# Somnath Chattaraj
# IT-A3-067 ASM Lab Assignment-1

---

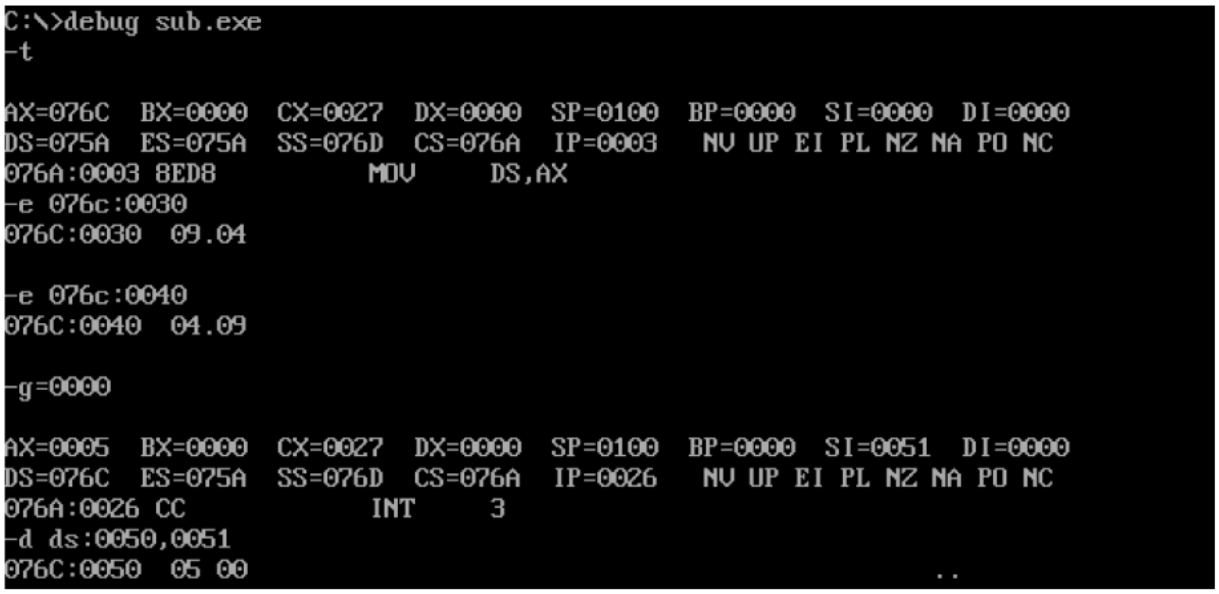
**Q1.) Write an Assembly Language Program to add two sixteen-bit numbers. The numbers are stored in DS: 0030H and DS: 0040H. Store the result in DS: 0050H, DS: 0051H, and DS: 0052H.**

```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov si, 0030h
    mov ax, [si]

    mov si, 0040h
    add ax, [si]
    adc dx, 0

    mov si, 0050h
    mov [si], ax
    inc si
    inc si
    mov [si], dx

    int 03h
main endp
end main
```

```
C:\>debug add16bit.exe
-t

AX=076C  BX=0000  CX=001F  DX=0000  SP=0100  BP=0000  SI=0000  DI=0000
DS=075A  ES=075A  SS=076C  CS=076A  IP=0003   NV UP EI PL NZ NA PO NC
076A:0003 8ED8          MOV     DS,AX
-e 076c:0030
076C:0030  00.05   FF.00

-e 076c:0040
076C:0040  04.04   40.00

-g=0000

AX=0009  BX=0000  CX=001F  DX=0000  SP=0100  BP=0000  SI=0051  DI=0000
DS=076C  ES=075A  SS=076C  CS=076A  IP=001A   NV UP EI PL NZ NA PO NC
076A:001A CC            INT     3
-d ds:0050,0052
076C:0050  09 00 00                                              ...
-
```

