Experiment No 2: 16-bit Arithmetic Operations

Date: 28-08-2020

NAME: Gayathri M
REG.NO: 185001050

1. AIM:

Program for adding 2, 16-bit numbers.

ALGORITHM:

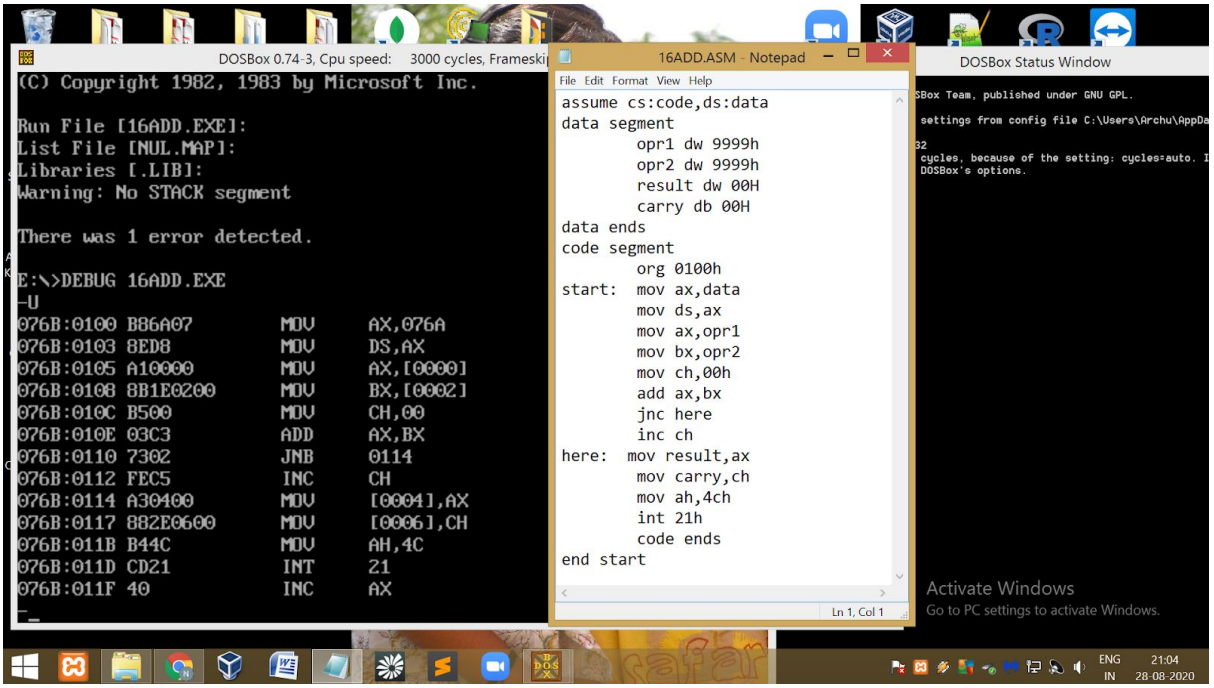
- Initialize the data segment
- Move data segment address to ds
- Load operand-1 to ax and operand-2 to bx
- Load 00h to ch register for carry
- Add ax and bx
- If there is no carry being generated, goto here segment else, increment ch by 1 and go to here segment
- In here segment,
    - Load ax to result
    - Load ch to carry
    - Terminate the program

PROGRAM:

PROGRAM	COMMENTS
Start:	
mov ax,data mov ds,ax	Transferring address of data segment to ds
mov ax,opr1	Value of opr1 is loaded to ax
mov bx,opr2	Value of opr2 is loaded to bx
mov ch,00h	Initializing the value of ch
add ax,bx	ax=ax+bx
jnc here	Jump to “here” segment if no carry is generated
inc ch	Increments ch by 1

