MP Experiment No. 2

.model small

data segment

str db "hello world$"

ends

code segment

assume cs:code, ds:data

START :

mov ax, data

mov ds, ax

lea dx, str

mov ah, 09h

int 21h

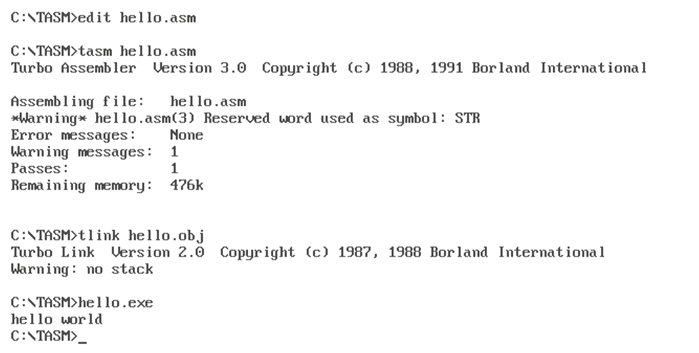
mov ah, 4ch

int 21h

ends

end start

**Output :**



**Experiment No : 3**

//16 bit addition

model small

.data

a dw 0202hh

b dw 0408h

.code

mov ax,@data

mov ds,ax

mov ax,a

mov bx,b

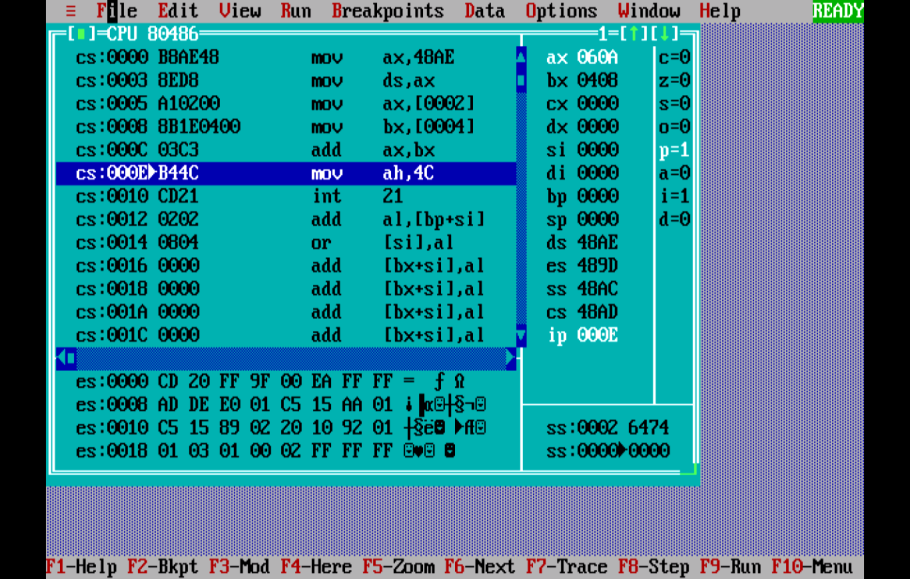
add ax,bx

mov ah,4ch

int 21h

end

OUTPUT



//8 bit addition

.model small

.data

a db 09H

b db 02H

.code

mov ax,@data

mov ds,ax

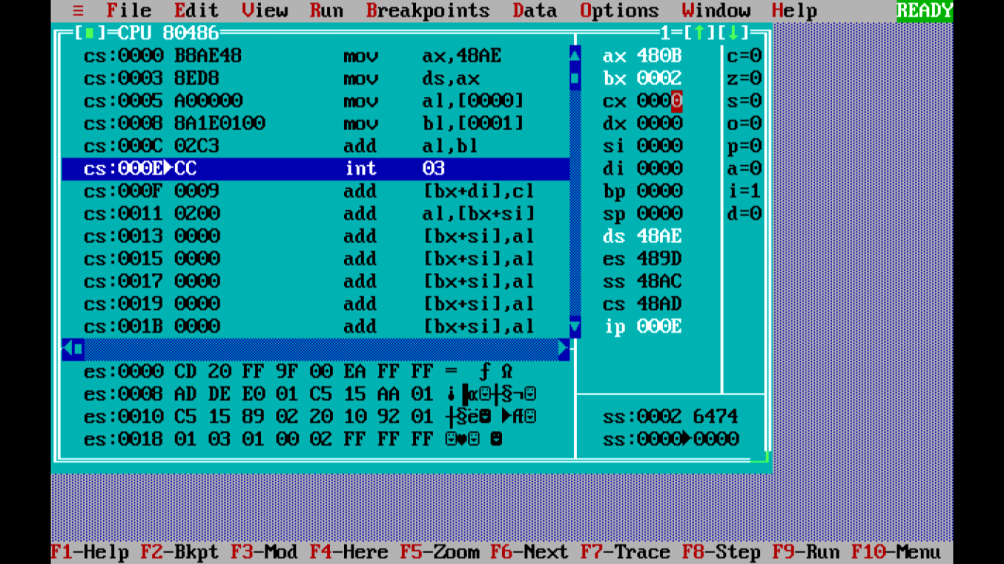
mov al,a

mov bl,b

add al,bl

int 3

end



//8 bit substraction

.model small

.data

a db 09H

b db 02H

.code

mov ax,@data

mov ds,ax

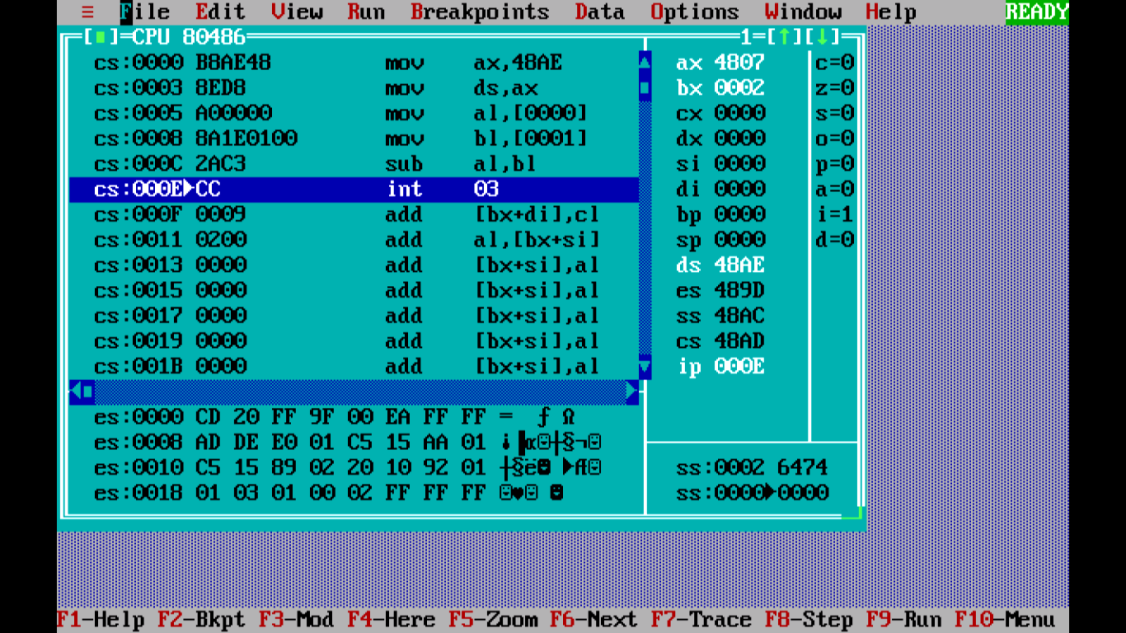