**Q2.) Write an Assembly Language Program to subtract an 8-bit number stored in DS: 0030H from a number stored in DS: 0040H using 2's complement method. Store the result in DS: 0050H, and DS: 0051H.**

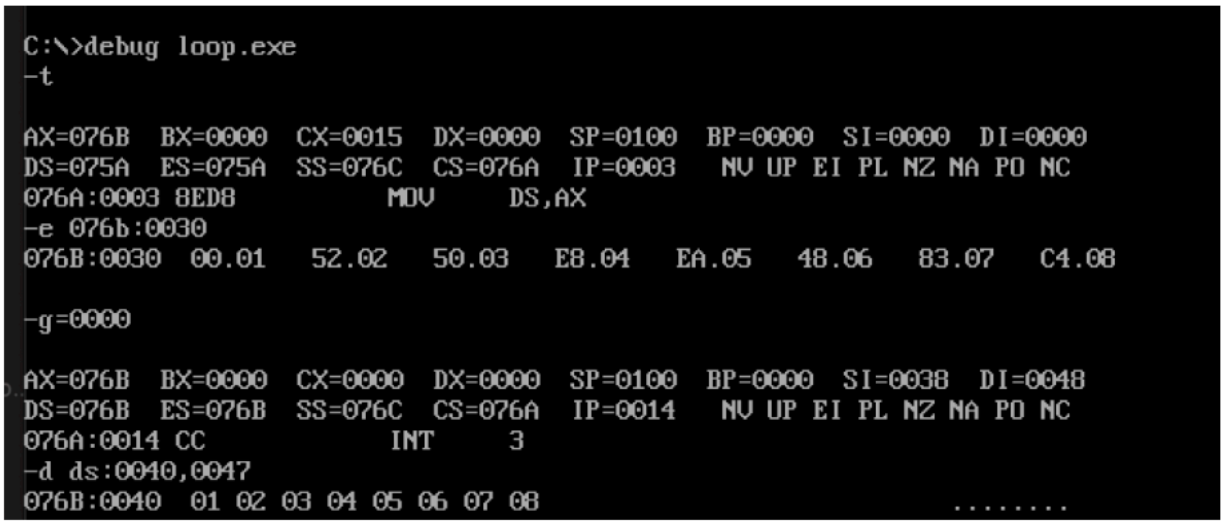
```
..model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov si, 0030h
    mov al, [si]
    not al
    inc al

    mov si, 0040h
    add al, [si]

    jc L1
    not al
    inc al

L1:
    mov si, 0050h
    mov [si], al
    cmc
    mov ah, 00h
    adc ah, ah
    inc si
    mov [si], ah

    int 03h
main endp
end main
```