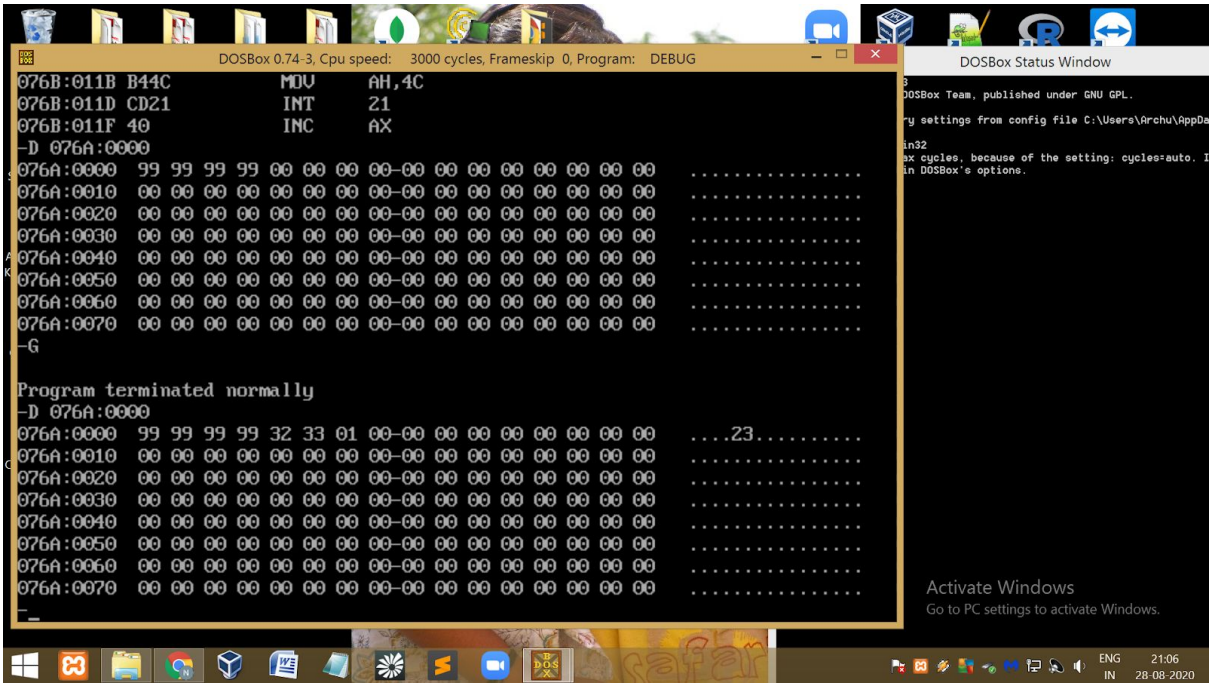
Here:	
mov result,ax	Load register value of ax to result
mov carry,ch	Load ch value to carry
mov ah,4ch int 21h	Termination of execution
code ends	Ending the segment with the segment name

UNASSEMBLED CODE:



SAMPLE INPUT/OUTPUT:

(ax=9999; bx=9999)



RESULT:

The addition of 2, 16-bit numbers is thus shown.

2. AIM:

Program for subtracting 2, 16-bit numbers.

ALGORITHM:

- Initialize the data segment
- Move data segment address to ds
- Load operand-1 to ax and operand-2 to bx
- Load 00h to ch register and subtract ax and bx
- If ax is greater than bx, goto here segment else, increment ch by 1 and find the 2's complement of ah and goto segment here
- In here segment,
    - Load ax to result
    - Load ch to carry
    - Terminate the program

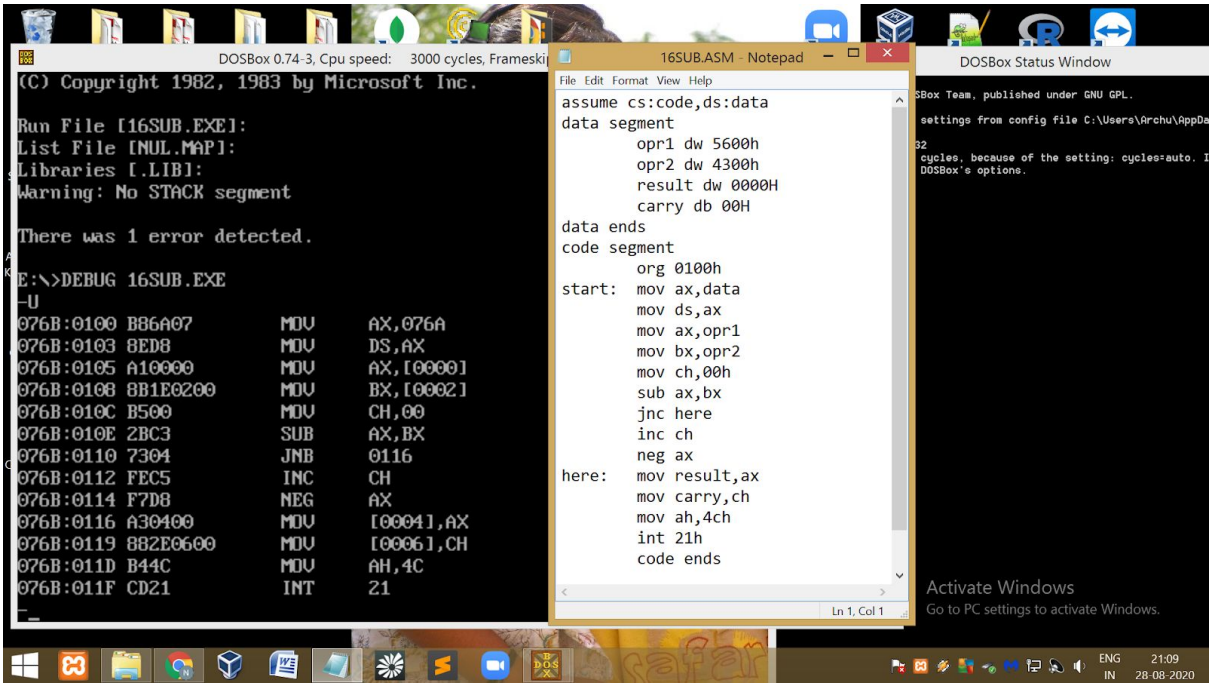
PROGRAM:

PROGRAM	**COMMENTS**

Start:
mov ax,data
mov ds,ax
	Transferring address of data segment to ds

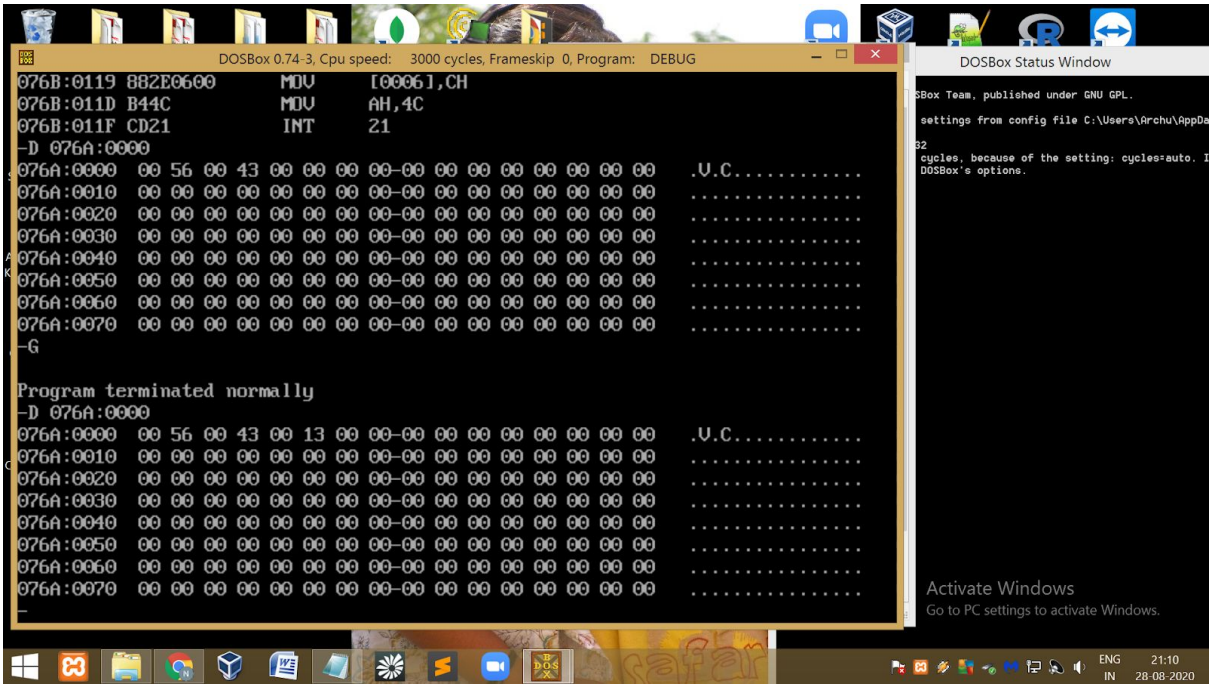
mov ax,opr1	Value of opr1 is loaded to ax
mov bx,opr2	Value of opr2 is loaded to bx
mov ch,00h	Initializing the value of ch
sub ax,bx	ax=ax-bx
jnc here	Jump to "here" segment if ax>bx
inc ch	Increments ch by 1
neg ah	2's complement of ah
Here:	
mov result,ax	Load register value of ax to result
mov carry,ch	Load ch value to carry
mov ah,4ch int 21h	Termination of execution
code ends	Ending the segment with the segment name

UNASSEMBLED CODE:



SAMPLE INPUT/OUTPUT

ax=4300; bx=5600 (ax<bx)



RESULT:

The subtraction of 2, 16-bit numbers is thus shown.

