SHAHEED ZULFIKAR ALI BHUTTO
INSTITUTE OF SCIENCE AND TECHNOLOGY

Computer Organization and Assembly Language

Laboratory Manual

CSCL-2201

Semester 3rd

Fall-2022

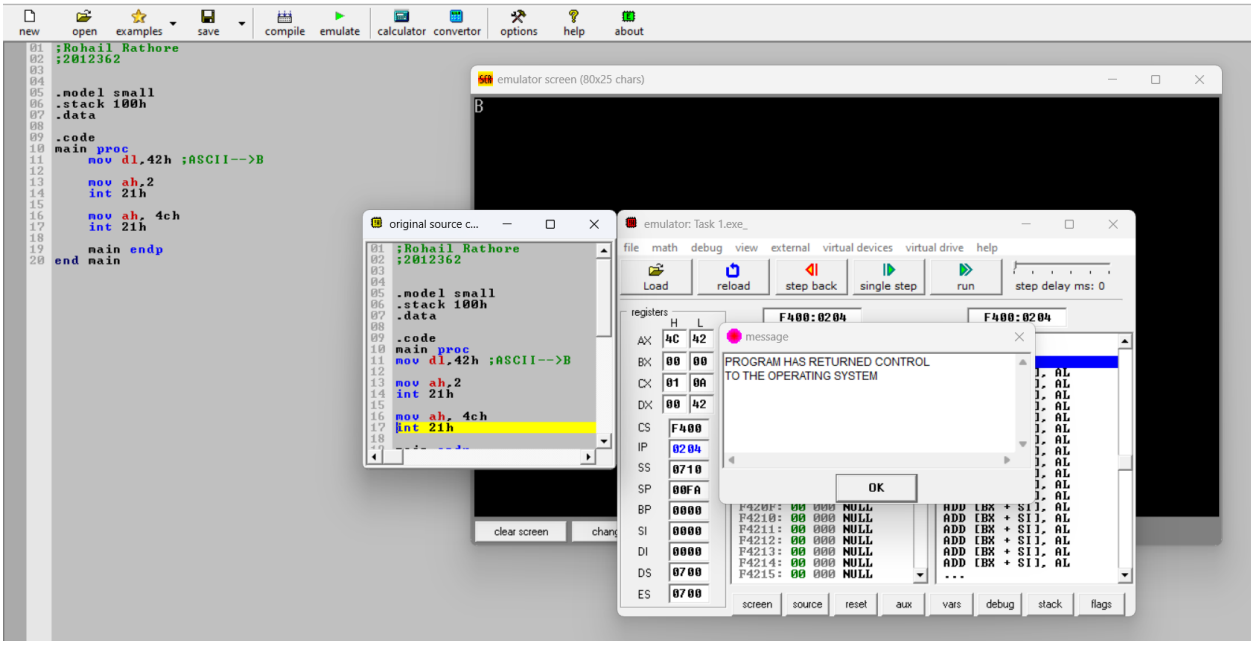
Student's Name: Rohail Rathore

Roll Number: 2012362

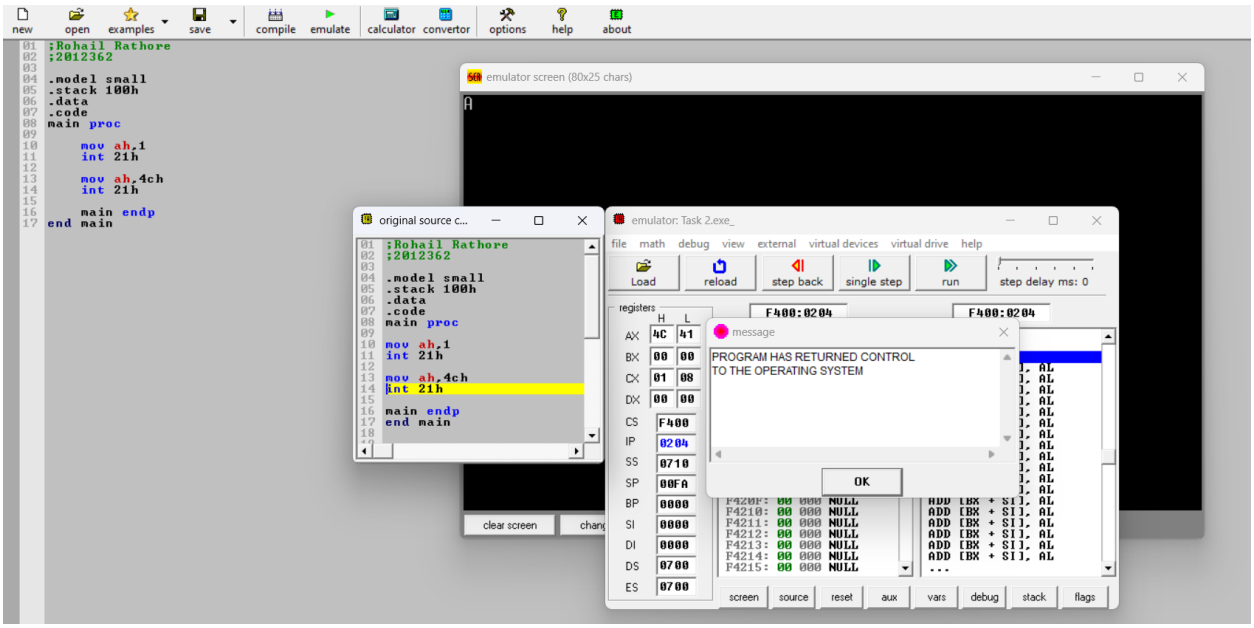
Lab Instructor: Engr. Mubeen Ahmed Khan

Department of Computing Sciences
Shaheed Zulfiqar Ali Bhutto Institute Of Science And Technology,Karach
iTel:(021)111-922-478, Fax: (021)35830446, Email: info@szabist.edu.pk

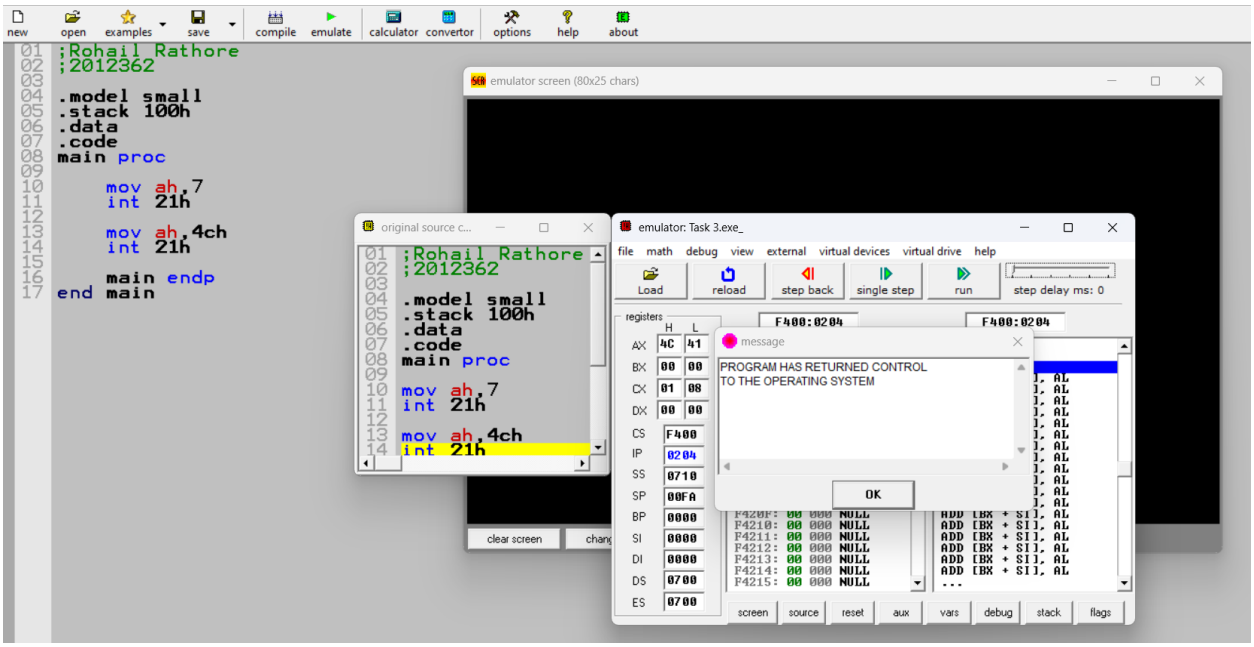
TASK-1:


new open examples save compile emulate calculator convertor options help about
;Rohail Rathore
;2012362
.model small
.stack 100h
.data
.code
main proc
mov dl,42h ;ASCII-->B
mov ah,2
int 21h
mov ah, 4ch
int 21h
main endp
end main
emulator screen (80x25 chars)
B
clear screen
original source c...
emulator: Task 1.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
registers
AX 4C 42
BX 00 00
CX 01 0A
DX 00 42
CS F400
IP 0204
SS 0710
SP 00FA
BP 0000
SI 0000
DI 0000
DS 0700
ES 0700
F400:0204
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
screen source reset aux vars debug stack flags