mov al,a

mov bl,b

sub al,bl

int 3

end



//16 bit substraction

model small

.data

a dw 0408h

b dw 0202h

.code

mov ax,@data

mov ds,ax

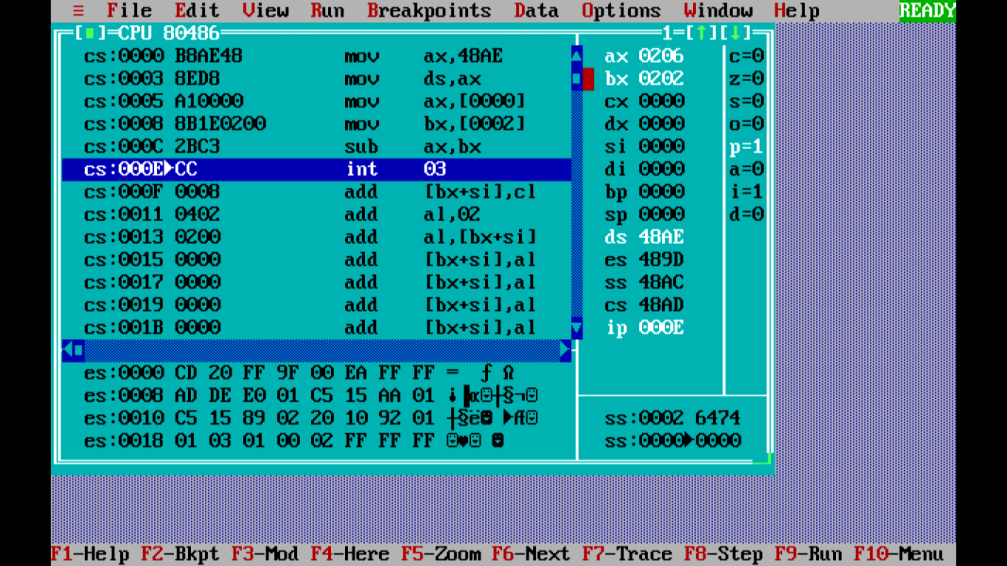
mov ax,a

mov bx,b

sub ax,bx

int 3

end



// 8 bit multiplication

model small

.data

a db 04h

b db 02h

.code

mov ax,@data

mov ds,ax

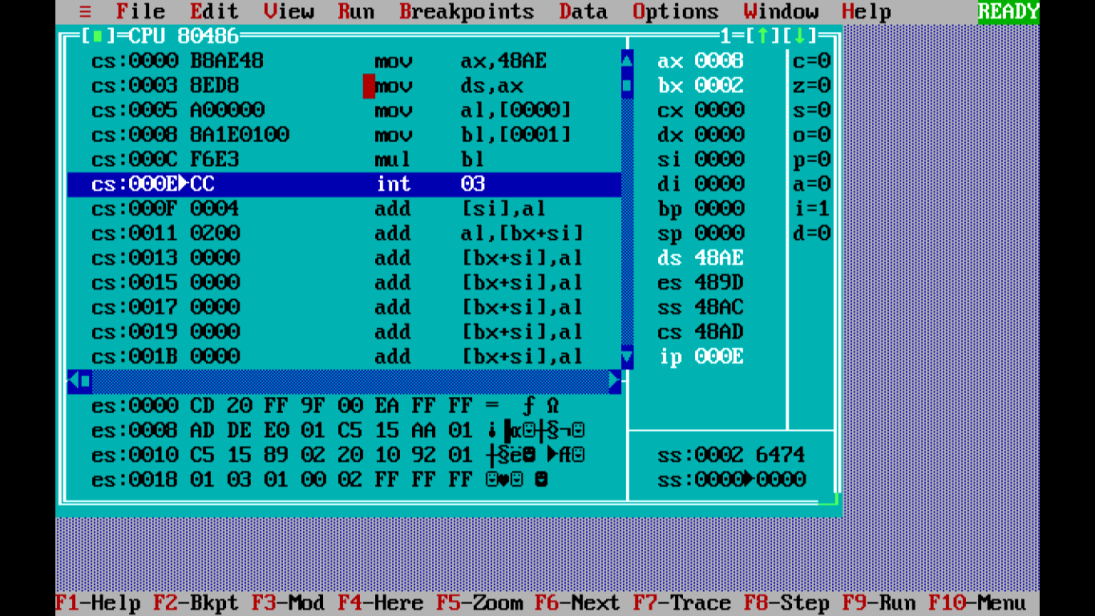
mov al,a

mov bl,b

mul bl

int 3

end