---

**Q3.) Write a program to transfer a block of 8 data bytes from memory location DS: 0030H to DS: 0040H.**

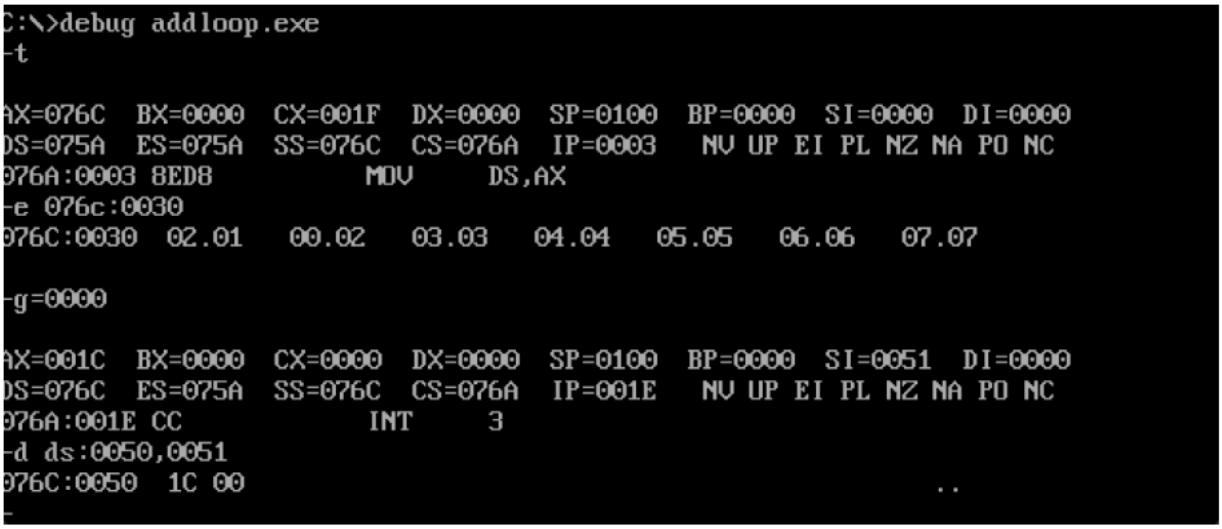
```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax
    mov es, ax

    mov si, 0030h
    mov di, 0040h
    mov cx, 0008h

    cld
l1:
    movsb
    loop l1

    int 03h
main endp
end main
```




---

**Q4.) Write an 8086 Assembly Language Program for the addition of 7 eight-bit numbers stored from DS: 0030H. Store the result in DS: 0050H and DS: 0051H.**

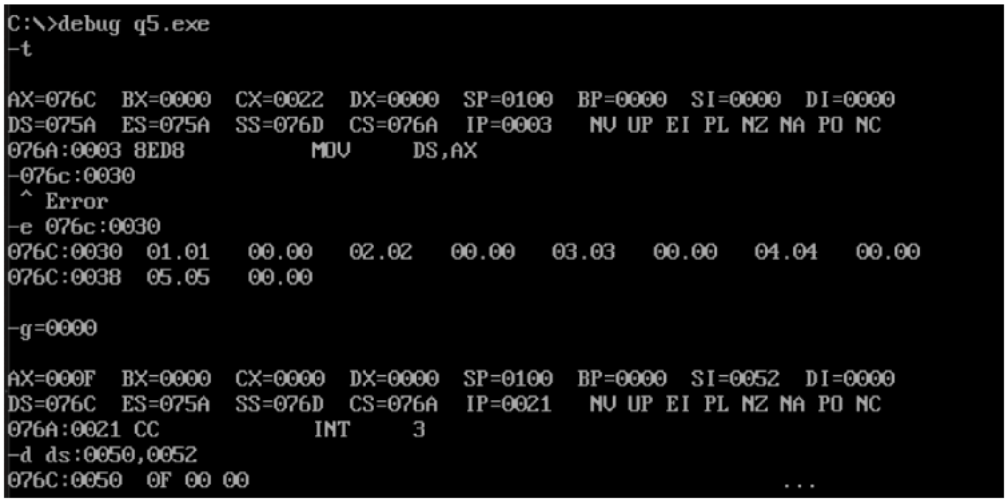
```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov al, 00h
    mov ah, 00h
    mov cl, 07h
    mov si, 0030h

l1:
    add al, [si]
    adc ah, 00h
    inc si
    loop l1

    mov si, 0050h
    mov [si], al
    inc si
    mov [si], ah

    int 03h
main endp
end main
END MAIN
```




---

**Q5.) Write an 8086 Assembly Language Program for the addition of 5 sixteen-bit numbers stored from DS: 0030H. Store the result in DS: 0050H, DS: 0051H, DS: 0052H.**

