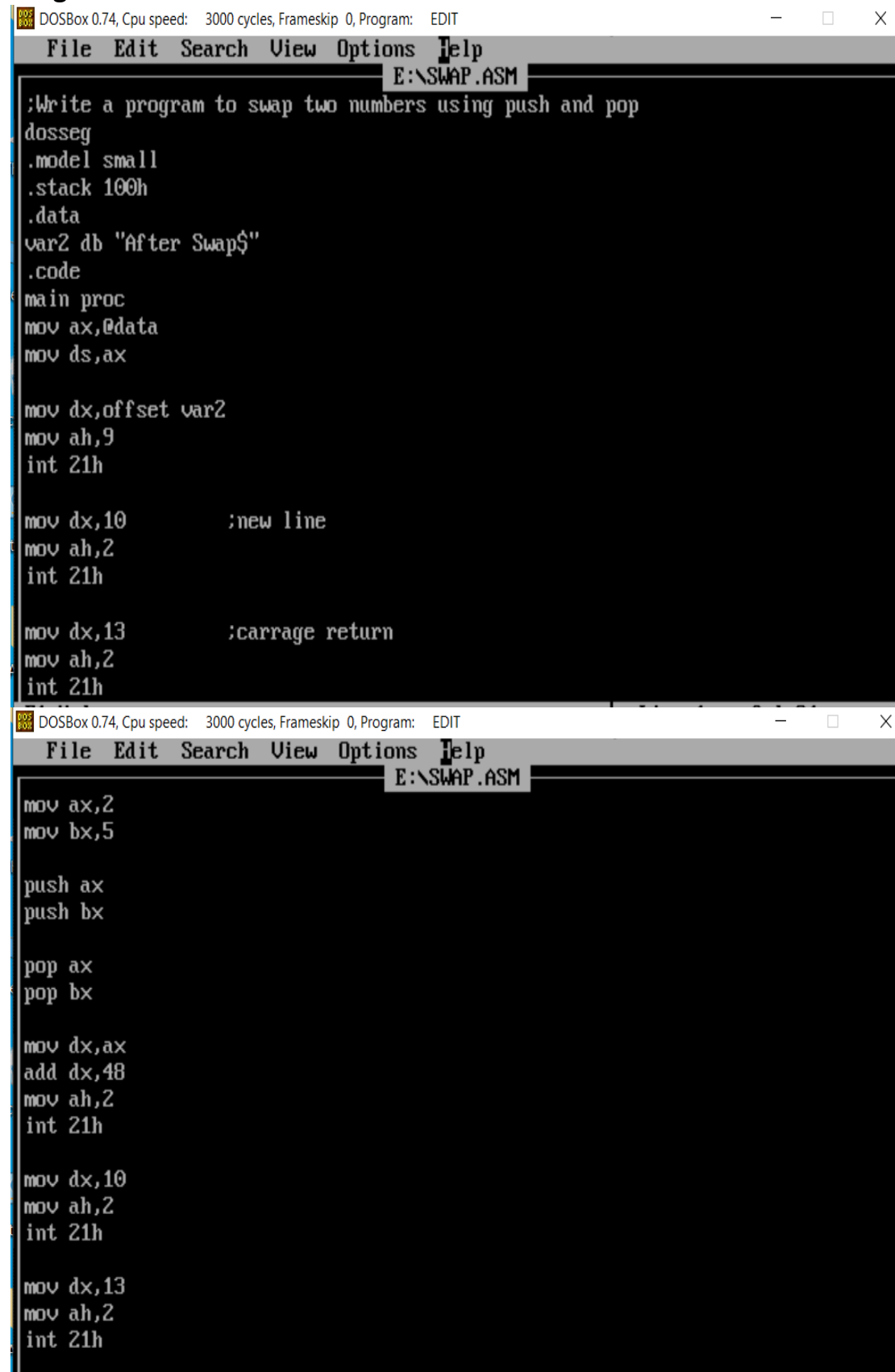


**NAME: MUHAMMAD ZOHAIB KHAN**  
**FATHER NAME: MUHAMMAD MUMTAZ**  
**KHAN**  
**SEAT NO: B21110106061**  
**CLASS: BSSE (4th SEMESTER)**  
**COURSE NAME: ASSEMBLY LANGUAGE**  
**COURSE CODE: 402**  
**DEPARTMENT: COMPUTER SCIENCE**

