Assignment 10 | MFP
CE-092

Assignment submission for Microprocessor Fundamentals and Programming subject week 10.

nevilparmar24@gmail.com

Task 1:

Convert 8 bit binary number to gray code.

Driver Code:

```
extrn convertToGray:far
data segment
    n db 0cdh
    answer dw ?
data ends


stack_seg segment stack
    dw 20 dup(0)
    top label word
stack_seg ends


code segment
    assume cs:code, ds:data, ss:stack_seg
    mov ax,data
    mov ds,ax
    mov ax,stack_seg
    mov ss,ax
    mov sp,top
```

```
    xor ax,ax
    mov al,n
    call convertToGray
    mov answer,ax
    int 03h
code ends
end
```

External File Code :

```
public convertToGray

code_ext segment
assume cs:code_ext
proc convertToGray far
    mov bl,al
    clc
    rcr al,1
    xor al,bl
    ret
endp convertToGray
code_ext ends
end
```

Output:

Task 2:

Convert temp from Fahrentheit to Celsius.

Driver Code:

```
extrn cel:far
    data segment public
    n db 0cdh
    answer db ?
data ends

stack_seg segment stack
    dw 20 dup(0)
    top label word
stack_seg ends

code segment
assume cs:code, ds:data, ss:stack_seg
    mov ax,data
    mov ds,ax
```
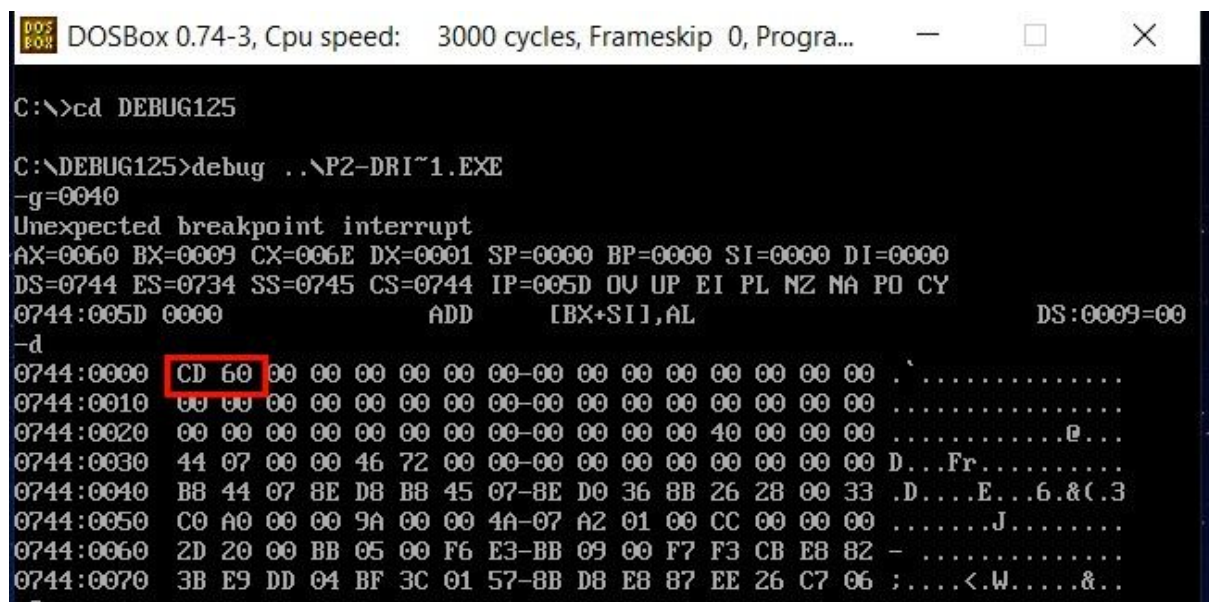
```
        mov ax,stack_seg
        mov ss,ax
        mov sp,top
        xor ax,ax
        mov al,n
        call cel
        mov answer,al
        int 03h
code ends
end
```

External File Code :

```
public cel

code_ext segment
assume cs:code_ext
        proc cel far
        sub ax,32
        mov bx,0005h
        mul bl
        mov bx,0009h
        div bx
        ret
        endp cel
code_ext ends
end
```

Output:

Task 3:

Write a program to find the 2's complement of a 16 bit number.

Driver Code:

```
extrn compliment:far

data segment public
    n dw 0abch
    answer dw ?
data ends

stack_seg segment stack
    dw 20 dup(0)
    top label word
stack_seg ends

code segment
assume cs:code, ds:data, ss:stack_seg
    mov ax,data
    mov ds,ax
```

```asm
        mov ax,stack_seg
        mov ss,ax
        mov sp,top
        xor ax,ax
        mov ax,n
        call compliment
        mov answer,ax
        int 03h
code ends
end
```

External File Code :

```asm
public compliment

code_ext segment
assume cs:code_ext
proc compliment far
        cmp ax,0000h
        jz stop
        not ax
        add ax,0001h
stop:
        ret
endp compliment
code_ext ends
end
```

Output:

Nevil Parmar
CE-092
https://nevilparmar.me