3. AIM:

Program for multiplication of 2, 16-bit numbers.

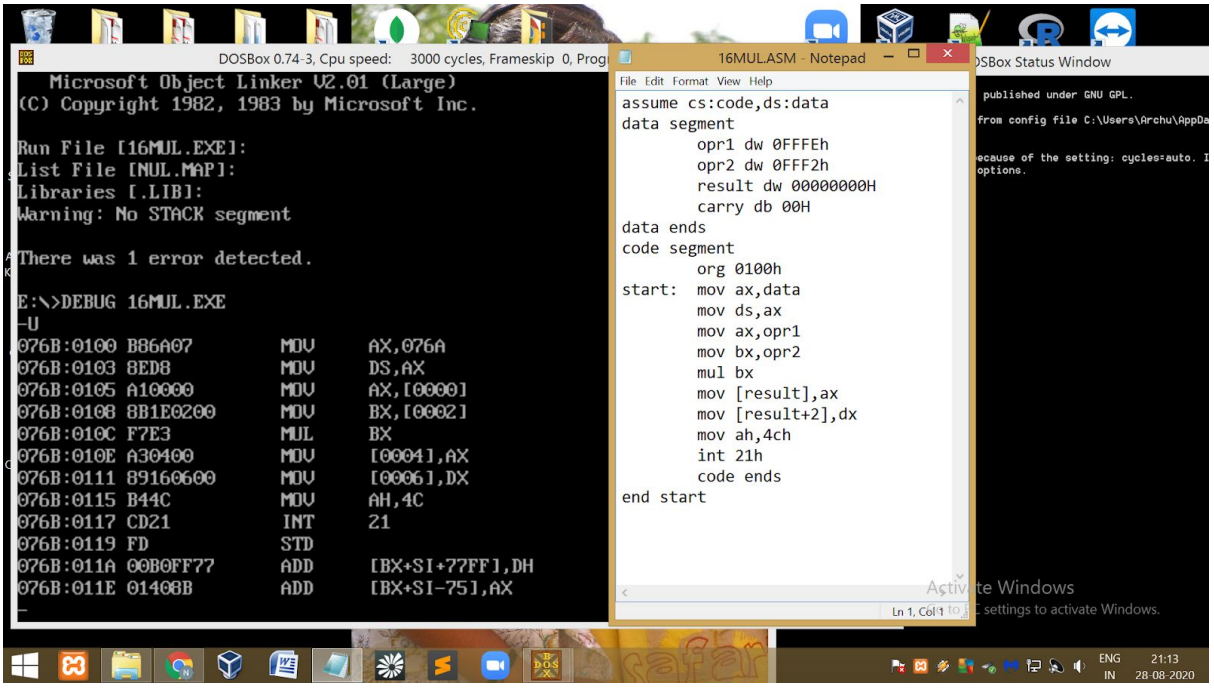
ALGORITHM:

- Initialize the data segment
- Move data segment address to ds
- Load operand-1 to ax and operand-2 to bx
- Multiply bx (dxax=ax x bx)
- Load ax to result
- Load dx to location result+2
- Terminate the program