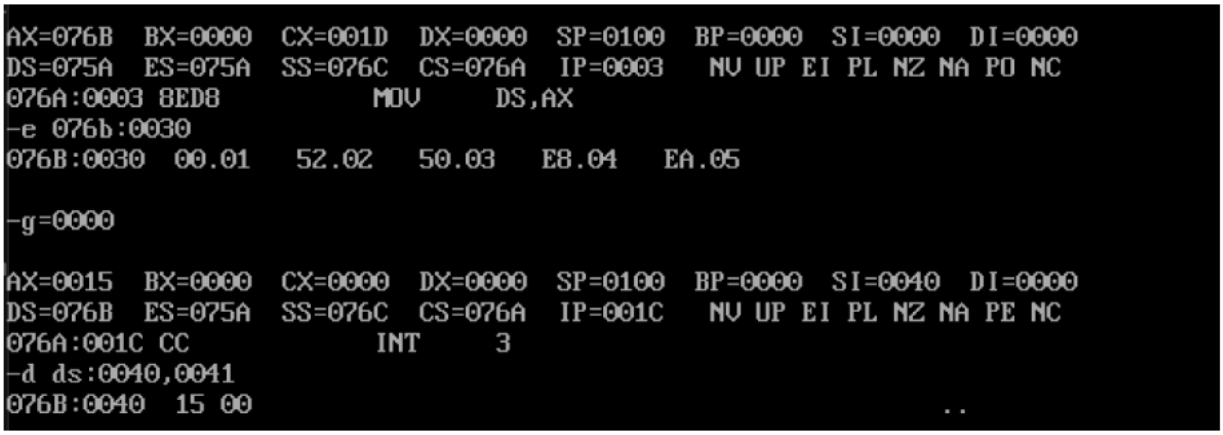
```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov ax, 0000h
    mov bl, 00h
    mov cl, 05h
    mov si, 0030h

l1:
    add ax, [si]
    adc bl, 00h
    inc si
    inc si
    loop l1

    mov si, 0050h
    mov [si], ax
    inc si
    inc si
    mov [si], bl

    int 03h
main endp
end main
```




---

**Q6.) Write an Assembly Language Program for the addition of five BCD numbers stored from DS: 0030H. Store the result in DS: 0040H and DS: 0041H.**

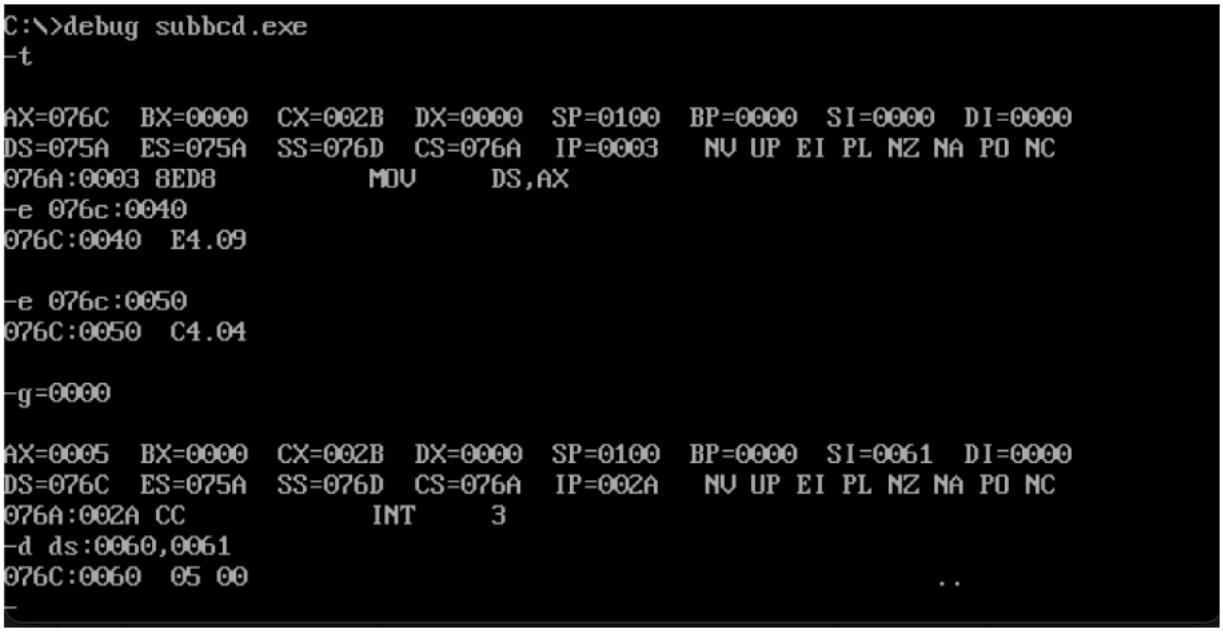
```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov ax, 0000h
    mov cx, 0005h
    mov si, 0030h

l1:
    add al, [si]
    daa
    adc ah, 00h
    inc si
    loop l1

    mov si, 0040h
    mov [si], ax

    int 03h
main endp
end main
```




