National University of Computer and Emerging Sciences

Lab Manual 01
Computer Organization and Assembly Language Lab

Course Instructor	Miss Aleena
Lab Instructor	Maham Saleem
Section	3F1
Semester	Fall 2021

Department of Computer Science

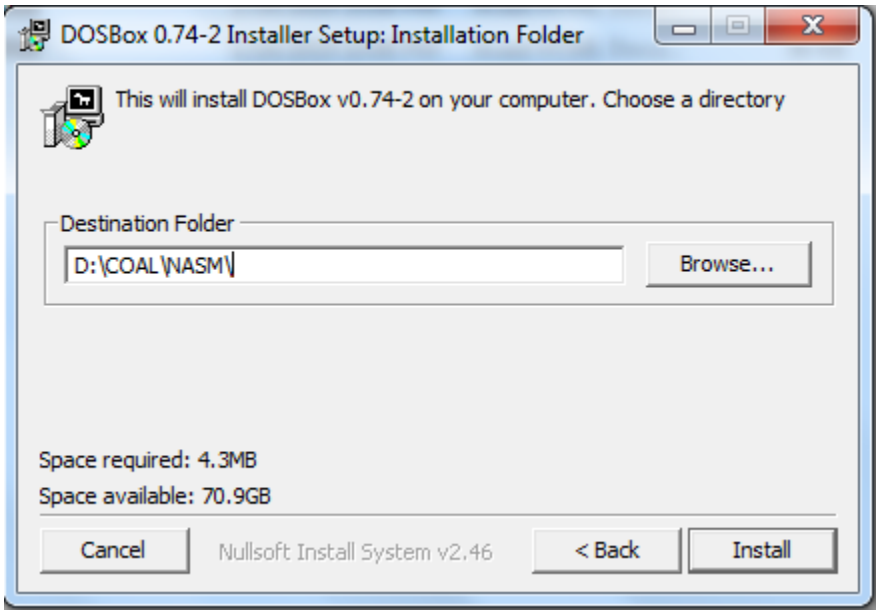
FAST-NU, Lahore, Pakistan

Activity 1: Setup

Download and install NASM, AFD and DOSBOX, according to the instructions, in your NASM folder.

AFD: https://drive.google.com/file/d/1eXnD3JEwBelFiJT6iIk7gluudV2Fu_iX/view?usp=sharing
NASM: https://drive.google.com/file/d/1ZoeE2MxjNaK7DdJKCacYfAJyn006MI_F/view?usp=sharing
Dosbox: https://drive.google.com/file/d/1DnaDIk4RoGBFDP1y4Dr3q89xwM3gx1d1/view?usp=sharing

Tutorial part 1: https://drive.google.com/file/d/1N3lWL8hsN0ZbhF3tlNwCWWwjJ_eHQqk6/view?usp=sharing
Tutorial part 2: https://drive.google.com/file/d/10p8qyaOVOwF5lDighrMKE-uNYQX-c3bL/view?usp=sharing



After installations double click "DOSBox 0.74-2 Options.bat" file and at the end of the file paste following lines:

```
MOUNT C D://COAL//NASM
C:
```

(We are mounting C drive to our folder where we have saved AFD and we will save our .asm file in this directory)

Activity 2: Running your First Program

Follow these step in order to run your first program:

1- Copy/paste following code in notepad

```
; this is a comment. Comment starts with semicolon
```

