**Solutions to Lab-05R**

**Computer Organization and Assembly Language**

# Activity 1

Write a subroutine to determine p and m described as follows: If your roll number is 16L-

4195 then p = (4 + 1 + 9 + 5) ≫ 2 = 4, and m = p + 2 = 6.

## **Assembly Language Code**

[org 0x0100]

mov ax,[rollnum]

mov bx,0

mov cx,0

start:

loop1:

add cx,1

and ax,000Fh

add word[sum],ax

mov ax,[rollnum]

mov bx,0

loop2:

add bx,1

shr ax,4

cmp cx,bx

jne loop2

cmp cx,4

jne loop1

mov dx,[sum]h

;shifting right 2 times

shr dx,2

mov[a],dx

mov ax,0

add ax,[a]

add ax,2

mov word[b],ax

mov dx,[b]

mov ax,0x4c00

int 21h

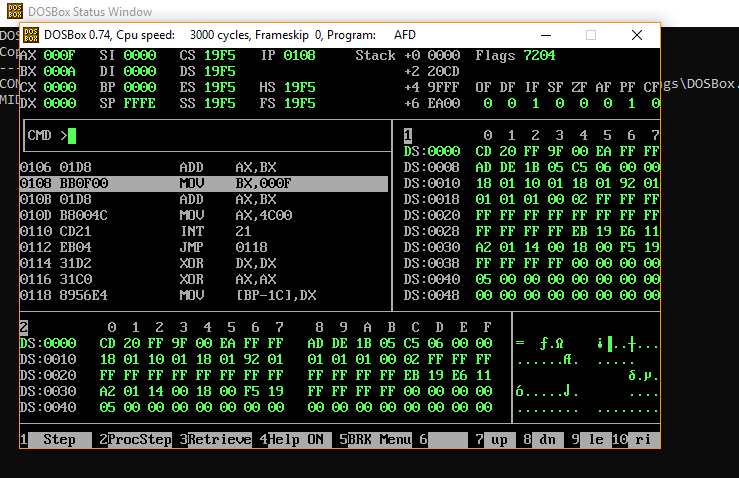
rollnum : dw 7744h

sum: dw 0

a : dw 0

b: dw 0

## **Debugging Screenshots**



# Activity 2

Write a program that calculates the following series:

f = ∑p

i

m

i=1

Use p and m obtained from the subroutine of Activity 1 and f MUST be stored in a 32bit

number. For the calculation of this series, you are required to make multiple subroutines.

Parameter passing from one subroutine

## **Assembly Language Code**

[org 0x0100]

jmp start

total: dw 0

a : dw 2

b : dw 5

mov ax,2

series:

mov dx,[b]

inc dx

loop3:

dec dx ;5

cmp dx,0 ;5>0

je end

loop2:

mov cx,dx ;cx 5

loop1:

mul ax; 4,8,16,32

dec cx ;4,3,2,1,0

cmp cx,0

jne loop1

cmp cx,0

je loop3

end : mov [total],ax

ret

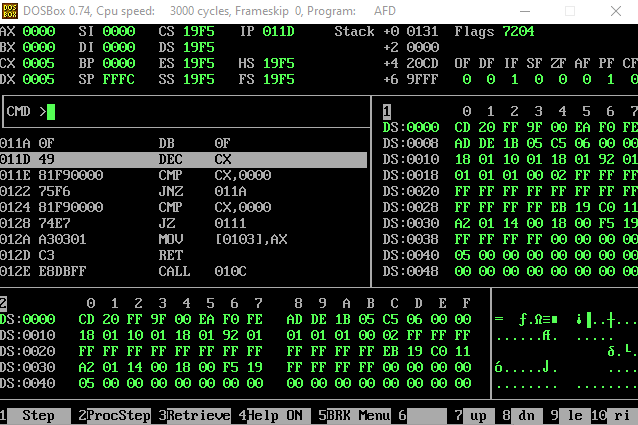
start:

call series

mov ax,0x4c00

int 0x21

## **Debugging Screenshots**



# Activity 3

Write a subroutine to determine n and r described as follows: If your roll number is 16L-4195

then n = max(4 + 1 + 9 + 5) = 9, and r = n − 2 = 7.