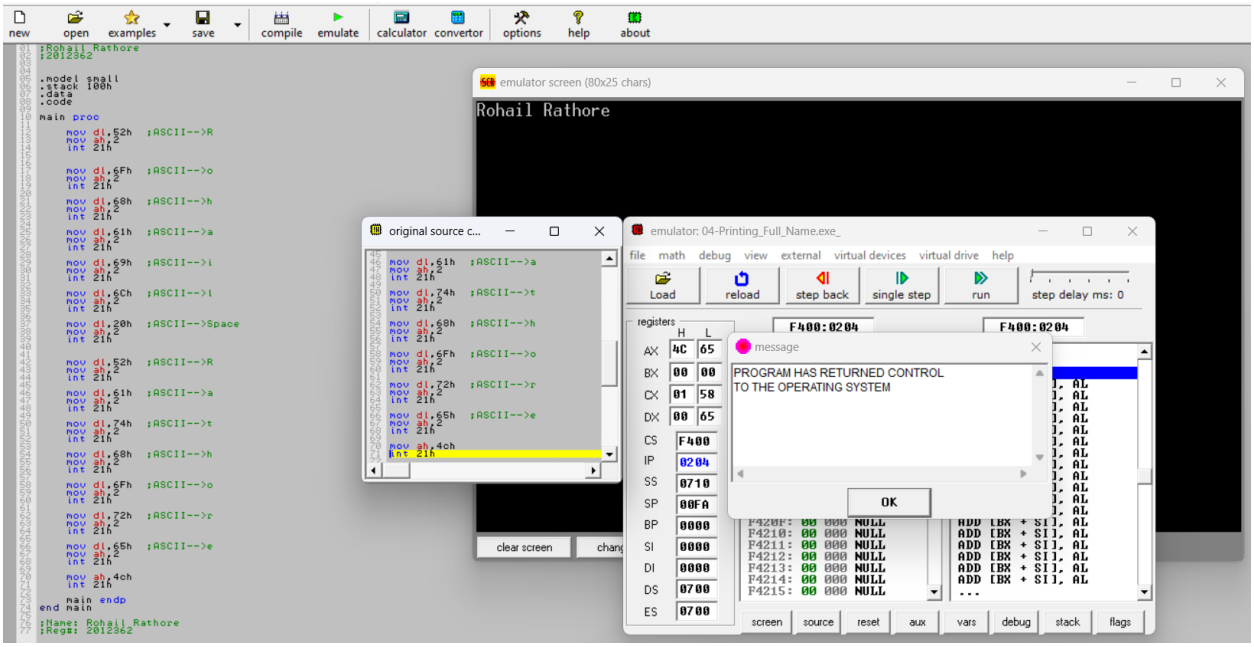
Task-2:


new open examples save compile emulate calculator convertor options help about
;Rohail Rathore
;2012362
.model small
.stack 100h
.data
.code
main proc
mov ah,1
int 21h
mov ah,4ch
int 21h
main endp
end main
emulator screen (80x25 chars)
A
clear screen
original source c...
emulator: Task 2.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
registers
AX 4C 41
BX 00 00
CX 01 08
DX 00 00
CS F400
IP 0204
SS 0710
SP 00FA
BP 0000
SI 0000
DI 0000
DS 0700
ES 0700
F400:0204
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
screen source reset aux vars debug stack flags

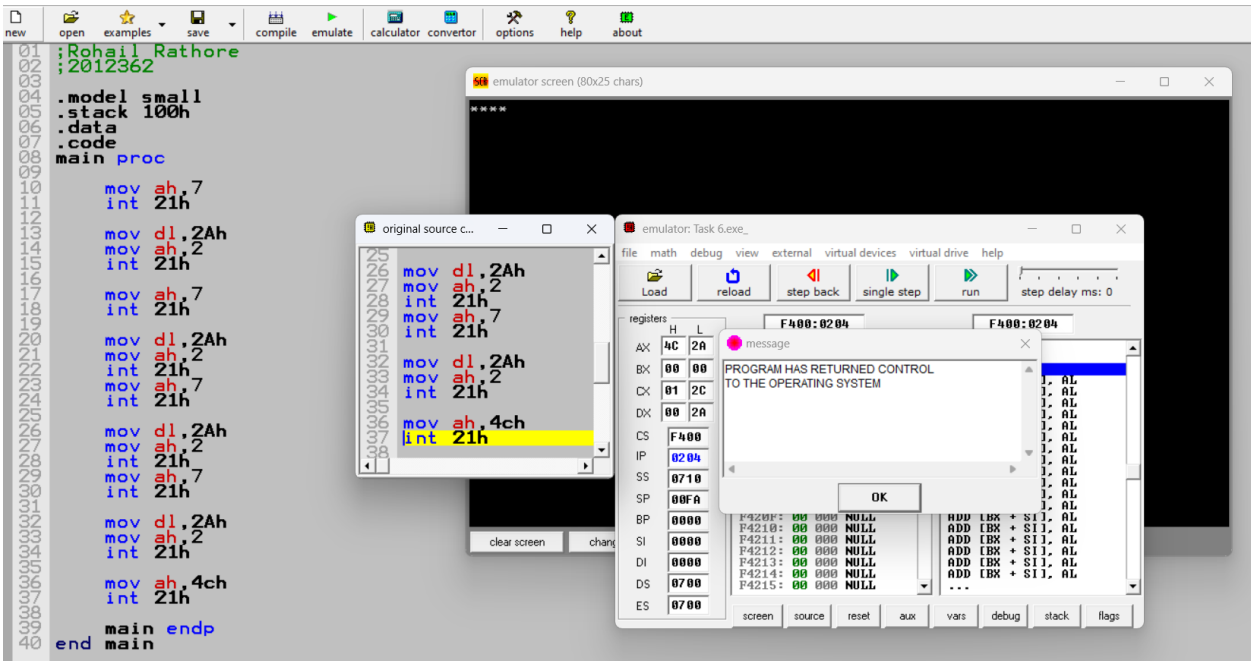
Task-3:



Task-4:

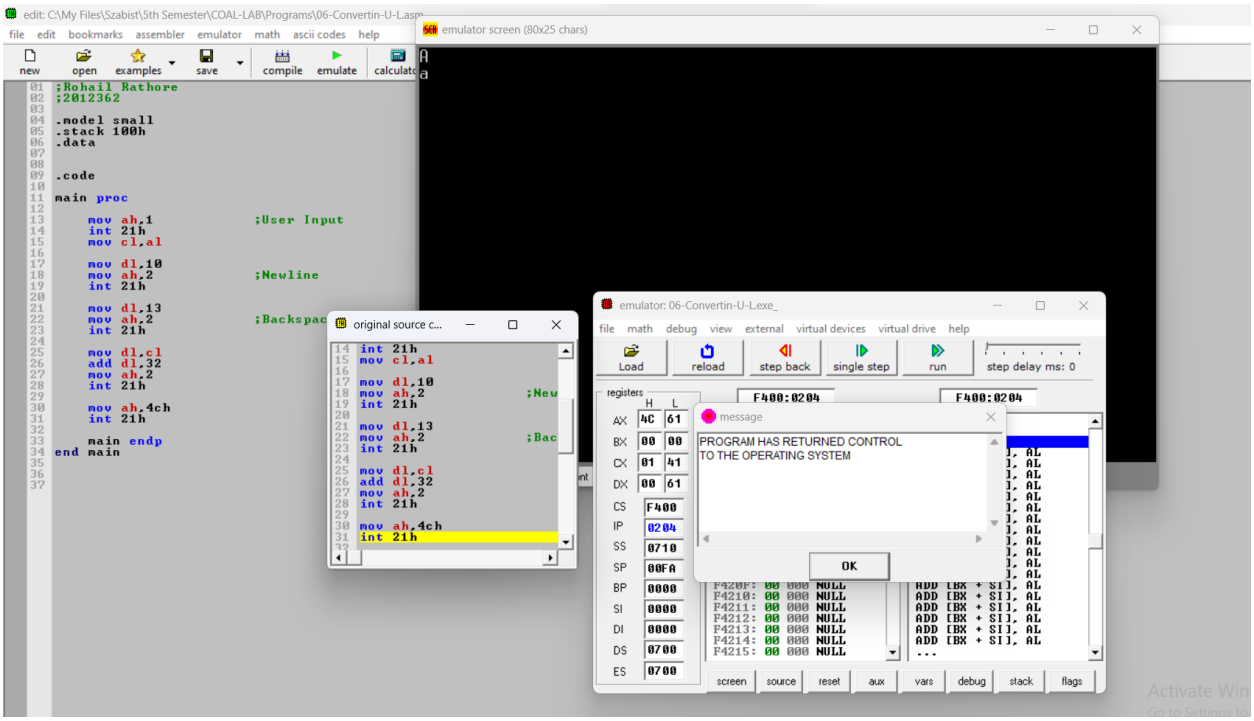


Task-5:



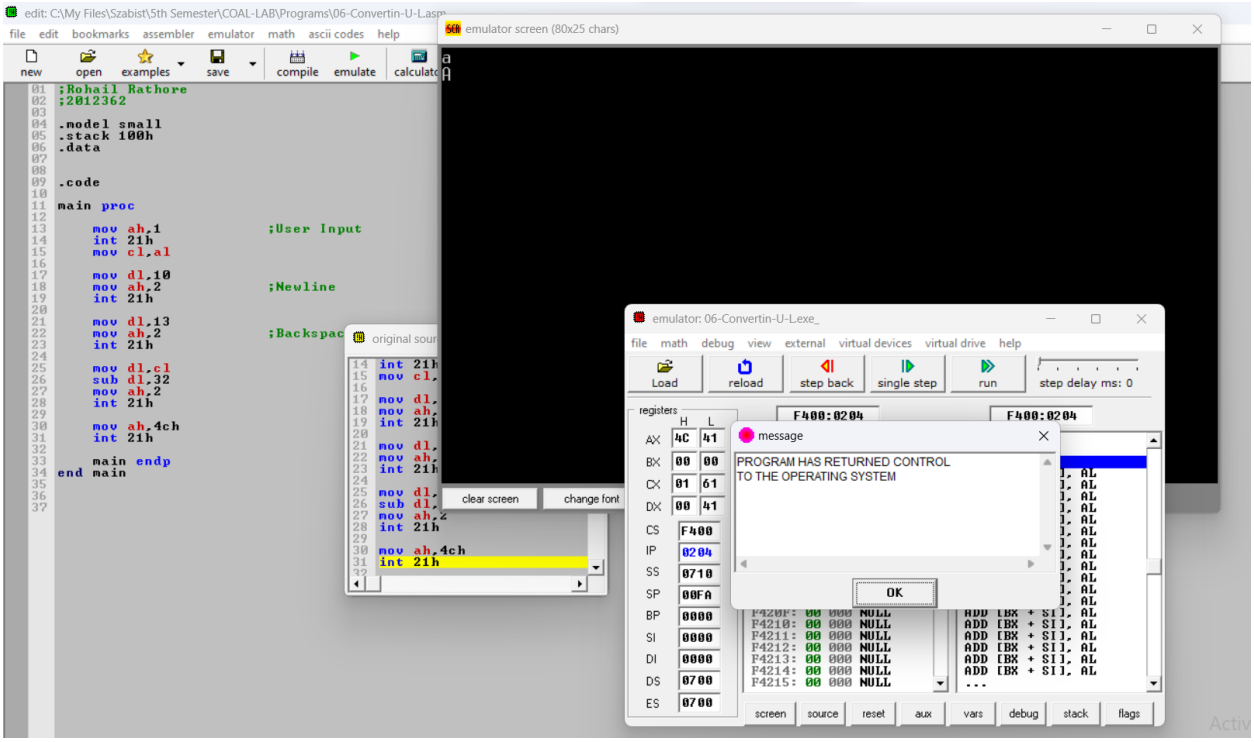
```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data
.code
main proc

    mov ah,7
    int 21h

    mov dl,2Ah
    mov ah,2
    int 21h

    mov ah,7
    int 21h

    mov dl,2Ah
    mov ah,2
    int 21h
    mov ah,7
    int 21h

    mov dl,2Ah
    mov ah,2
    int 21h
    mov ah,7
    int 21h

    mov dl,2Ah
    mov ah,2
    int 21h

    mov ah,4ch
    int 21h

    main endp
end main
```

