LAB Assignment 67005_Saghir Ali

TASK 1:

```
  File  Edit  Search  View  Options  Help
                      C:\LAB.ASM
dosseg                    ;Tells assembler to order segemnts in standard way
.model small              ; Define memory model
.stack 100h               ; reserve 256 bytes for stack
.data                     ; data segments
.code                     ;code segment starts
main proc                 ; main procedure
mov dl, 's'               ;put ASCII of 's' in DL Register
mov ah, 2                 ; function 2 of INT 21H
int 21h                   ; CALL DOS INTRUPT TO PRINT 'S'
mov dl, 'a'               ; LOAD 'A'
mov ah,2
int 21h
mov dl,'g'                ; LOAD G
mov ah, 2
int 21h
mov dl, 'h'               ; LOAD H
mov ah, 2
int 21h
mov dl, 'i'               ; LOAD I
mov ah, 2
int 21h
mov dl, 'r'               ; LOAD R
```

```
Z:\>SET BLASTER=A220 I7 D1 H5 I6

Z:\>mount c c:/mp
Drive C is mounted as local directory c:/mp\

Z:\>c:

C:\>edit LAB.asm

C:\>link LAB.obj;

Microsoft (R) Overlay Linker  Version 3.60
Copyright (C) Microsoft Corp 1983-1987.  All rights reserved.


C:\>LAB.exe
saghir
```
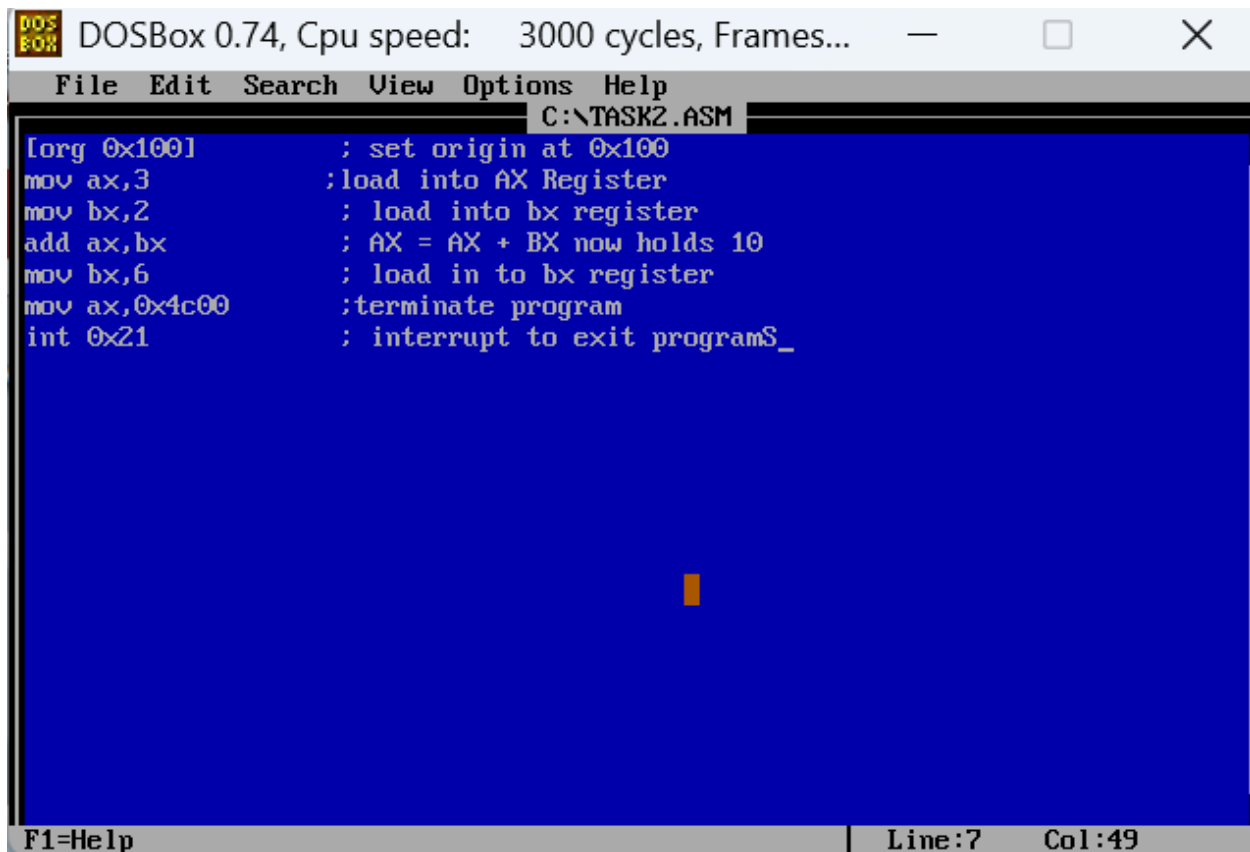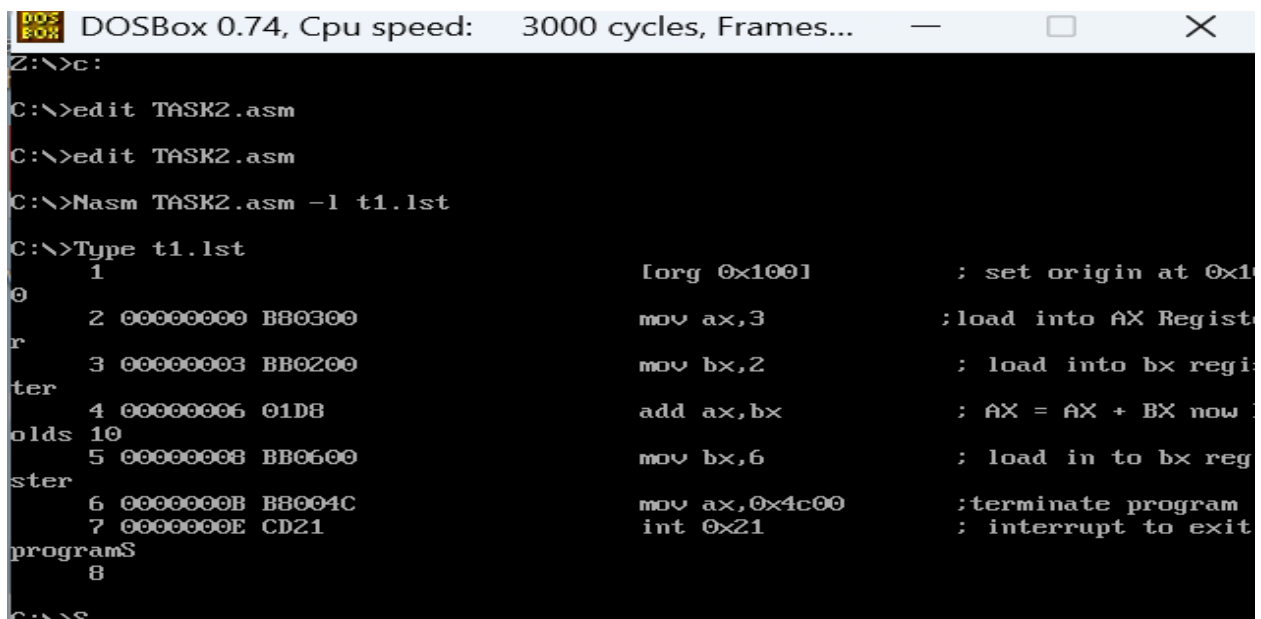
TASK 2:

```
[org 0x100]          ; set origin at 0x100
mov ax,3             ;load into AX Register
mov bx,2             ; load into bx register
add ax,bx            ; AX = AX + BX now holds 10
mov bx,6             ; load in to bx register
mov ax,0x4c00        ;terminate program
int 0x21             ; interrupt to exit programS_
```