//16 bit multiplication

model small

.data

a dw 0408h

b dw 0202h

.code

mov ax,@data

mov ds,ax

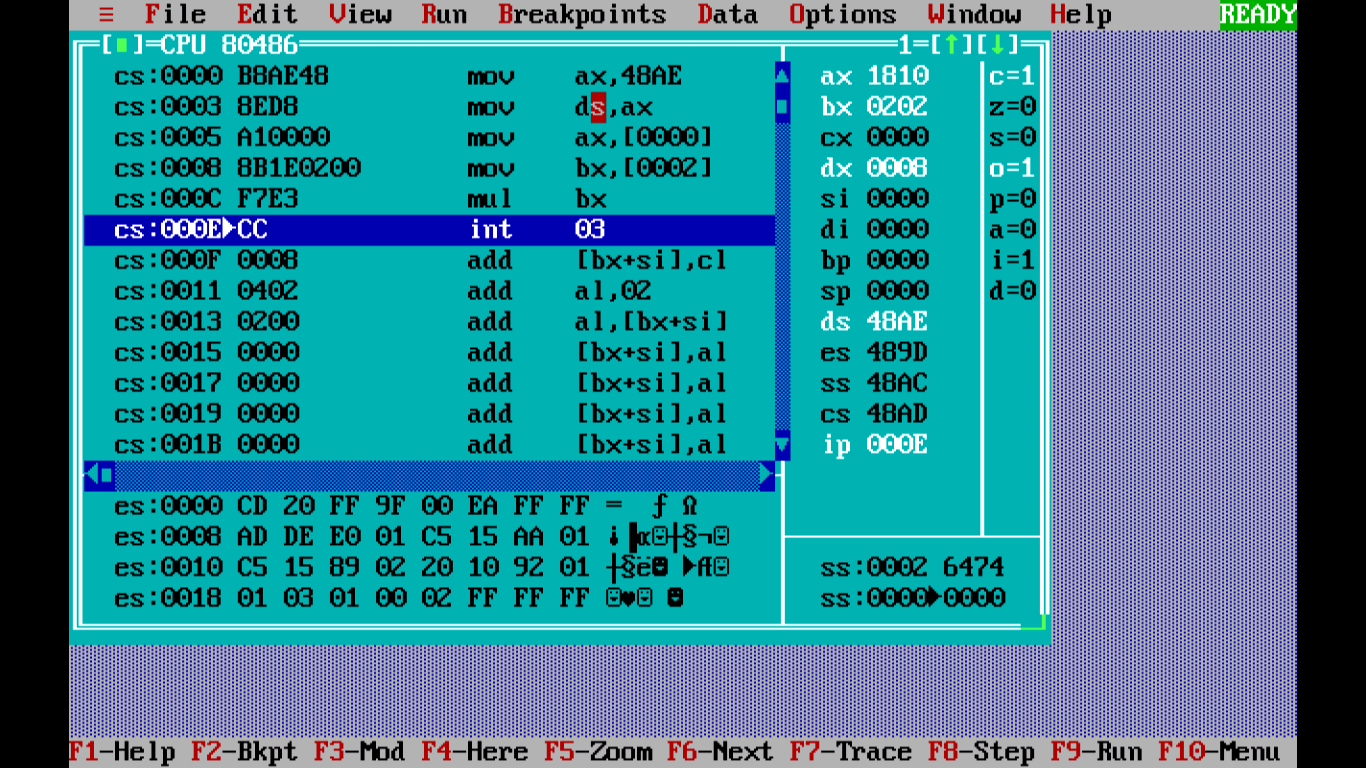
mov ax,a

mov bx,b

mul bx

int 3

end



//16bit division

.model small

.data

a dw 0005

b dw 0002

.code

mov ax,@data

mov ds,ax

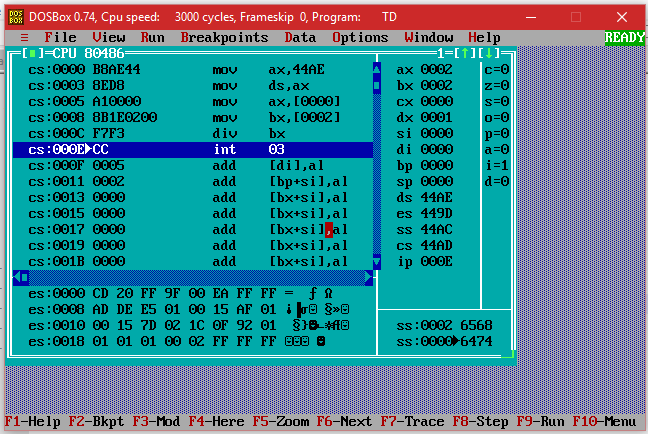
mov ax,a

mov bx,b

div bx

int 3

end

****

Experiment No : 4

data segment

n db 06h

fact dw ?