Task-6:

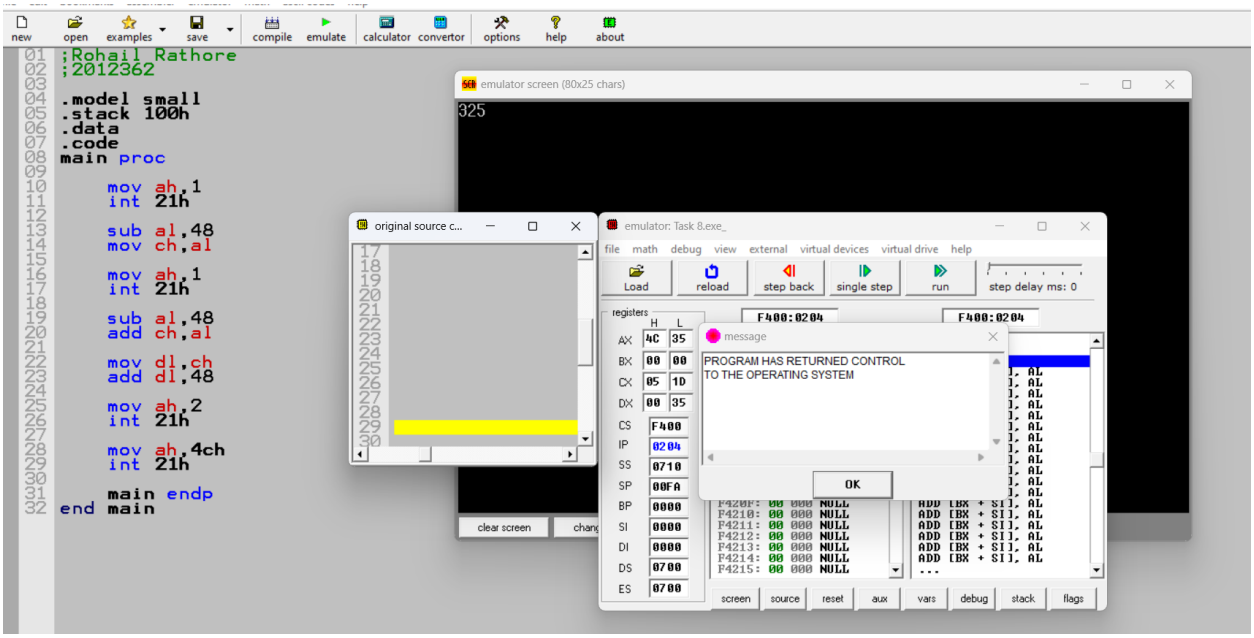


```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data

.code

main proc

    mov ah,1            ;User Input
    int 21h
    mov cl,al

    mov dl,10
    mov ah,2            ;Newline
    int 21h

    mov dl,13
    mov ah,2            ;Backspac
    int 21h

    mov dl,cl
    add dl,32
    mov ah,2
    int 21h

    mov ah,4ch
    int 21h

    main endp
end main
```

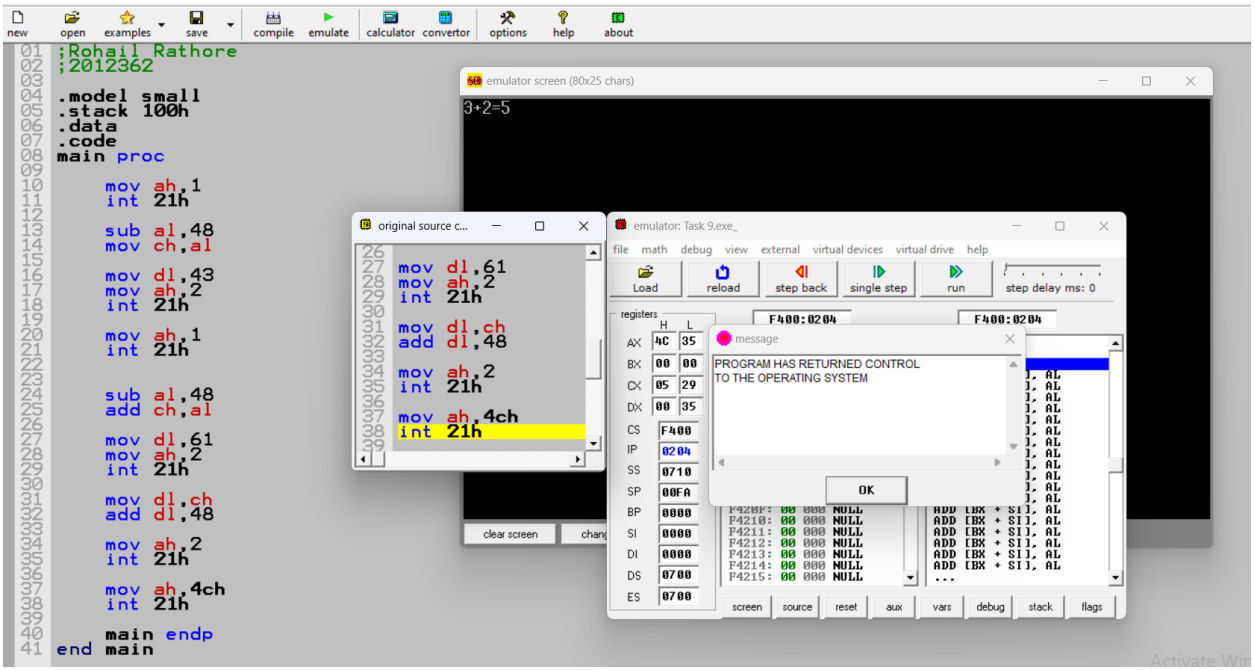
Task-7:


;Rohail Rathore
;2012362
.model small
.stack 100h
.data
.code
main proc
mov ah,1 ;User Input
int 21h
mov cl,al
mov dl,10
mov ah,2 ;Newline
int 21h
mov dl,13
mov ah,2 ;Backspac
int 21h
mov dl,cl
sub dl,32
mov ah,2
int 21h
mov ah,4ch
int 21h
main endp
end main
emulator screen (80x25 chars)
emulator: 06-Convertin-U-Lexe_
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
clear screen
change font

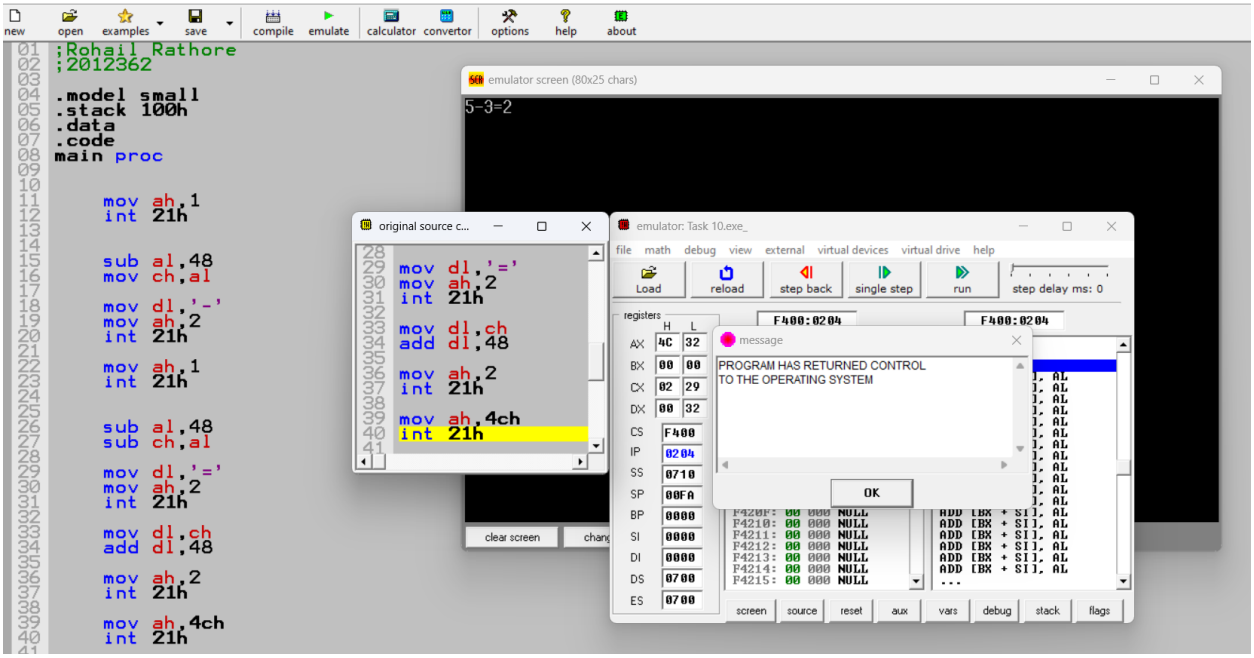
Task-8:


;Rohail Rathore
;2012362
.model small
.stack 100h
.data
.code
main proc
mov ah,1
int 21h
sub al,48
mov ch,al
mov ah,1
int 21h
sub al,48
add ch,al
mov dl,ch
add dl,48
mov ah,2
int 21h
mov ah,4ch
int 21h
main endp
end main
emulator screen (80x25 chars)
325
original source c...
emulator: Task 8.exe_
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK

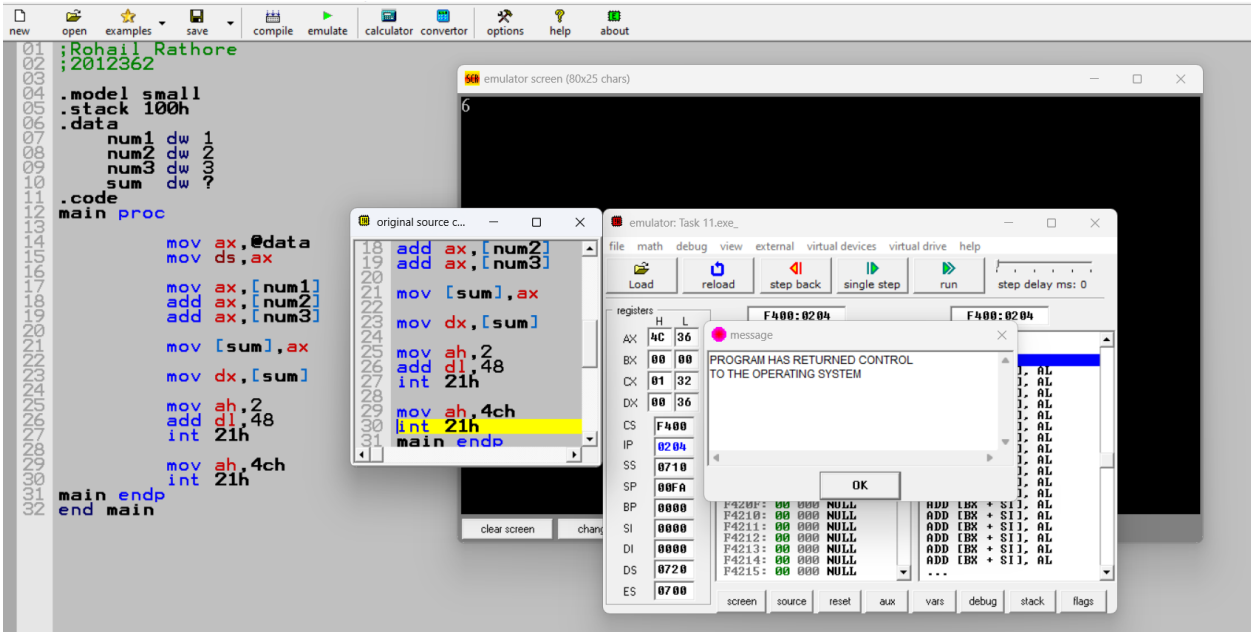
Task-9:



Task-10:

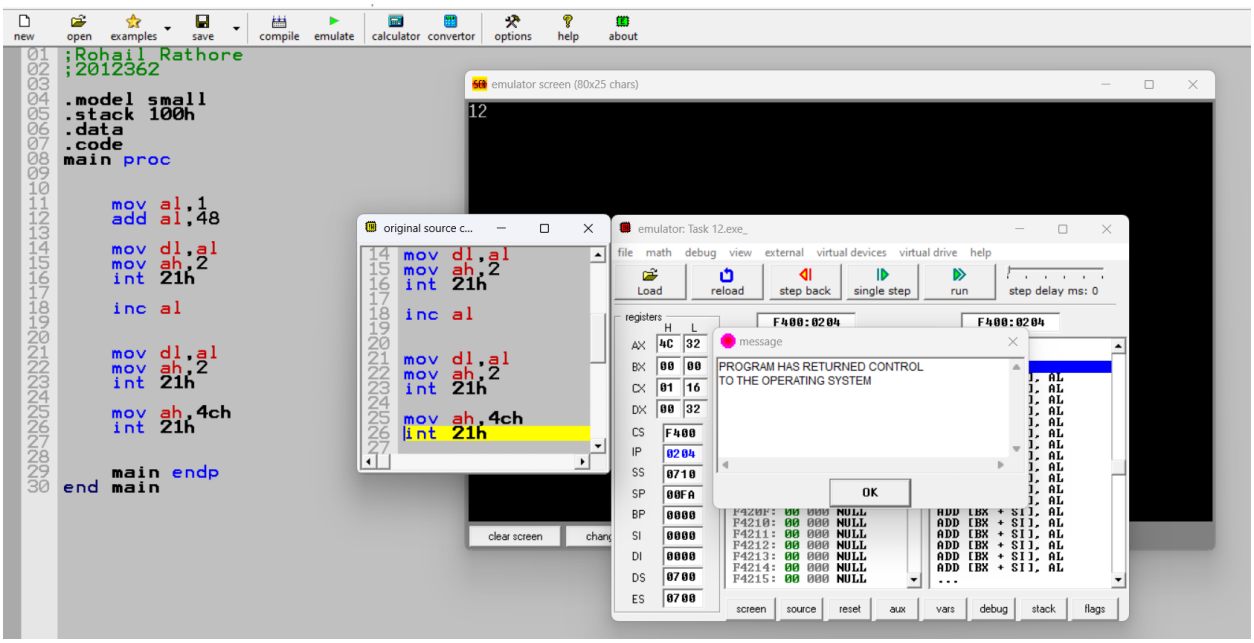


Task-11:



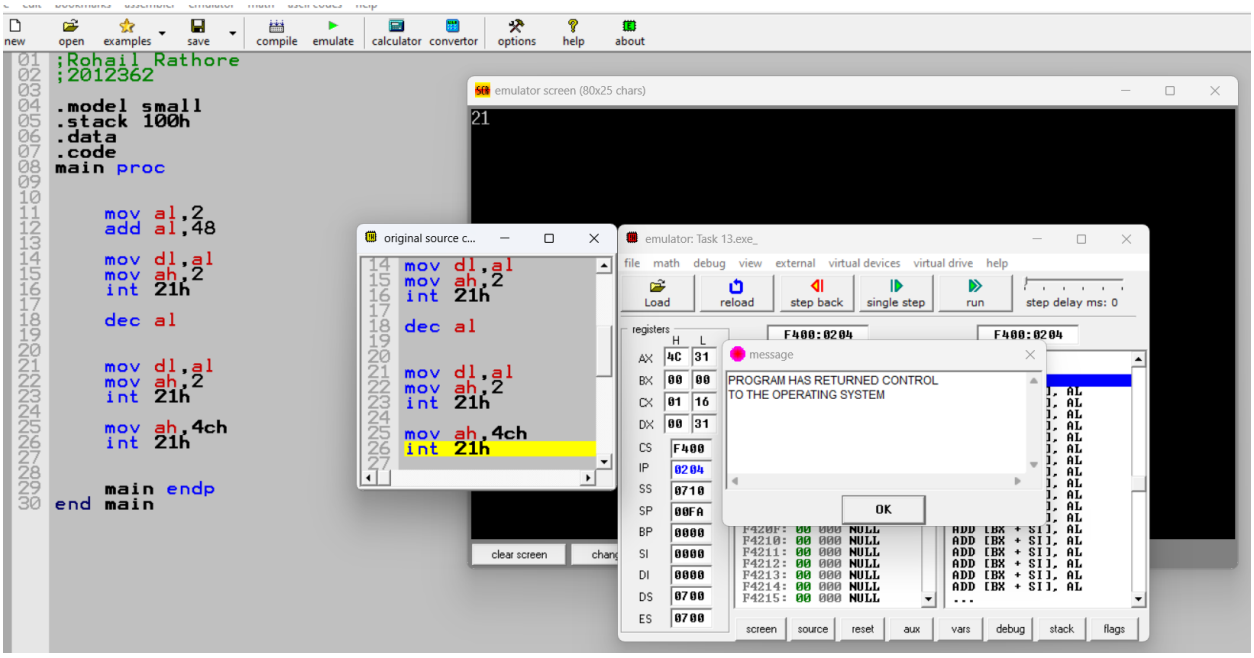
```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data
    num1 dw 1
    num2 dw 2
    num3 dw 3
    sum  dw ?
.code
main proc

        mov ax,@data
        mov ds,ax

        mov ax,[num1]
        add ax,[num2]
        add ax,[num3]

        mov [sum],ax

        mov dx,[sum]

        mov ah,2
        add dl,48
        int 21h

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

Task-12:

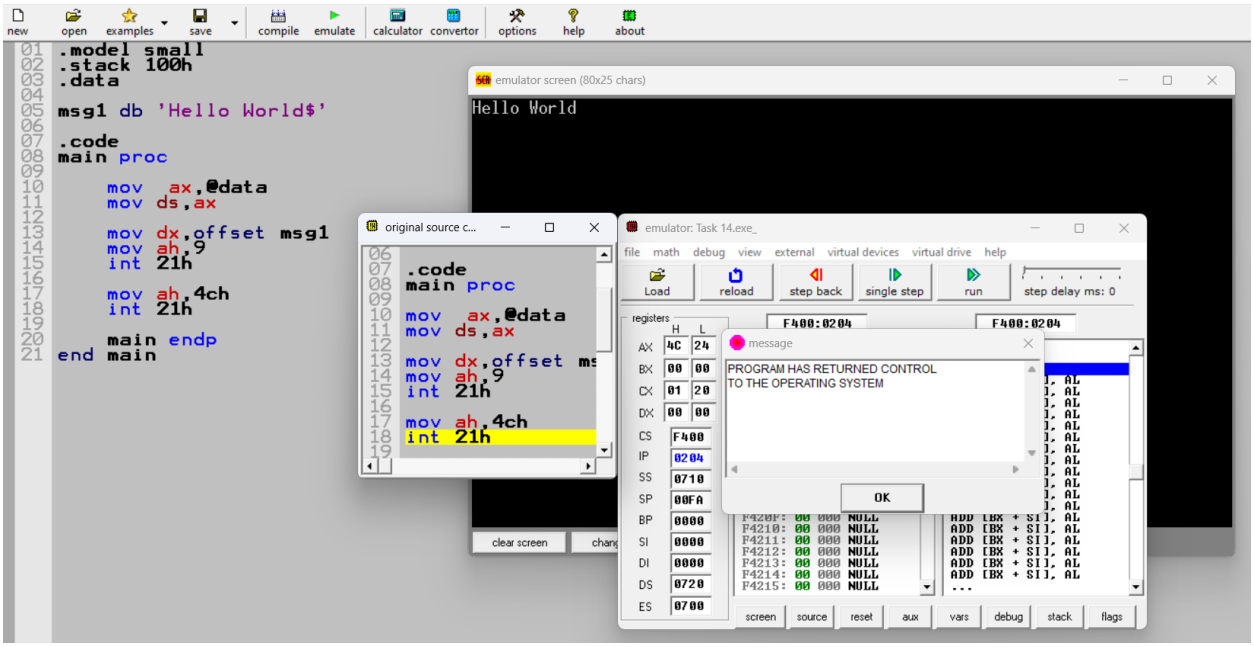


```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data
.code
main proc


    mov al,1
    add al,48

    mov dl,al
    mov ah,2
    int 21h

    inc al


    mov dl,al
    mov ah,2
    int 21h

    mov ah,4ch
    int 21h


    main endp
end main
```

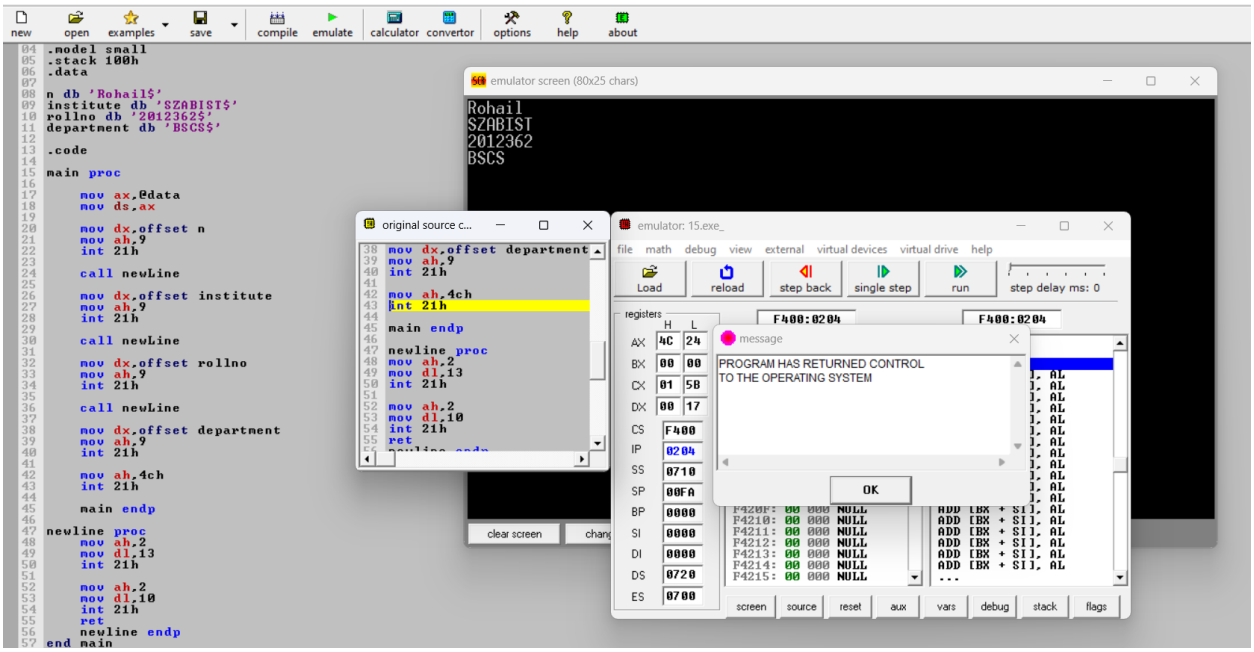
Task-13:



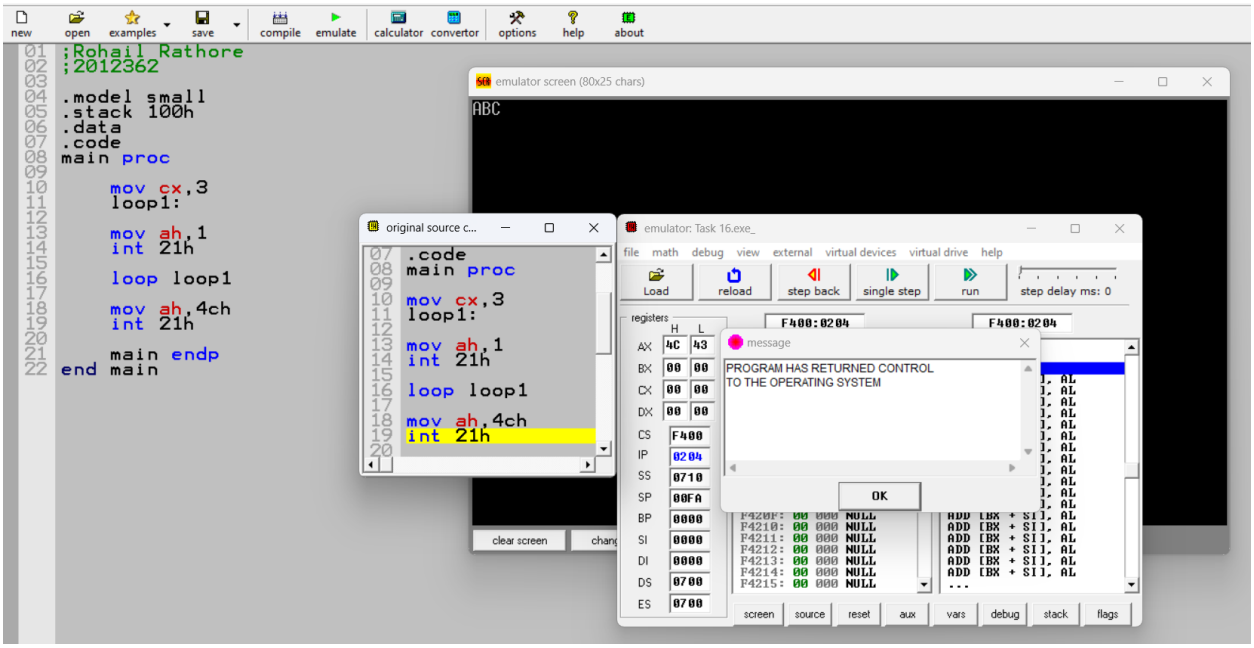
Task-14:



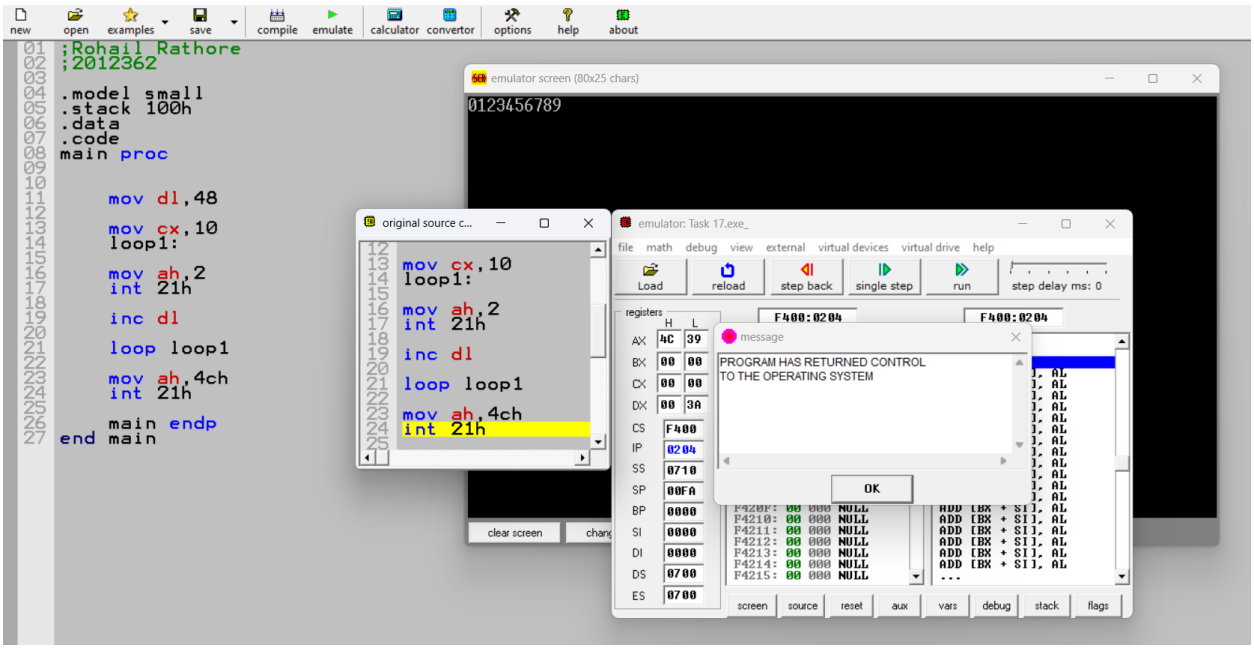
Task-15:



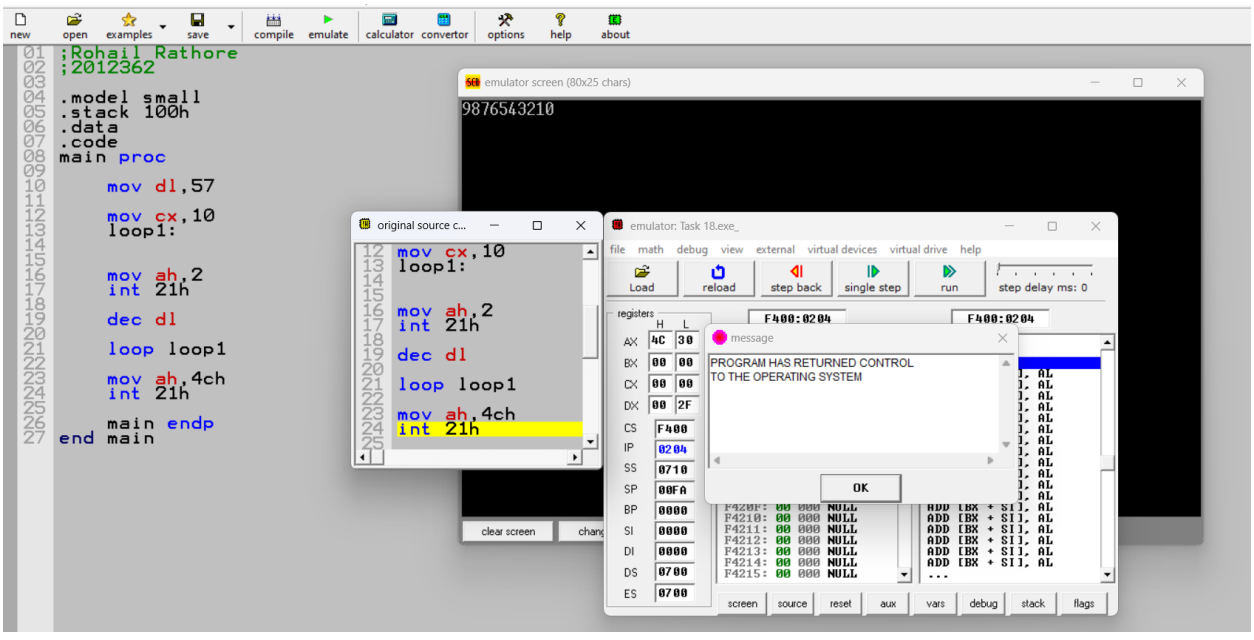
Task-16:



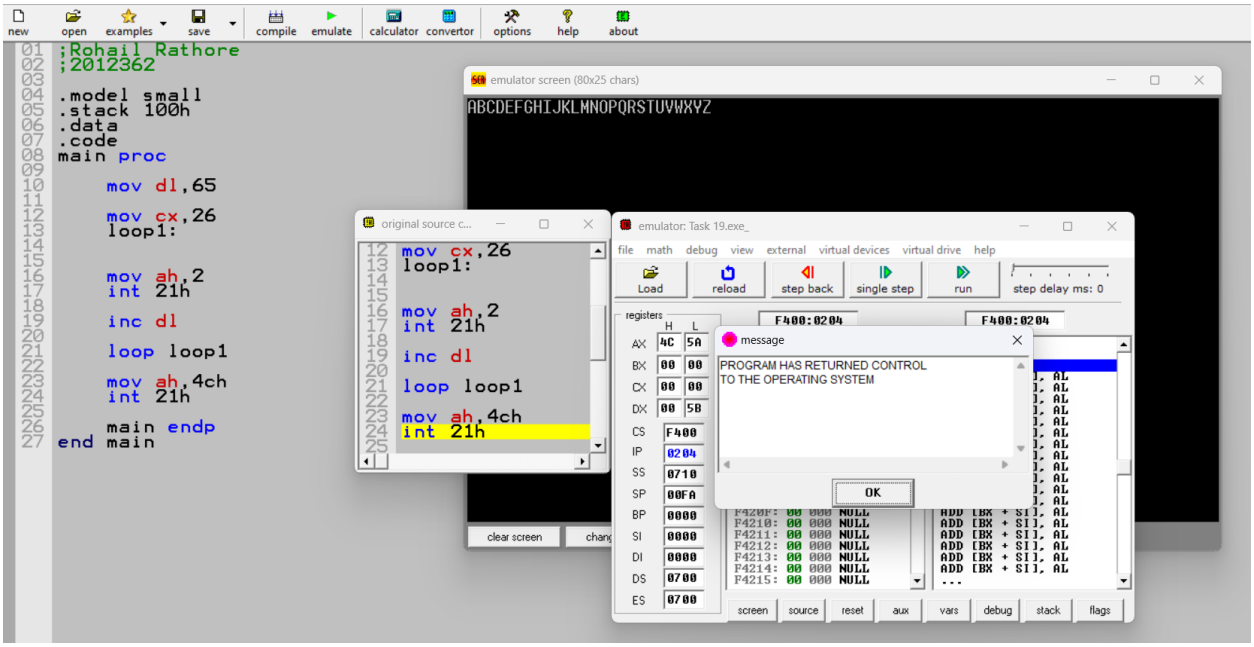
Task-17:



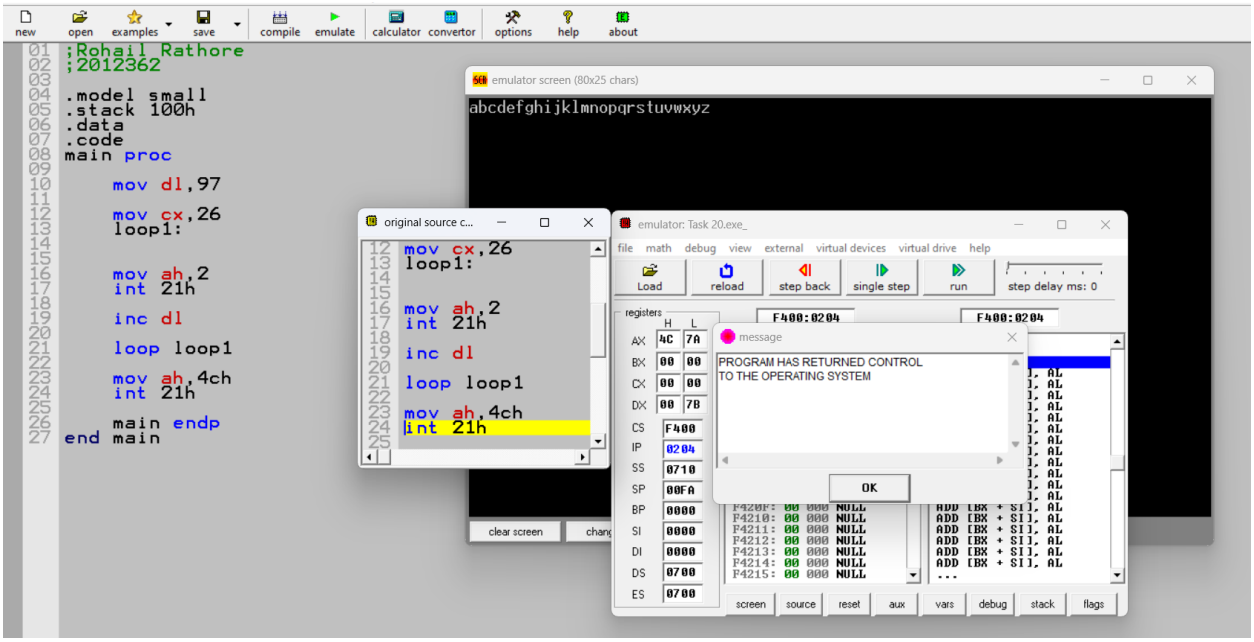
Task-18:



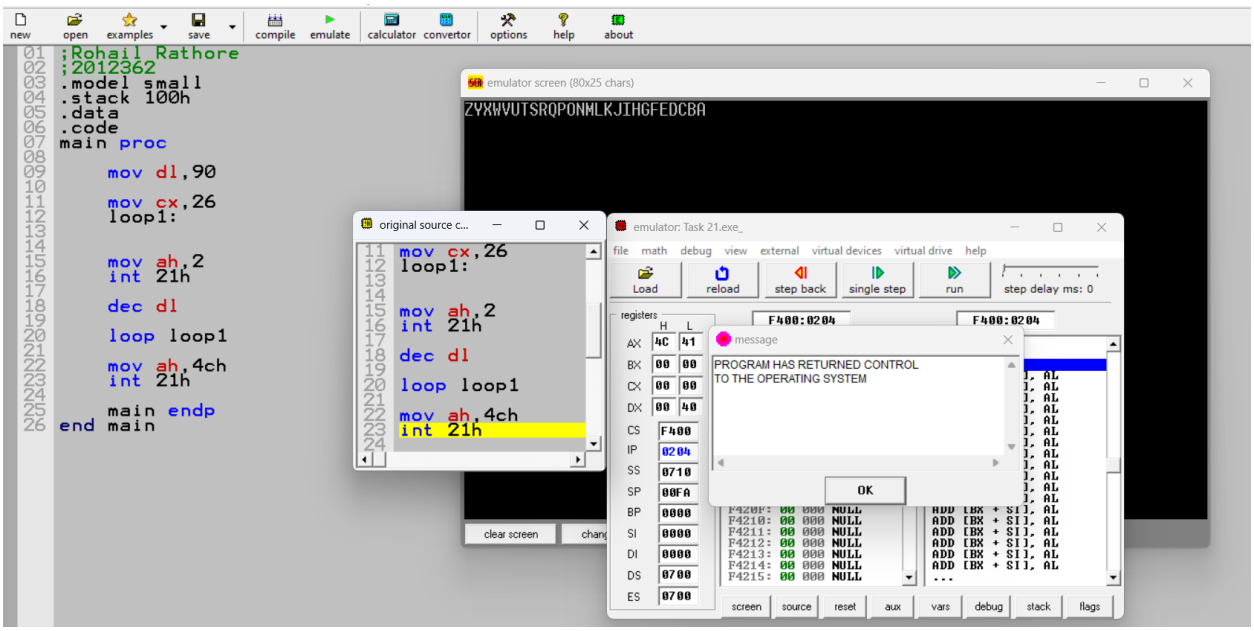
Task-19:



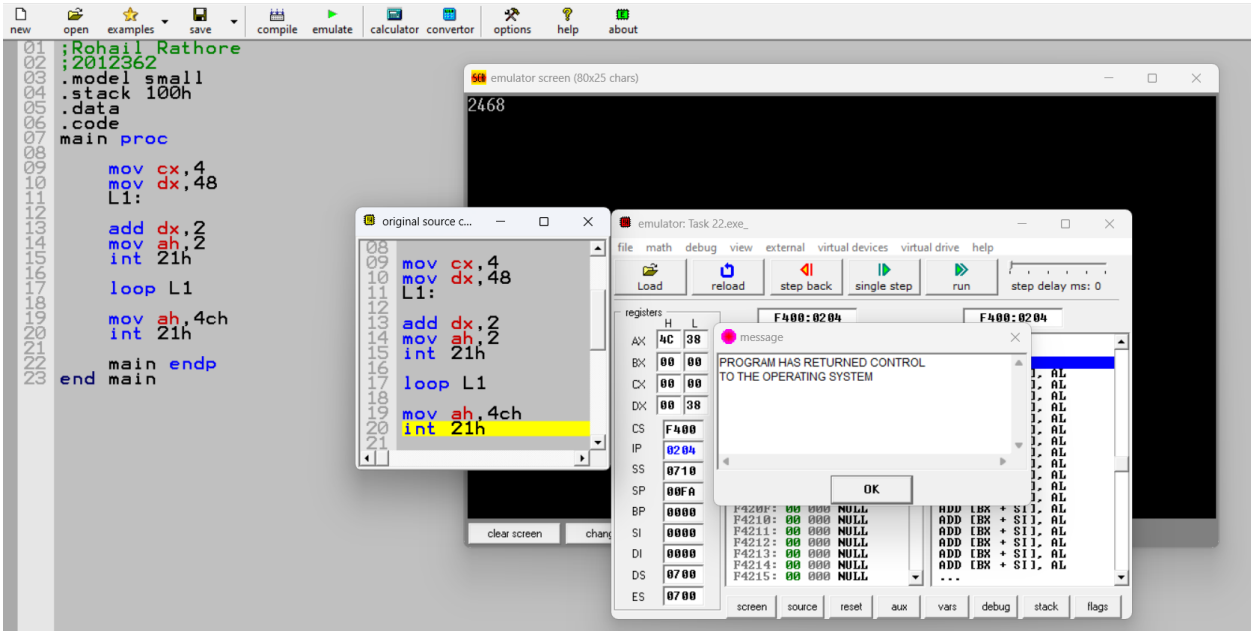
Task-20:



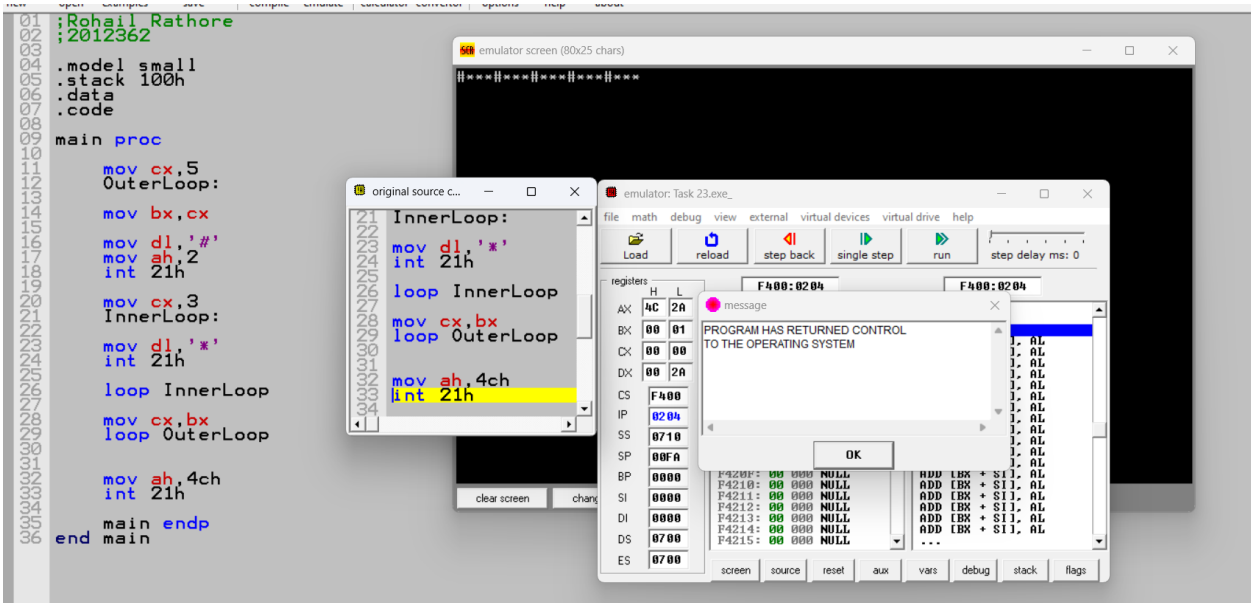
Task-21:



Task-22:



Task-23:



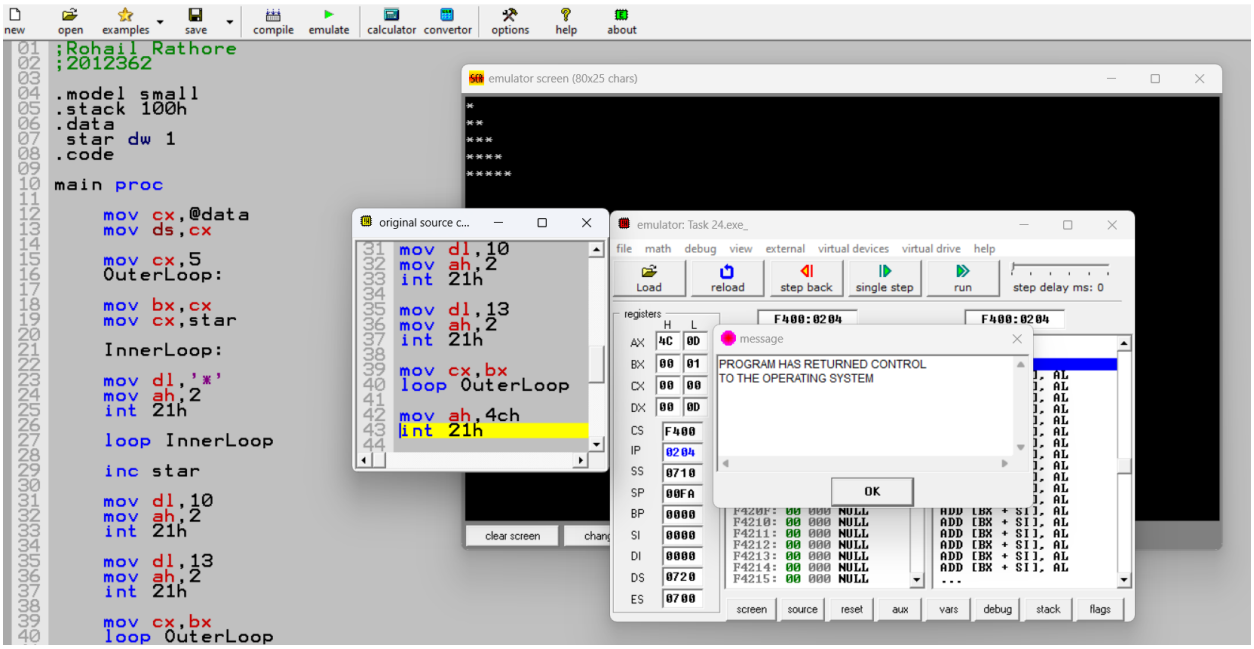
```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data
.code