## **Assembly Language Code**

[org 0x0100]

mov cx, 0

mov bx, 0

mov ax, [rollno]

loop1:

add cx, 1

and ax, 000Fh

cmp ax, [max]

jae maximum

back:

mov ax, [rollno]

mov bx, 0

loop2:

add bx, 1

shr ax, 4

cmp cx, bx

jne loop2

cmp cx, 4

jne loop1

mov dx, [max]

jmp findn

maximum:

mov word[max], ax

jmp back

findn:

mov ax, [max]

mov word[n], ax

mov ax, [n]

mov word[r], ax

sub word[r], 2

mov dx, [r]

mov ax, 0x4c00

int 21h

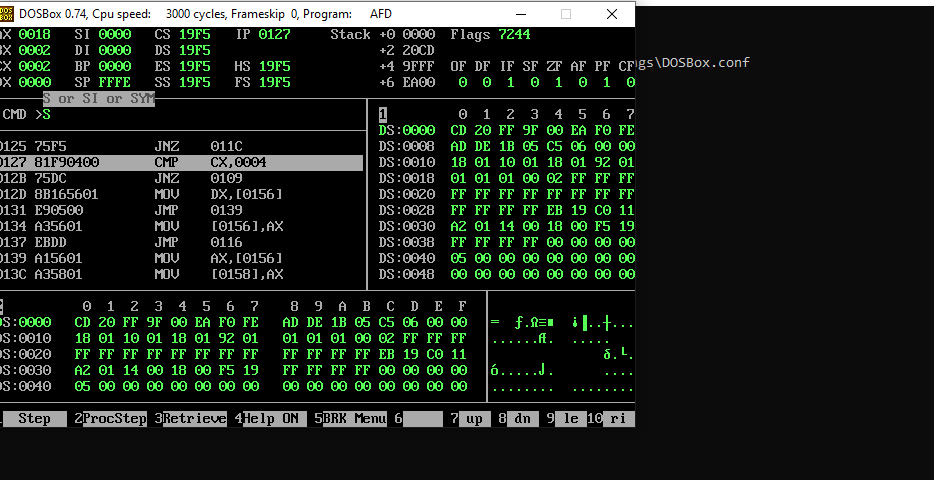
rollno: dw 1820h

max: dw 0

n: dw 0

r: dw 0

## **Debugging Screenshots**



# Activity 4

Write a subroutine to find factorial of the given number. Use subroutine developed in

Activity 3 and show the answers for n and r obtained.

## **Assembly Language Code**

[org 0x0100]

jmp start

num1: dw 5

factorial:

mov cx ,[num1]

mov bx,cx

dec cx

mov ax,0

l1:

mov dx,cx

l2 :

add ax,bx

dec dx

jnz l2

mov bx,ax

dec cx

cmp cx,2

jne l1

ret

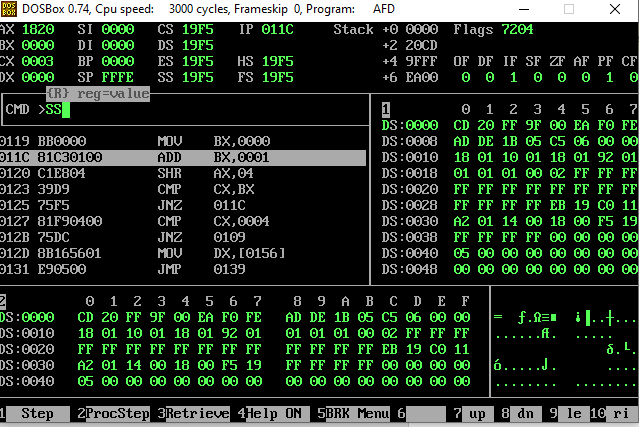
start:

call factorial

mov ax,0x4c00

int 0x21

## **Debugging Screenshots**



# Activity 5

Write a program capable of calculating nPr (for the n and r obtained from Activity 3):

nPr = n!/(n − r)!

Note: For the values of n and r from activity 3 you do not need to use any division command or

subroutine.

## **Assembly Language Code**

[org 0x0100]

jmp start

num1: dw 5

factorial:

mov cx ,[num1]

mov bx,cx

dec cx

mov ax,0

l1:

mov dx,cx

l2 :

add ax,bx

dec dx

jnz l2

mov bx,ax

dec cx

cmp cx,2

jne l1

ret

start:

call factorial

mov ax,0x4c00

int 0x21

## **Debugging Screenshots**