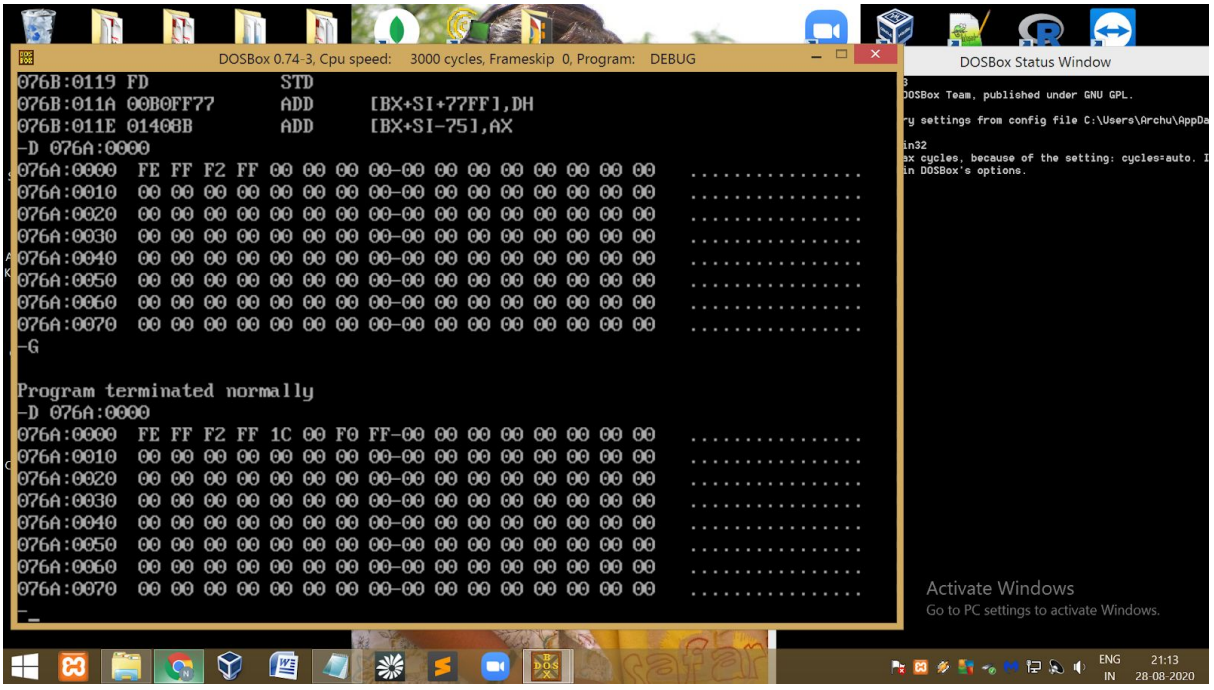
PROGRAM:

PROGRAM	**COMMENTS**
Start:	
mov ax,data	Transferring address of data segment to ds
mov ds,ax	
mov ax,opr1	Value of opr1 is loaded to ax
mov bx,opr2	Value of opr2 is loaded to bx
mul bx	dxax=ax x bx
mov [result],ax	Load register value of ax to result
mov[result+2],dx	Load register value of dx to location [result+2]
mov ah,4ch	
int 21h	Termination of execution
code ends	Ending the segment with the segment name

UNASSEMBLED CODE:



SAMPLE INPUT/OUTPUT (ax=FFFE ; bx=FFF2)



RESULT:

The multiplication of 2, 16-bit numbers is thus shown.

4. AIM:

Program for division of 2, 16-bit numbers.

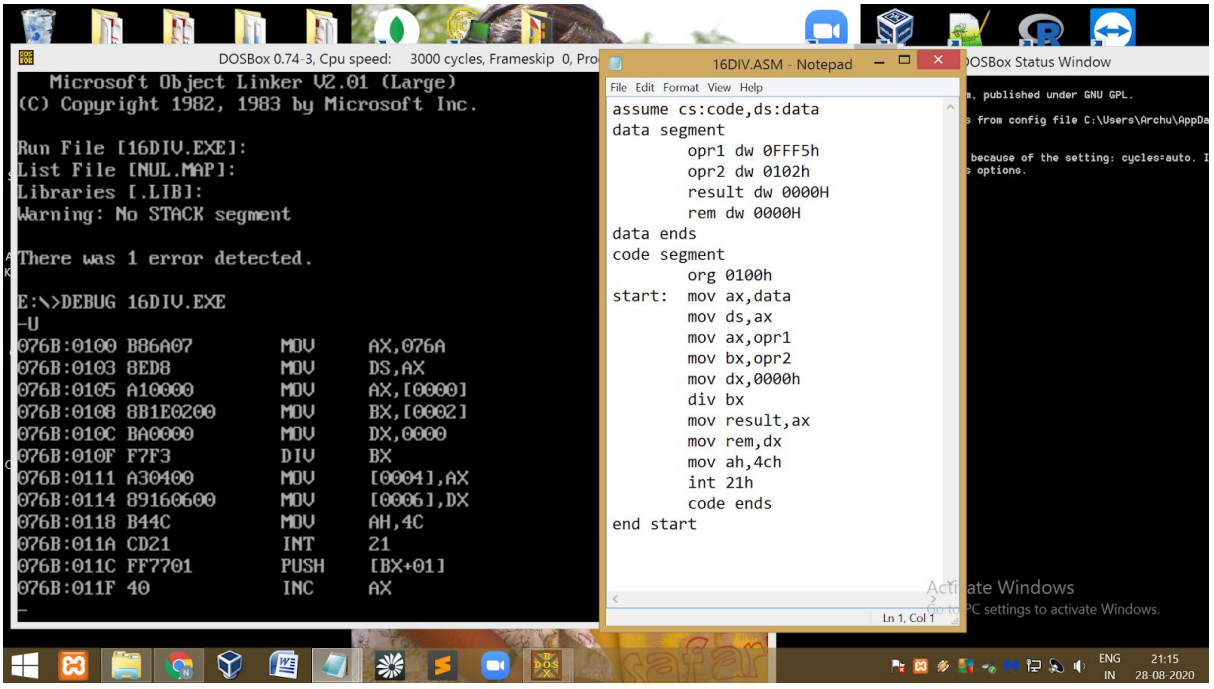
ALGORITHM:

- Initialize the data segment
- Move data segment address to ds
- Load operand-1 to ax and operand-2 to bx
- Load dx with 0000h
- Divide bx (ax = dxax / bx ; remainder in dx)
- Load ax to result
- Load dx to rem (remainder)
- Terminate the program

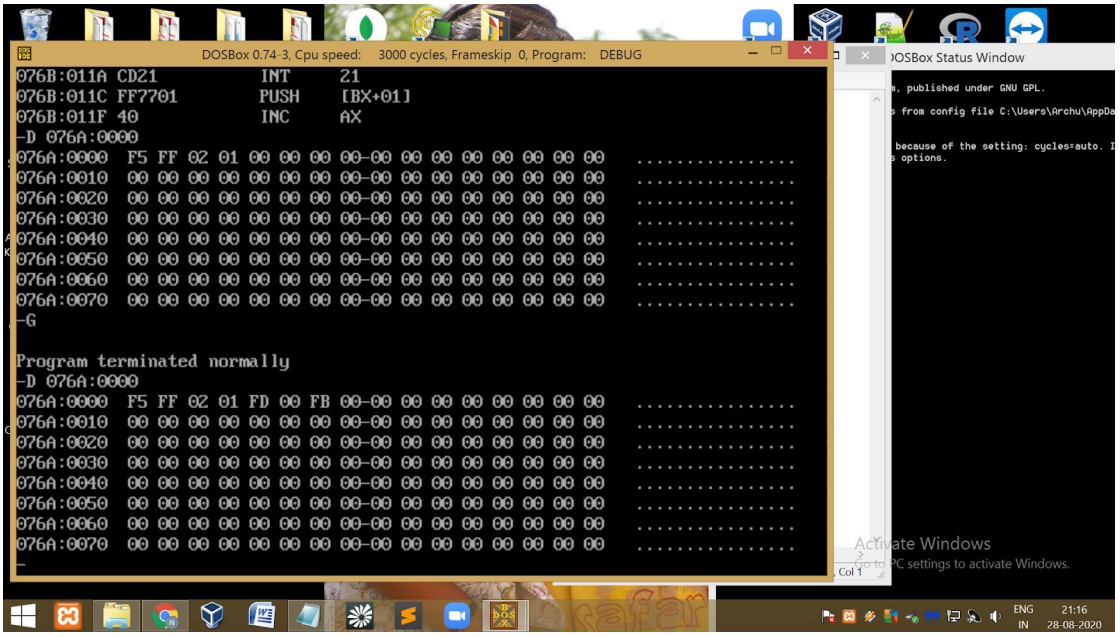
PROGRAM:

PROGRAM	**COMMENTS**
Start:	
mov ax,data mov ds,ax	Transferring address of data segment to ds
mov dx,0000h	Register dx is loaded with 0000
mov ax,opr1	Value of opr1 is loaded to ax
mov bx,opr2	Value of opr2 is loaded to bx
div bx	ax = dxax / bx
mov result,ax	Load register value of ax to result
mov rem,dx	Load register value of dx to rem
mov ah,4ch int 21h	Termination of execution
code ends	Ending the segment with the segment name

UNASSEMBLED CODE:



SAMPLE INPUT/OUTPUT
(ax=FFF5 ; bx=0102)



RESULT:

The division of 2,16-bit numbers is thus shown.