data ends

code segment

assume cs:code ,ds:data

start:

mov ax,data

mov ds,ax

mov ax,1

mov bl,n

mov bh,0

call factorial

mov fact,ax

mov ah,4ch

int 21h

factorial proc

cmp bx,1

je l1

push bx

dec bx

call factorial

pop bx

mul bx

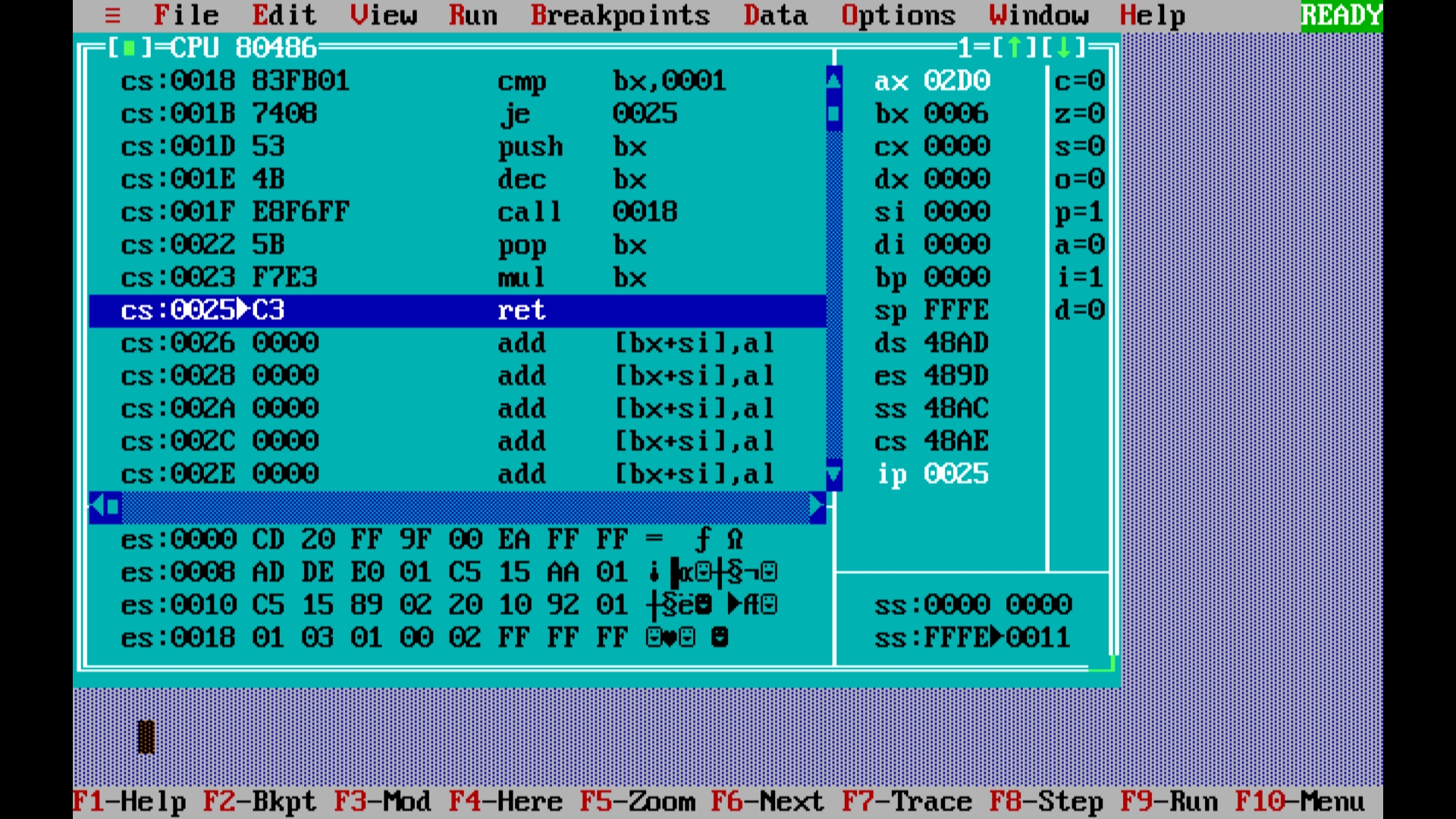
l1: ret

factorial endp

code ends

end start

output:



**Experiment No : 5**

.model small

.stack 100h

.data

n db 11h,99h,22h,88h,33h,77h,44h,66h

count dw 8

.code

begin:

mov ax,@data

mov ds,ax

mov cx,count

dec cx

nextscan:

mov bx,cx

mov si,0

nextcomp:

mov al,n[si]

mov dl,n[si+1]

cmp al,dl

jc noswap

mov n[si],dl

mov n[si+1],al

noswap:

inc si

dec bx

jnz nextcomp

loop nextscan

mov ah,4ch

int 21h

end begin

**OUTPUT:**