## Task 1: SWAP TWO NUMBERS

### Program:



The image shows two screenshots of a DOSBox window running a program named E:\SWAP.ASM. The window title bar indicates 'DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT'. The menu bar includes 'File', 'Edit', 'Search', 'View', 'Options', and 'Help'. The first screenshot shows the initial assembly code, which includes comments and instructions for setting up the program, including a comment about using push and pop to swap two numbers. The second screenshot shows the continuation of the assembly code, focusing on the push and pop instructions used for the swap.

```
;Write a program to swap two numbers using push and pop
dosseg
.model small
.stack 100h
.data
var2 db "After Swap$"
.code
main proc
mov ax,@data
mov ds,ax

mov dx,offset var2
mov ah,9
int 21h

mov dx,10      ;new line
mov ah,2
int 21h

mov dx,13      ;carrage return
mov ah,2
int 21h

mov ax,2
mov bx,5

push ax
push bx

pop ax
pop bx

mov dx,ax
add dx,48
mov ah,2
int 21h

mov dx,10
mov ah,2
int 21h

mov dx,13
mov ah,2
int 21h
```

```
mov dx,bx
add dx,48
mov ah,2
int 21h

mov ah,4ch
int 21h

main endp
end main
```

## OUTPUT:

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
Z:\>e:

E:\>edit swap.asm

E:\>masm swap.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

51670 + 464874 Bytes symbol space free

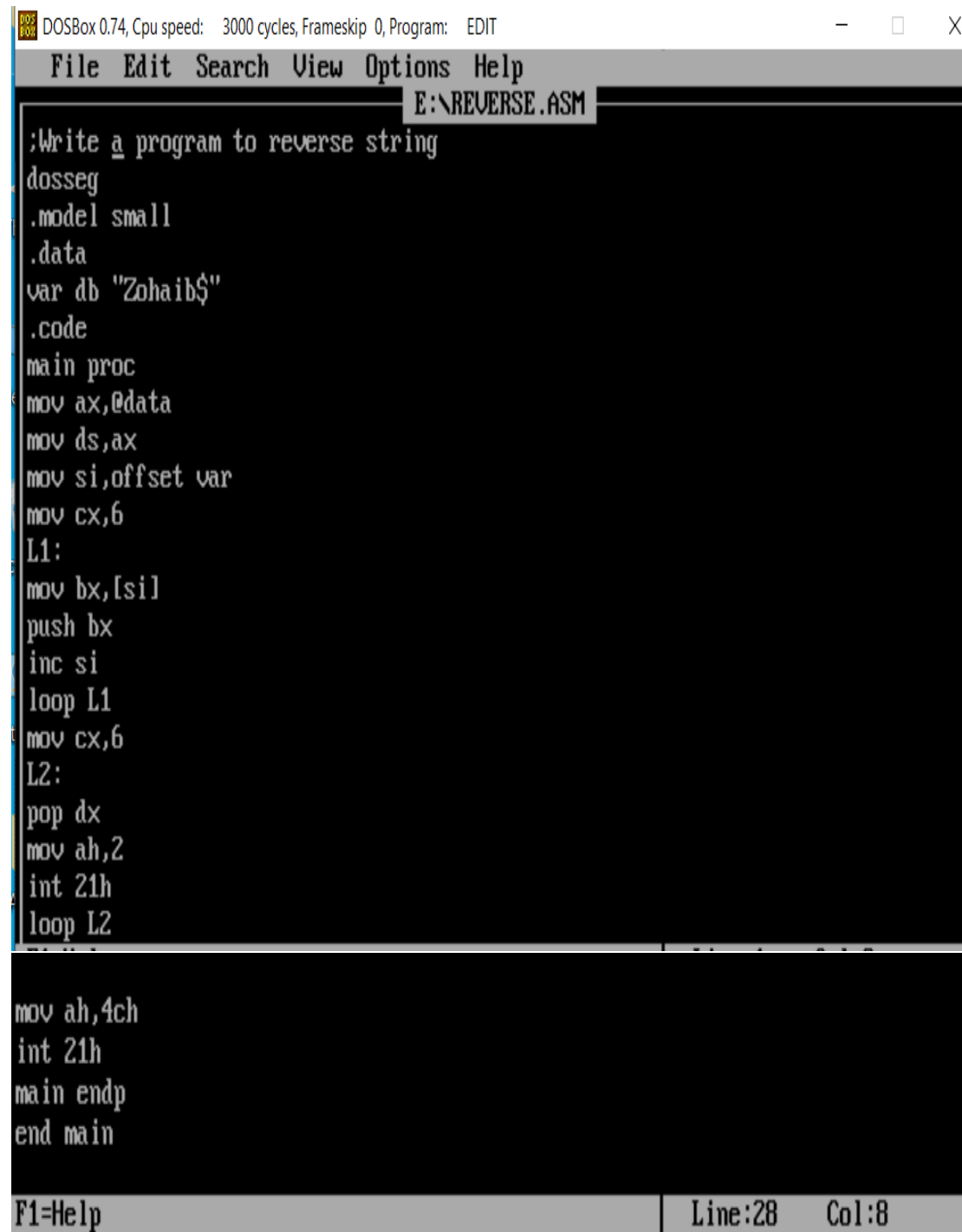
0 Warning Errors
0 Severe Errors

E:\>link swap.obj;

