mov bx,0ah

mov ax,sum

con: mov dx,00h

div bx

mov result[si],dl

inc si

cmp ax,00h

jne con

next\_hex1:dec si

mov dl,result[si]

call display\_no ;call of procedure display\_number

cmp si,00h

jne next\_hex1

ret

hex2bcd endp ;end of procedure bcd2hex

end

OUTPUT:



