|  |
| --- |
| FAST National University |
| **Introduction to DOSBox**  **Lab 1** |
|  |

**Computer Organization and Assembly Language**

|  |  |
| --- | --- |
| **Student Name** | Moeez Ali |
| **Registration #** | 21L-1798 |
| **Instructor** | Hazoor Ahmad |
| **Class** | CS3 |
| **Section** | A1, A2, D1, D2 |
| **Semester** | Fall 2022 |

Fast School of Computing

FAST-NU, Lahore, Pakistan

# Activity 1

|  |  |  |  |
| --- | --- | --- | --- |
| ZF | CF | SF | OF |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 |

# Activity 2

## **Assembly Language Code**

[org 0x0100]

mov cx,[num1]

mov ax,0

value:

add ax,[num1]

sub cx,1

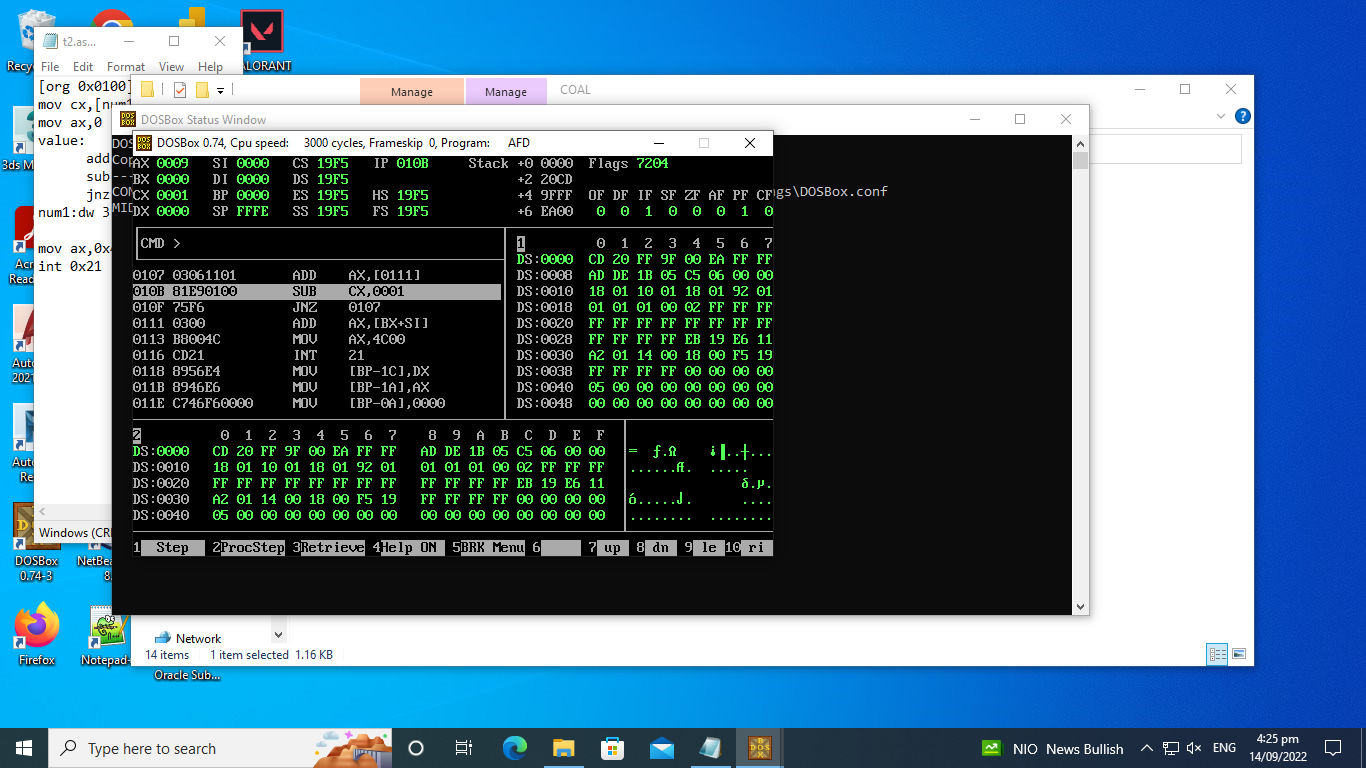
jnz value

num1:dw 3

mov ax,0x4c00

int 0x21

## **Debugging Screenshots**



# Activity 3

## **Assembly Language Code**

[org 0x0100]