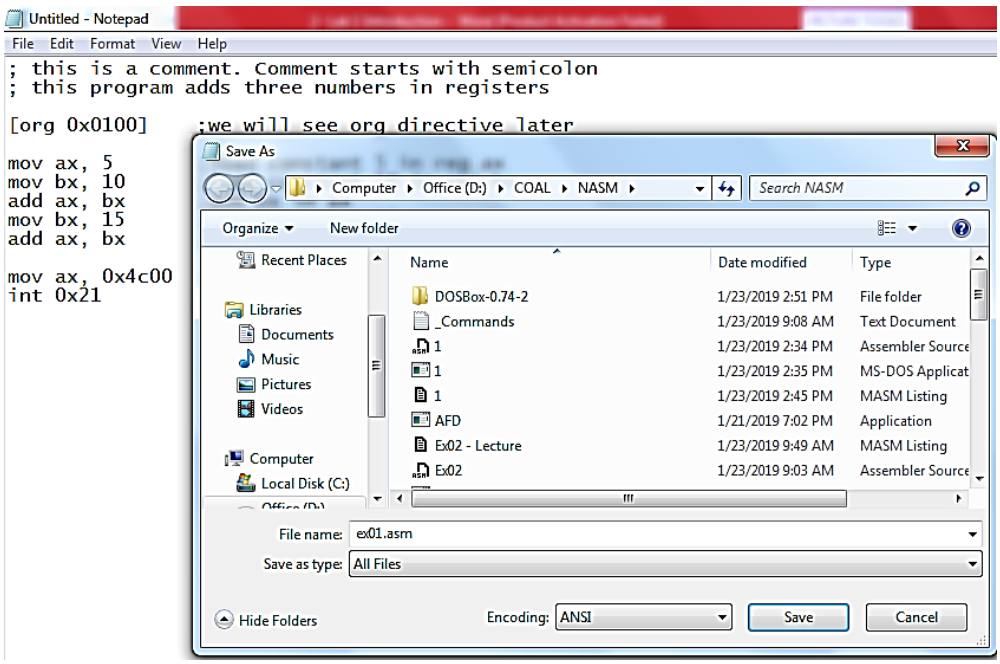
```
; this program adds three numbers in registers

[org 0x0100]     ;we will see org directive later

mov ax, 5        ; AX = 5
mov bx, 10       ; BX = 10
add ax, bx       ; AX = AX + BX
mov bx, 15       ; BX = 15
add ax, bx       ; AX = AX + BX

mov ax, 0x4c00 ;terminate the program
int 0x21
```

2- Save this file as “ex01.asm” in your NASM folder e.g. “D:\COAL\NASM”:



3- Go to NASM installation directory (e.g. “D:\COAL\NASM”). Double click **nasmpath.bat** (batch file) and type following command there. (Your .asm file and nasm should be in one folder)

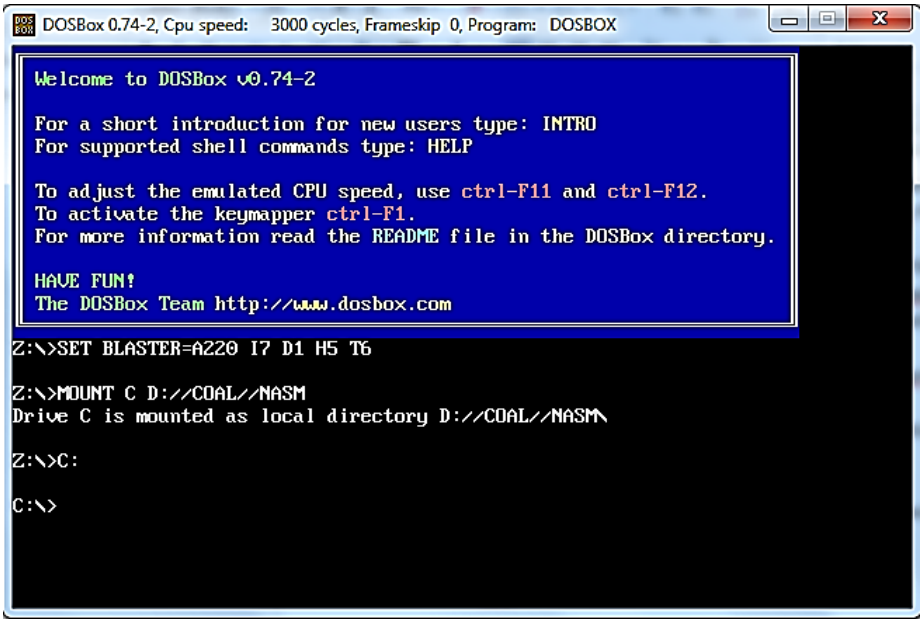
nasm ex01.asm -o ex01.com -l ex01.lst

4- Above command will assemble your code and create ex01.com and ex01.lst files. Open ex01.lst file in notepad.

a. What is opcode of instruction “mov ax, someConstant”
b. Verify the above opcode everywhere the instruction has been used.
c. What does “B80500” mean?
d. Verify the opcode of instruction “mov bx, someConstant” throughout the machine code.
e. What is the offset of first instruction?
f. Why are offsets of second and third instructions 3 and 6?

g. What should be the size of ex01.com file?
h. Right click ex01.com and verify its size.