---

**Q7.) Write an Assembly Language Program to subtract a BCD number stored in DS: 0040H from a BCD number stored in DS: 0050H. Store the result in DS: 0060H and DS: 0061H.**

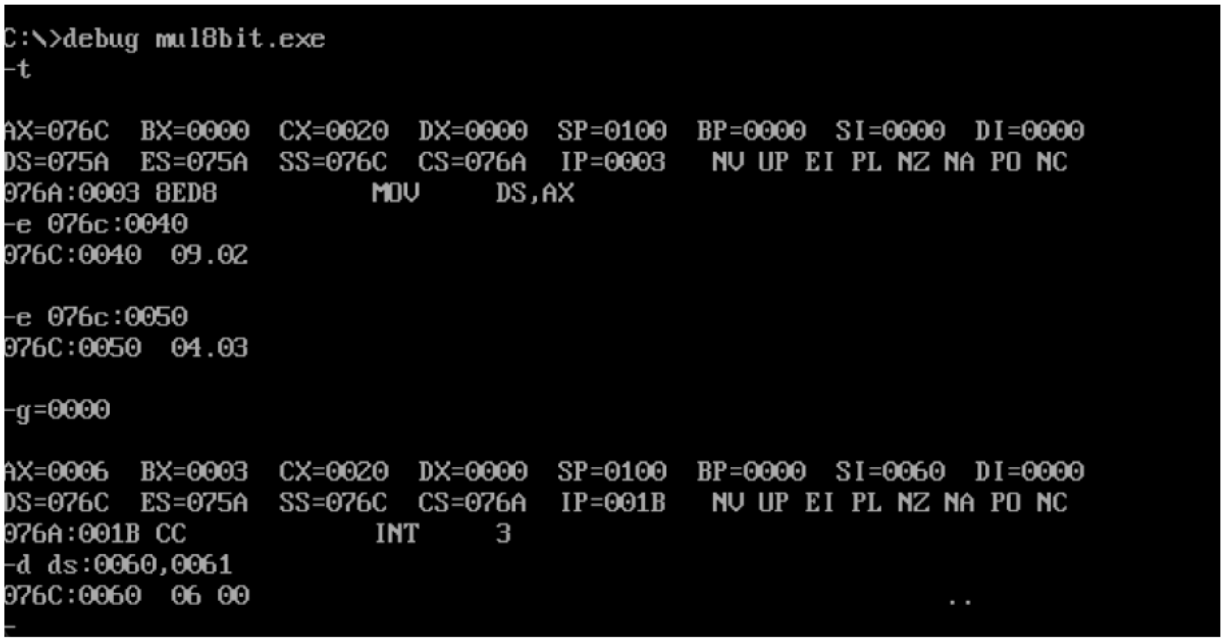
```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov ax, 0000h
    mov si, 0040h
    mov al, [si]

    mov si, 0050h
    sub al, [si]
    das
    mov dl, 00h
    jnc l1

    mov cl, al
    mov al, 99h
    sub al, cl
    add al, 01h
    daa
    mov dl, 01h

l1:
    mov si, 0060h
    mov [si], al
    inc si
    mov [si], dl

    int 03h
main endp
end main
```