mov ax,8

mov cx,0

mov bx,0

l2:cmp ax,[array+bx]

je l1

l1:add cx,1

add bx,2

cmp bx,24

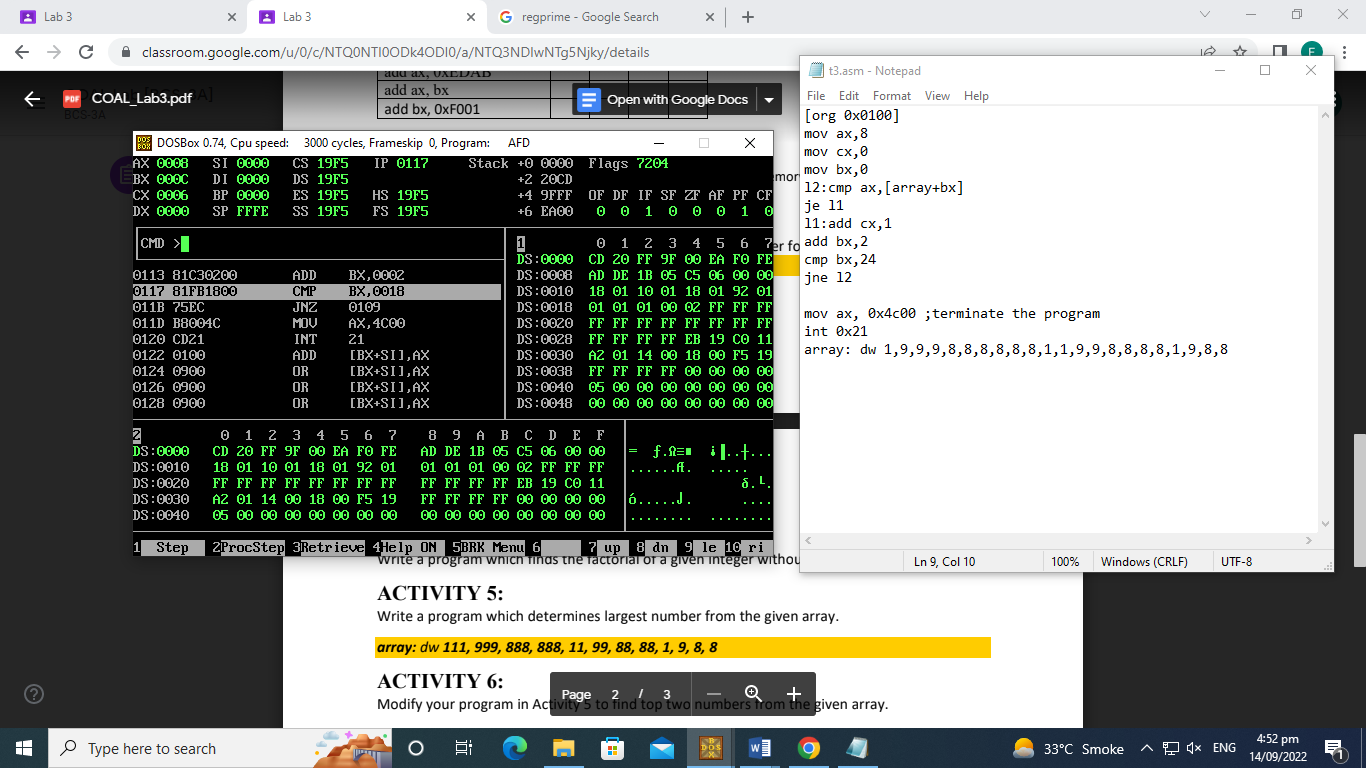
jne l2

mov ax, 0x4c00 ;terminate the program

int 0x21

array: dw 1,9,9,9,8,8,8,8,8,8,1,1,9,9,8,8,8,8,1,9,8,8

## **Debugging Screenshots**



# Activity 4

## **Assembly Language Code**

[[org 0x0100]