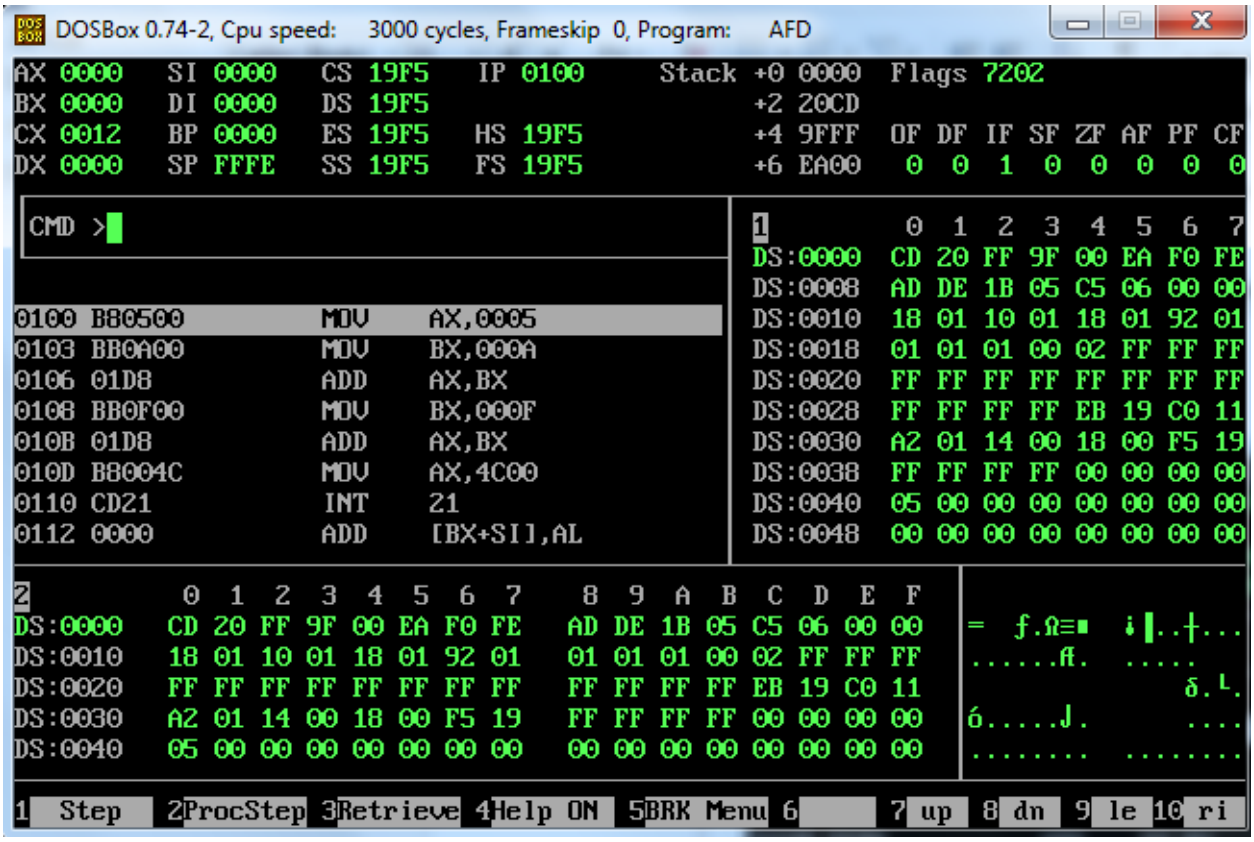
5- Open DOSBox (by double clicking dosbox.exe), following window will appear:



6- Write following command and press enter.

Afd ex01.com

(Your AFD.exe should be in same directory where we have installed everything)



7- Above command will open the debugger and load your ex01.com file in it.
   a. What is the value of IP register? And what will be its effect?
   b. Note the initial values of data registers
   c. Press F1 and watch the values of data registers

Activity 3: Explore different functions available in debugger (after completing activity 4).

Activity 4: Modify this program to generate the sum of first five entries of table of 3, using registers, and watch its execution in the debugger.

Help: [Approach 1] Can you do this using two registers only? [Approach 2] Can you do this using one register only if we have **add ax, 3** available in our instruction set? Try both of these approaches and watch the first five entries of table of 3 in AX.