---

**Q8.) Write an Assembly Language Program to multiply two eight-bit numbers stored in DS: 0040H and DS: 0050H. Store the result in DS: 0060H.**

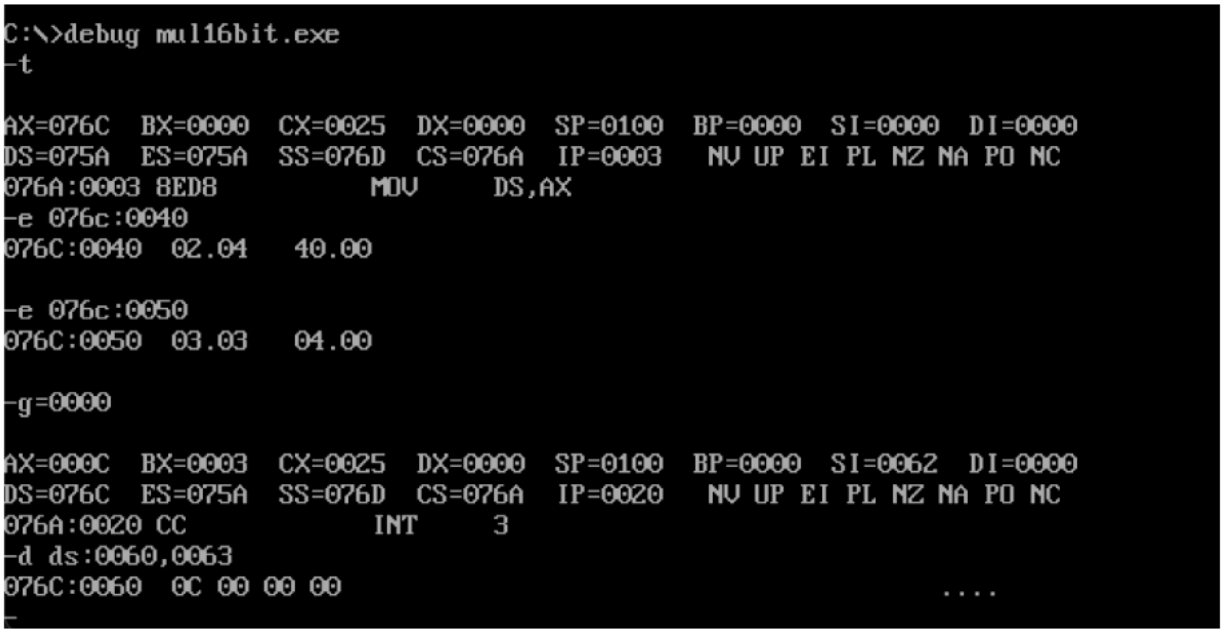
```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov ax, 0000h
    mov bl, 00h
    mov si, 0040h
    mov al, [si]

    mov si, 0050h
    mov bl, [si]
    mul bl

    mov si, 0060h
    mov [si], ax

    int 03h
main endp
end main
```




---

**Q9.) Write an Assembly Language Program to multiply two sixteen-bit numbers stored in DS: 0040H and DS: 0050H. Store the result in DS: 0060H.**