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

E:\>swap.exe
After Swap
5
2
```

## Task 2: Reverse String



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT
File Edit Search View Options Help
E:\REVERSE.ASM
;Write a program to reverse string
dosseg
.model small
.data
var db "Zohaib$"
.code
main proc
mov ax,@data
mov ds,ax
mov si,offset var
mov cx,6
L1:
mov bx,[si]
push bx
inc si
loop L1
mov cx,6
L2:
pop dx
mov ah,2
int 21h
loop L2

mov ah,4ch
int 21h
main endp
end main
F1=Help Line:28 Col:8
```

## OUTPUT:

```
E:\>edit reverse.asm

E:\>masm reverse.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

51624 + 464920 Bytes symbol space free

0 Warning Errors
0 Severe Errors

E:\>link reverse.obj;

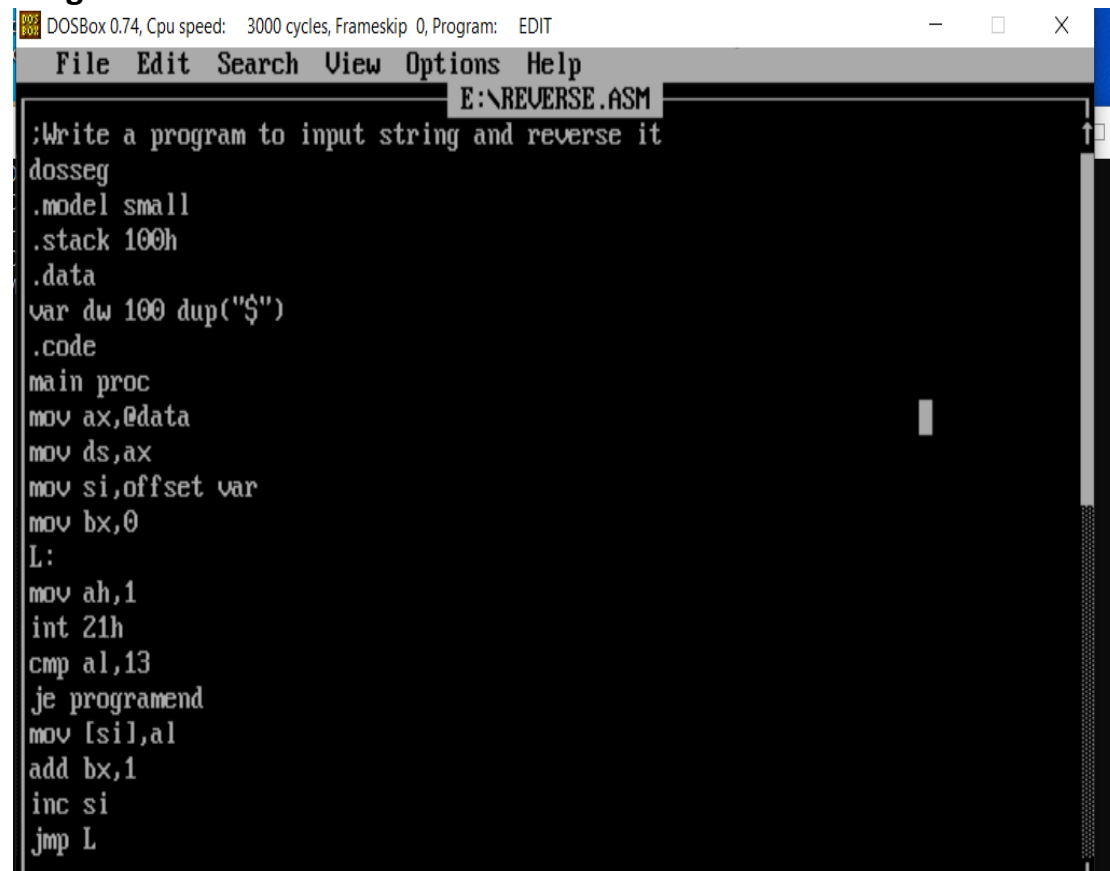
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

LINK : warning L4021: no stack segment

E:\>reverse.exe
biahoZ
```

## Task 3(input string and reverse it)

### Program:



The screenshot shows a DOSBox window with the title bar "DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT". The menu bar includes "File", "Edit", "Search", "View", "Options", and "Help". The active window is titled "E:\REVERSE.ASM" and contains the following assembly code:

```
;Write a program to input string and reverse it
dosseg
.model small
.stack 100h
.data
var dw 100 dup("$")
.code
main proc
mov ax,@data
mov ds,ax
mov si,offset var
mov bx,0
L:
mov ah,1
int 21h
cmp al,13
je programend
mov [si],al
add bx,1
inc si
jmp L
```

```
programend:
mov dx,offset var
mov ah,9
int 21h

mov cx,bx
mov si,offset var
L1:
mov ax,[si]
push ax
inc si
loop L1

L2:
pop ax
loop L2
mov ah,9
int 21h

mov ah,4ch
int 21h
main endp
```