DOSBox 0.74, Cpu speed: 3000 cycles, Frames...

File Edit Search View Options Help

C:\TASK2.ASM

F1=Help   Line:7   Col:49

```
Z:\>c:

C:\>edit TASK2.asm

C:\>edit TASK2.asm

C:\>Nasm TASK2.asm -l t1.lst

C:\>Type t1.lst
     1                              [org 0x100]          ; set origin at 0x1
0
     2 00000000 B80300             mov ax,3             ;load into AX Regist
r
     3 00000003 BB0200             mov bx,2             ; load into bx regi
ter
     4 00000006 01D8               add ax,bx            ; AX = AX + BX now
olds 10
     5 00000008 BB0600             mov bx,6             ; load in to bx reg
ster
     6 0000000B B8004C             mov ax,0x4c00        ;terminate program
     7 0000000E CD21               int 0x21             ; interrupt to exit
programS
     8

C:\>S
```

DOSBox 0.74, Cpu speed: 3000 cycles, Frames...

```
AX 0000   SI 0000   CS 19F5   IP 0100      Stack +0 0000   Flags 7202
BX 0000   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF   OF DF IF SF ZF AF PF C
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00    0  0  1  0  0  0  0
         S or SI or SYM
CMD >S
                                        1        0  1  2  3  4  5  6
                                        DS:0000  CD 20 FF 9F 00 EA F0 F
                                        DS:0008  AD DE 1B 05 C5 06 00 0
0100 7412         JZ      0114           DS:0010  18 01 10 01 18 01 92 0
0102 8B46F2       MOV     AX,[BP-0E]      DS:0018  01 01 01 00 02 FF FF F
0105 D1E0         SHL     AX,1            DS:0020  FF FF FF FF FF FF FF F
0107 9ACF27A201   CALL    01A2:27CF       DS:0028  FF FF FF FF EB 19 C0 1
010C 89C3         MOV     BX,AX           DS:0030  A2 01 14 00 18 00 F5 1
010E 89D0         MOV     AX,DX           DS:0038  FF FF FF FF 00 00 00 0
0110 89DA         MOV     DX,BX           DS:0040  05 00 00 00 00 00 00 0
0112 EB04         JMP     0118            DS:0048  00 00 00 00 00 00 00 0

2            0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
DS:0000  CD 20 FF 9F 00 EA F0 FE    AD DE 1B 05 C5 06 00 00    =  ƒ.Ω≡■   ¡ ..+..
DS:0010  18 01 10 01 18 01 92 01    01 01 01 00 02 FF FF FF    ......ff.   .....
DS:0020  FF FF FF FF FF FF FF FF    FF FF FF FF EB 19 C0 11              δ.└
DS:0030  A2 01 14 00 18 00 F5 19    FF FF FF FF 00 00 00 00    ó.....J.      ...
DS:0040  05 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    ........   .......

 Step     2ProcStep 3Retrieve 4Help ON  5BRK Menu 6        7 up  8 dn  9 le 10 ri
```


DOSBox 0.74, Cpu speed: 3000 cycles, Frames...

```
AX 0000   SI 0000   CS 19F5   IP 0102      Stack +0 0000   Flags 7200
BX 0000   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF   OF DF IF SF ZF AF PF C
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00    0  0  1  0  0  0  0
         S or SI or SYM
CMD >S
                                   -0020-
                                        1        0  1  2  3  4  5  6
                                        DS:0000  CD 20 FF 9F 00 EA F0 F
                                        DS:0008  AD DE 1B 05 C5 06 00 0
0100 7412         JZ      0114           DS:0010  18 01 10 01 18 01 92 0
0102 8B46F2       MOV     AX,[BP-0E]      DS:0018  01 01 01 00 02 FF FF F
0105 D1E0         SHL     AX,1            DS:0020  FF FF FF FF FF FF FF F
0107 9ACF27A201   CALL    01A2:27CF       DS:0028  FF FF FF FF EB 19 C0 1
010C 89C3         MOV     BX,AX           DS:0030  A2 01 14 00 18 00 F5 1
010E 89D0         MOV     AX,DX           DS:0038  FF FF FF FF 00 00 00 0
0110 89DA         MOV     DX,BX           DS:0040  05 00 00 00 00 00 00 0
0112 EB04         JMP     0118            DS:0048  00 00 00 00 00 00 00 0
0114 31D2         XOR     DX,DX

2            0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
DS:0000  CD 20 FF 9F 00 EA F0 FE    AD DE 1B 05 C5 06 00 00    =  ƒ.Ω≡■   ¡ ..+.
DS:0010  18 01 10 01 18 01 92 01    01 01 01 00 02 FF FF FF    ......ff.   .....
DS:0020  FF FF FF FF FF FF FF FF    FF FF FF FF EB 19 C0 11              δ.└
DS:0030  A2 01 14 00 18 00 F5 19    FF FF FF FF 00 00 00 00    ó.....J.      ...
DS:0040  05 00 00 00 00 00 00 00    00 00 00 00 00 00 00 00    ........   .......

1  Step    2ProcStep 3Retrieve 4Help ON  5BRK Menu 6        7 up  8 dn  9 le 10 ri
```