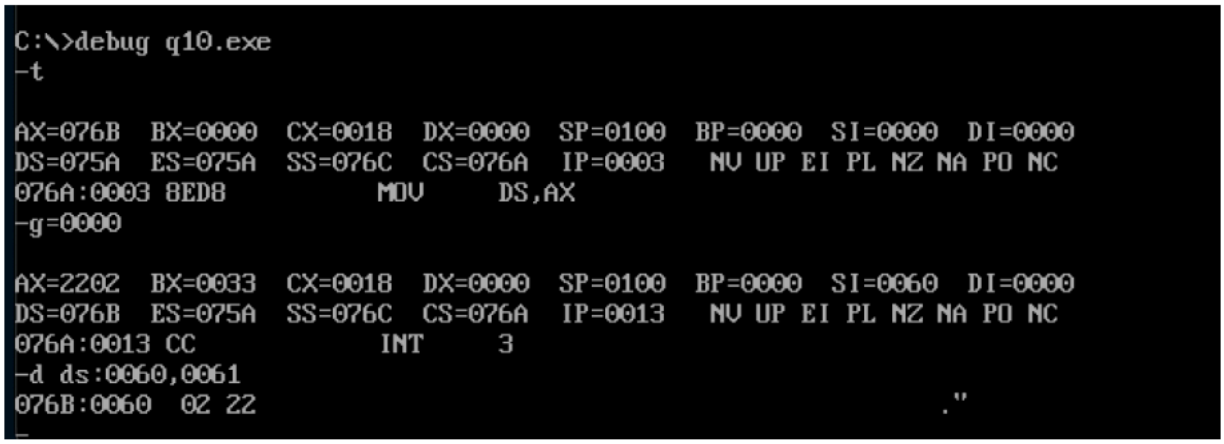
```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov ax, 0000h
    mov bx, 0000h

    mov si, 0040h
    mov ax, [si]
    mov si, 0050h
    mov bx, [si]
    mul bx

    mov si, 0060h
    mov [si], ax
    inc si
    inc si
    mov [si], dx

    int 03h
main endp
end main
```




---

## Q10.) Write an Assembly Language Program to divide 88H by 33H. Store the quotient in DS: 0060H and remainder in DS: 0061H

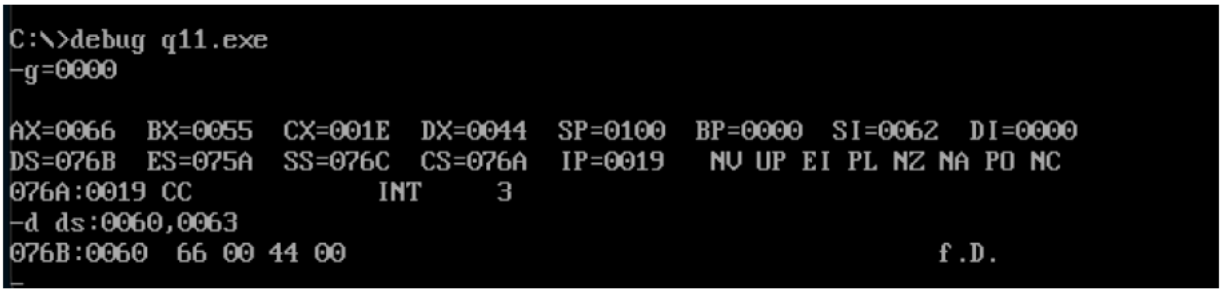
```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov ax, 0000h
    mov al, 88h
    mov bl, 33h
    div bl

    mov si, 0060h
    mov [si], ax

    int 03h
main endp
end main
```




---

**Q11.) Write an Assembly Language Program to divide 2222H by 55H. Store the quotient in DS: 0060H and remainder in DS: 0062H.**

```
.model small
.stack 100h
.data
.code
main proc
    mov ax, @data
    mov ds, ax

    mov dx, 0000h
    mov ax, 2222h
    mov bx, 0055h
    div bx

    mov si, 0060h
    mov [si], ax
    inc si
    inc si
    mov [si], dx

    int 03h
main endp
end main
```


C:\>debug q11.exe
-g=0000
AX=0066 BX=0055 CX=001E DX=0044 SP=0100 BP=0000 SI=0062 DI=0000
DS=076B ES=075A SS=076C CS=076A IP=0019 NV UP EI PL NZ NA PO NC
076A:0019 CC INT 3
-d ds:0060,0063
076B:0060 66 00 44 00 f.D.