mov ax,3

mov bx,ax

l2:mov cx,bx

l1:add [factorial],bx

sub cx,1

cmp cx,0

jne l1

sub bx,1

cmp bx,1

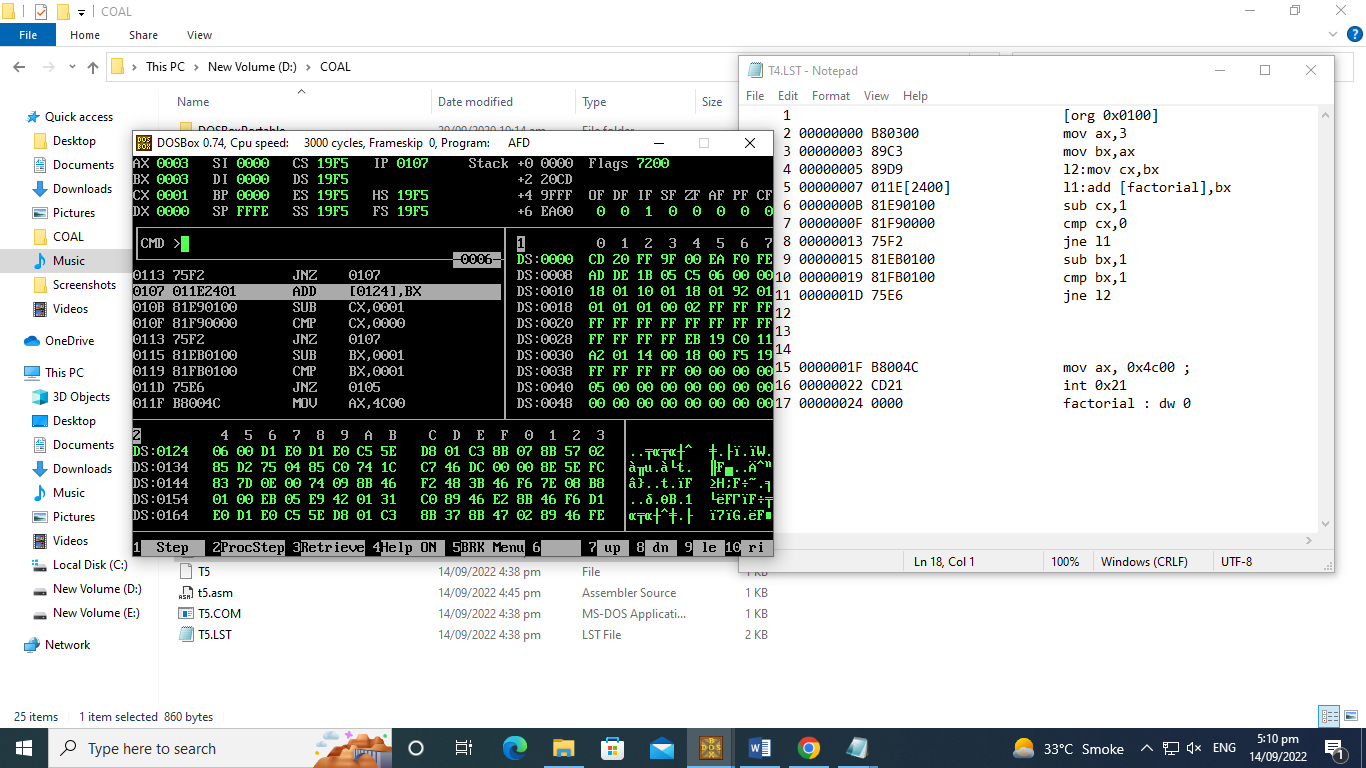
jne l2

mov ax, 0x4c00 ;

int 0x21

factorial : dw 0

## **Debugging Screenshots**



# Activity 5

## **Assembly Language Code**

[org 0x0100]

mov bx, 0 ; initialize array index to zero

mov ax, 0 ; initialize min to zero

mov ax, [array+bx] ; max number to ax

mov cx, 12

maxvalue: cmp ax, [array+bx] ; find the maximun number

jge maxloop ; if greater or equal number

mov ax, [array+bx] ; ax contains the maximun number

maxloop:

add bx, 2 ; advance bx to next index

loop maxvalue

mov [max], ax ; write back maximun in memory

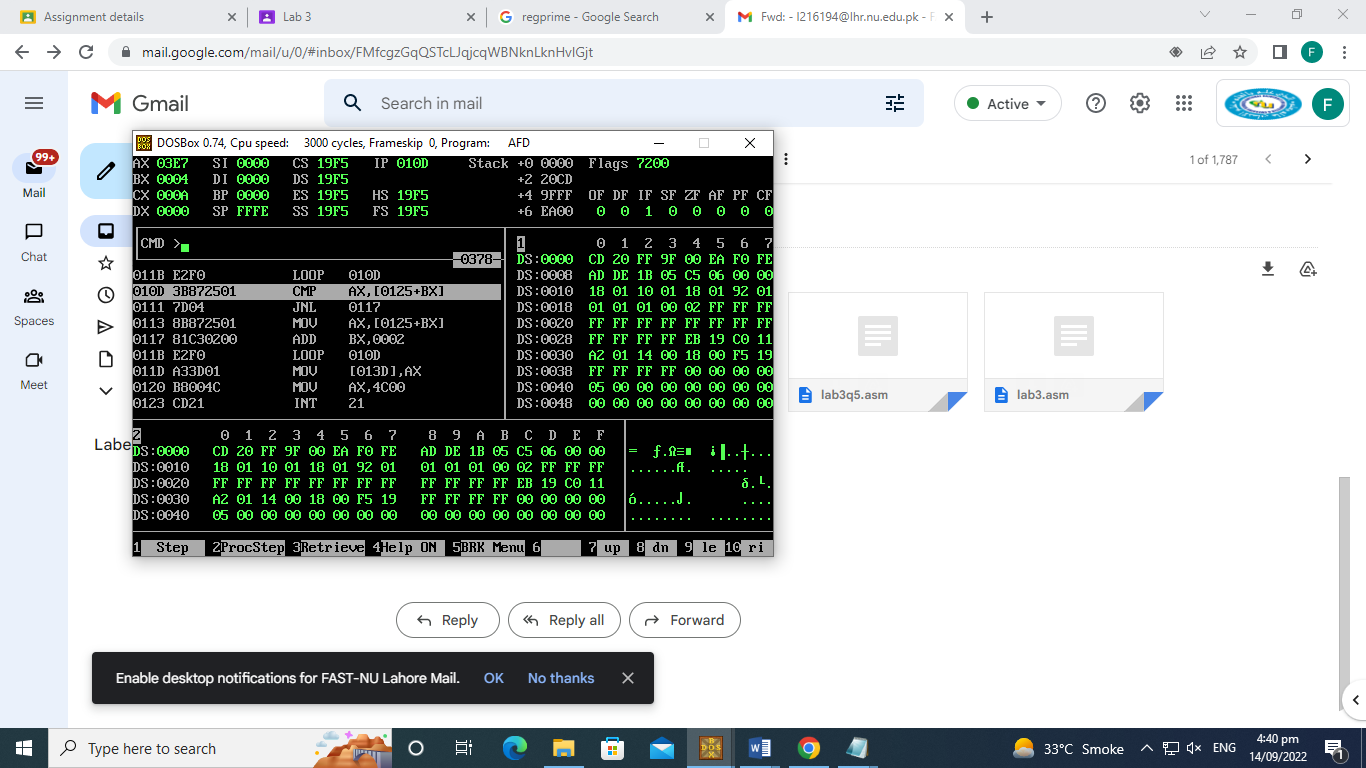
mov ax, 0x4c00 ; terminate program

int 0x21

array: dw 111, 999, 888, 888, 11, 99, 88, 88, 1, 9, 8, 8

max: dw 0

**Debugging Screenshots**



# Activity 6

## **Assembly Language Code**

[[[org 0x0100]

mov ax,[array]

mov bx,[array]

mov si,0

mov cx,[maxsize]

Max:

cmp ax,[array+si]

jnl Secondmax

mov ax,[array+si]

Secondmax:

cmp bx,[array+si]

jnl looop

cmp ax,[array+si]

je looop

mov bx,[array+si]

looop:

add si, 2

sub cx,1

jnz Max

mov word[max],ax

mov word[secondmax],bx

mov ax, 0x4c00 ;terminate quit the program

int 0x21

array: dw 111, 999, 888, 888, 11, 99, 88, 88, 1, 9, 8, 8

maxsize: dw 12

max: dw 0

secondmax: dw 0

## **Debugging Screenshots**