Screenshot 1

```
AX 0020   SI 0000   CS 19F5   IP 0105      Stack +0 0000   Flags 7200
BX 0000   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF   OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00    0  0  1  0  0  0  0  0
         {R} reg=value
 CMD >SS                                    1        0  1  2  3  4  5  6  7
                                            DS:0000  CD 20 FF 9F 00 EA F0 FE
 0102 8B46F2       MOV     AX,[BP-0E]        DS:0008  AD DE 1B 05 C5 06 00 00
 0105 D1E0         SHL     AX,1             DS:0010  18 01 10 01 18 01 92 01
 0107 9ACF27A201   CALL    01A2:27CF        DS:0018  01 01 01 00 02 FF FF FF
 010C 89C3         MOV     BX,AX            DS:0020  FF FF FF FF FF FF FF FF
 010E 89D0         MOV     AX,DX            DS:0028  FF FF FF FF EB 19 C0 11
 0110 89DA         MOV     DX,BX            DS:0030  A2 01 14 00 18 00 F5 19
 0112 EB04         JMP     0118             DS:0038  FF FF FF FF 00 00 00 00
 0114 31D2         XOR     DX,DX            DS:0040  05 00 00 00 00 00 00 00
 0116 31C0         XOR     AX,AX            DS:0048  00 00 00 00 00 00 00 00

2        0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
DS:0000  CD 20 FF 9F 00 EA F0 FE   AD DE 1B 05 C5 06 00 00   =  ƒ.Ω≡■  ¡|..┼...
DS:0010  18 01 10 01 18 01 92 01   01 01 01 00 02 FF FF FF   .......ﬁ.  .....
DS:0020  FF FF FF FF FF FF FF FF   FF FF FF FF EB 19 C0 11           δ.ᴸ.
DS:0030  A2 01 14 00 18 00 F5 19   FF FF FF FF 00 00 00 00   ó.....J.       ....
DS:0040  05 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ........  .......

 1  Step   2ProcStep 3Retrieve 4Help ON  5BRK Menu 6        7 up  8 dn  9 le 10 ri
```

Screenshot 2

```
X 0040   SI 0000   CS 19F5   IP 0107      Stack +0 0000   Flags 7210
X 0000   DI 0000   DS 19F5                      +2 20CD
X 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF   OF DF IF SF ZF AF PF CF
X 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00    0  0  1  0  0  1  0  0
        {R} reg=value
CMD >SSS                                    1        0  1  2  3  4  5  6  7
                                            DS:0000  CD 20 FF 9F 00 EA F0 FE
105 D1E0         SHL     AX,1              DS:0008  AD DE 1B 05 C5 06 00 00
107 9ACF27A201   CALL    01A2:27CF          DS:0010  18 01 10 01 18 01 92 01
10C 89C3         MOV     BX,AX              DS:0018  01 01 01 00 02 FF FF FF
10E 89D0         MOV     AX,DX              DS:0020  FF FF FF FF FF FF FF FF
110 89DA         MOV     DX,BX              DS:0028  FF FF FF FF EB 19 C0 11
112 EB04         JMP     0118               DS:0030  A2 01 14 00 18 00 F5 19
114 31D2         XOR     DX,DX              DS:0038  FF FF FF FF 00 00 00 00
116 31C0         XOR     AX,AX              DS:0040  05 00 00 00 00 00 00 00
118 8956E4       MOV     [BP-1C],DX         DS:0048  00 00 00 00 00 00 00 00

         0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
S:0000  CD 20 FF 9F 00 EA F0 FE   AD DE 1B 05 C5 06 00 00   =  ƒ.Ω≡■  ¡|..┼...
S:0010  18 01 10 01 18 01 92 01   01 01 01 00 02 FF FF FF   .......ﬁ.  .....
S:0020  FF FF FF FF FF FF FF FF   FF FF FF FF EB 19 C0 11           δ.ᴸ.
S:0030  A2 01 14 00 18 00 F5 19   FF FF FF FF 00 00 00 00   ó.....J.       ....
S:0040  05 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ........  .......

  Step   2ProcStep 3Retrieve 4Help ON  5BRK Menu 6        7 up  8 dn  9 le 10 ri
```