main proc

    mov cx,5
    OuterLoop:

    mov bx,cx

    mov dl,'#'
    mov ah,2
    int 21h

    mov cx,3
    InnerLoop:

    mov dl,'*'
    int 21h

    loop InnerLoop

    mov cx,bx
    loop OuterLoop


    mov ah,4ch
    int 21h

    main endp
end main
```

```
InnerLoop:

mov dl,'*'
int 21h

loop InnerLoop

mov cx,bx
loop OuterLoop


mov ah,4ch
int 21h
```

PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM

OK

TAsk-24:



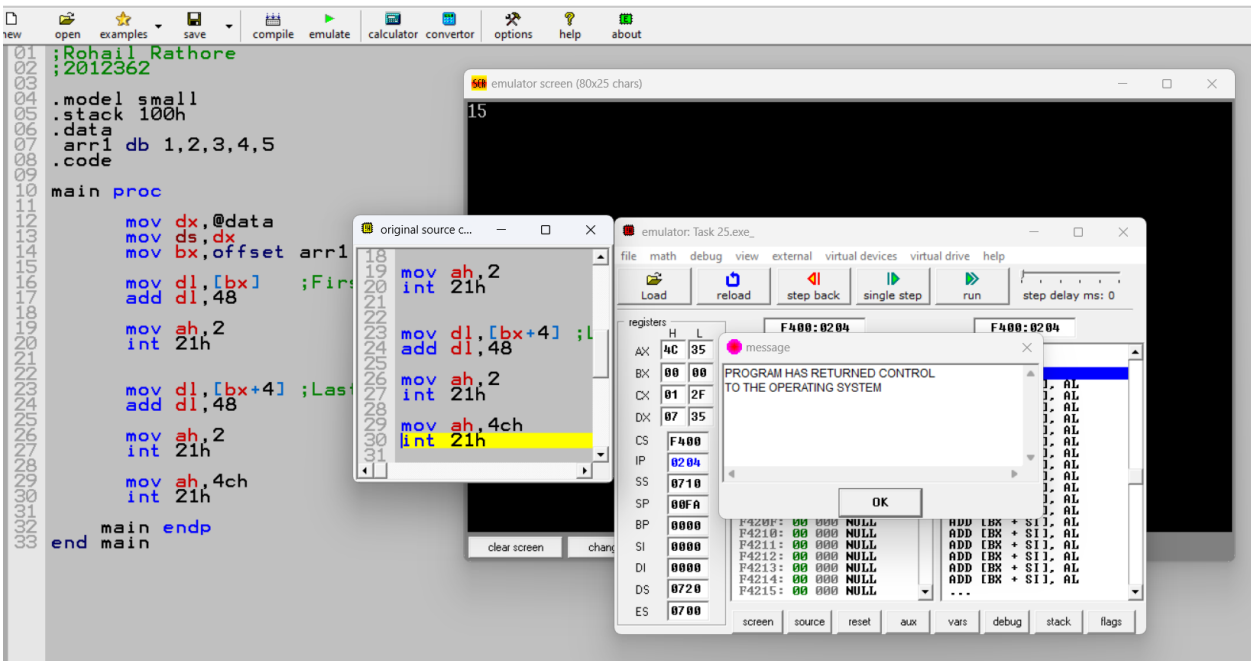
```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data
 star dw 1
.code

main proc

    mov cx,@data
    mov ds,cx

    mov cx,5
    OuterLoop:

    mov bx,cx
    mov cx,star

    InnerLoop:

    mov dl,'*'
    mov ah,2
    int 21h

    loop InnerLoop

    inc star

    mov dl,10
    mov ah,2
    int 21h

    mov dl,13
    mov ah,2
    int 21h

    mov cx,bx
    loop OuterLoop
```

```
mov dl,10
mov ah,2
int 21h

mov dl,13
mov ah,2
int 21h

mov cx,bx
loop OuterLoop

mov ah,4ch
int 21h
```

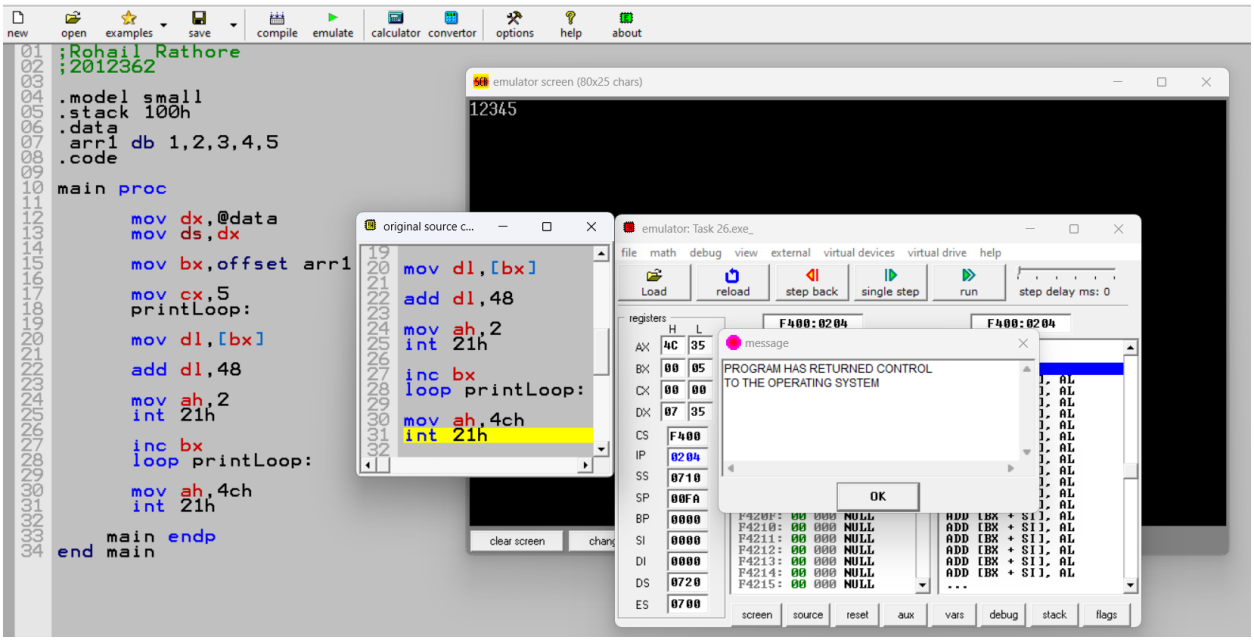
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM

OK

Task-25:



Task-26:



Task-27:

```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data

arr db 20 dup('$')

.code
main proc
    mov ax,@data
    mov ds,ax

    mov bx, offset arr
    inputLoop:

    mov ah,1
    int 21h

    cmp al,13
    je exit

    mov [bx],al
    inc bx

    jmp inputLoop

    exit:
    mov ah,4ch
    int 21h

    main endp
end main
```

```
Hello Wordl
```

```
int 21h

cmp al,13
je exit

mov [bx],al
inc bx

jmp inputLoop

exit:
mov ah,4ch
int 21h
```

PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM

OK

Task-28:

```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data

.code
main proc

    mov ah,1
    int 21h
    push ax

    mov ah,1
    int 21h
    push ax

    mov dl,10
    mov ah,2
    int 21h

    mov dl,13
    mov ah,2
    int 21h

    pop dx
    mov ah,2
    int 21h

    pop dx
    mov ah,2
    int 21h

    mov ah,4ch
    int 21h

    main endp
end main
```

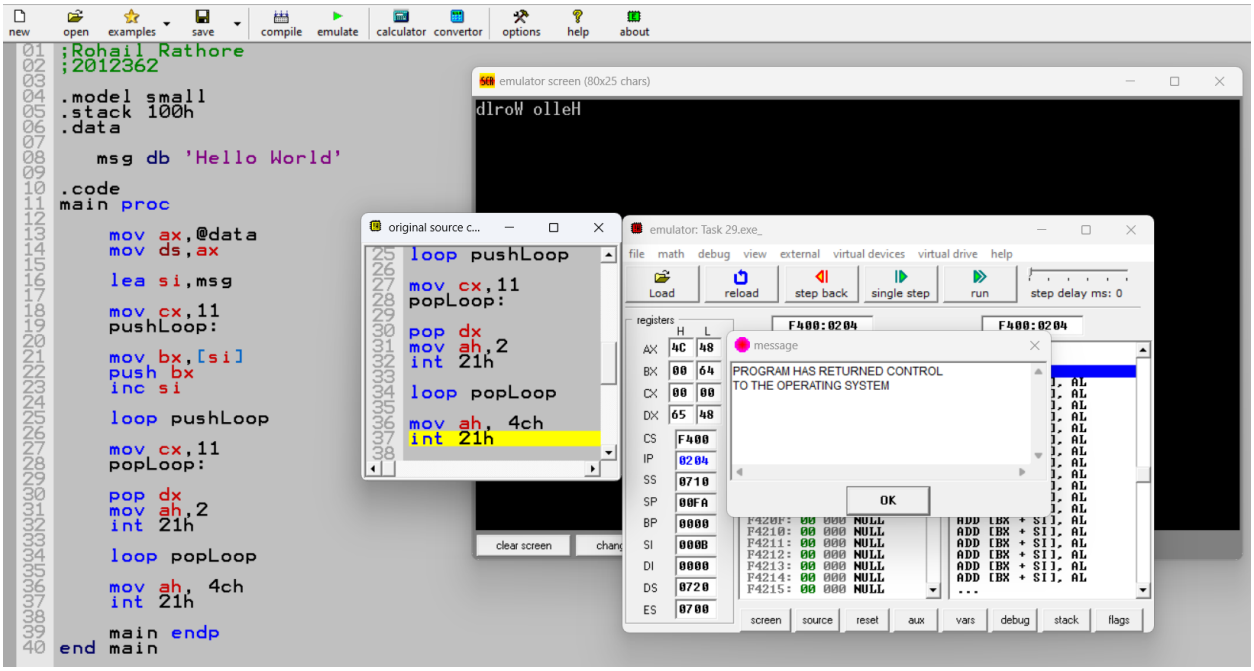
```
73
37
```

```
mov ah,2
int 21h

pop dx
mov ah,2
int 21h

pop dx
mov ah,2
int 21h

mov ah,4ch
int 21h
```

PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM

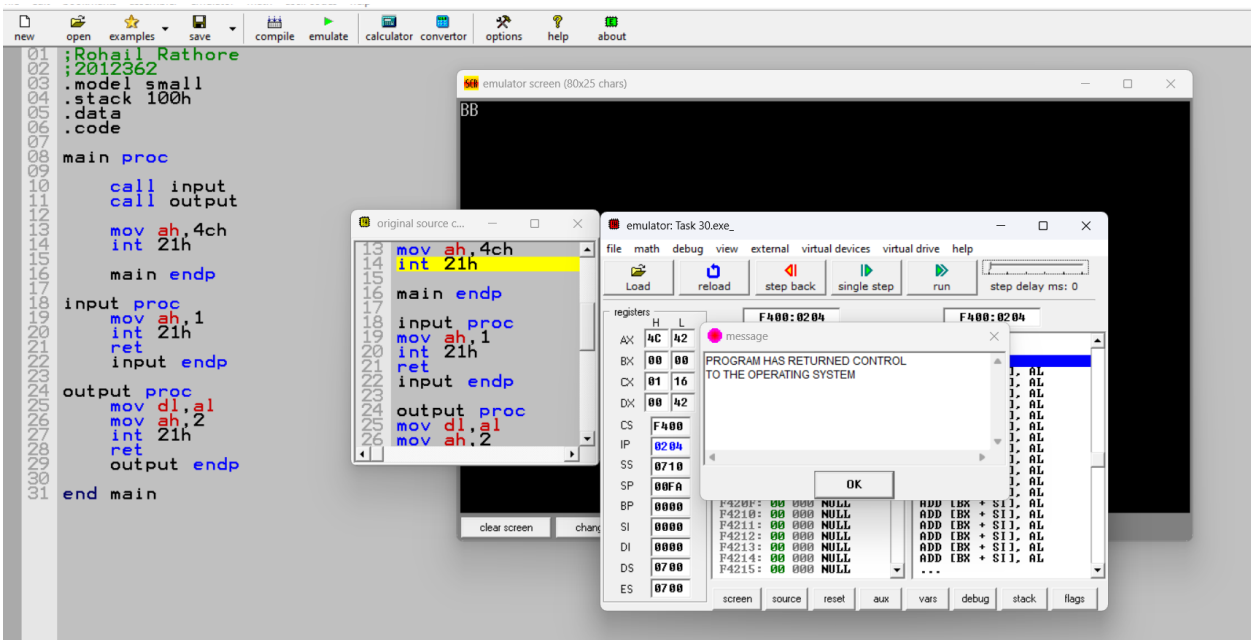
OK

Task-29:



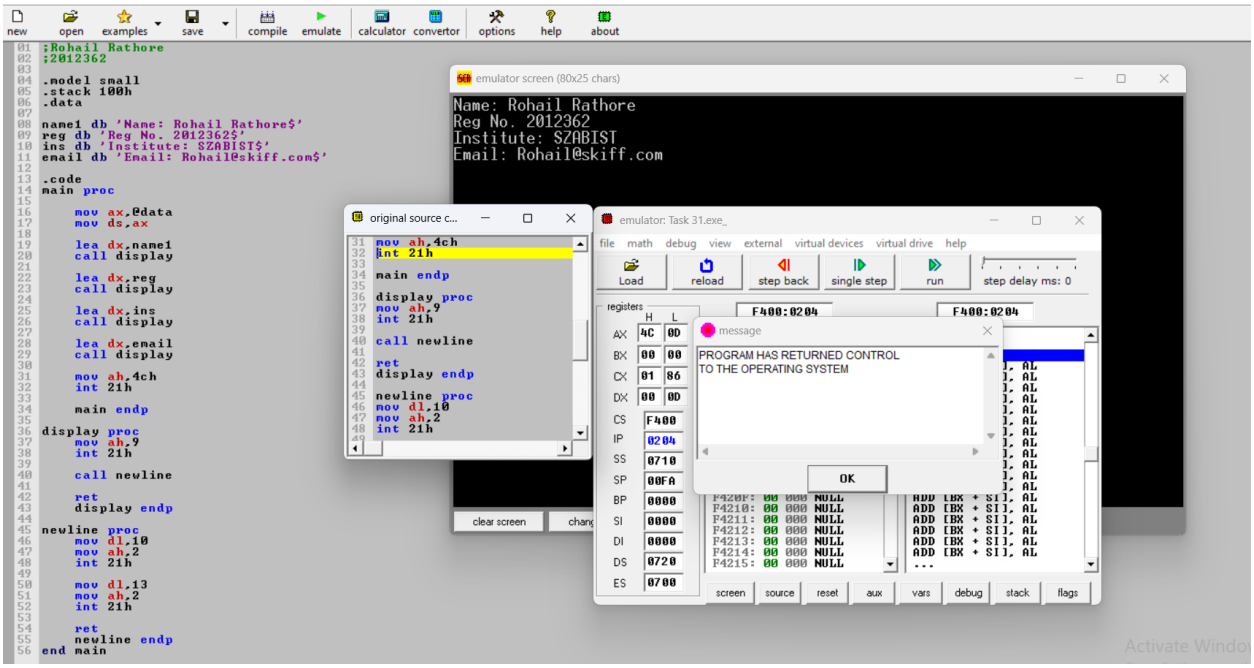
```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data

   msg db 'Hello World'

.code
main proc

    mov ax,@data
    mov ds,ax

    lea si,msg

    mov cx,11
    pushLoop:

    mov bx,[si]
    push bx
    inc si

    loop pushLoop

    mov cx,11
    popLoop:

    pop dx
    mov ah,2
    int 21h

    loop popLoop

    mov ah, 4ch
    int 21h

    main endp
end main
```


Task-30:



```
;Rohail Rathore
;2012362
.model small
.stack 100h
.data
.code

main proc

    call input
    call output

    mov ah,4ch
    int 21h

    main endp

input proc
    mov ah,1
    int 21h
    ret
    input endp

output proc
    mov dl,al
    mov ah,2
    int 21h
    ret
    output endp

end main
```

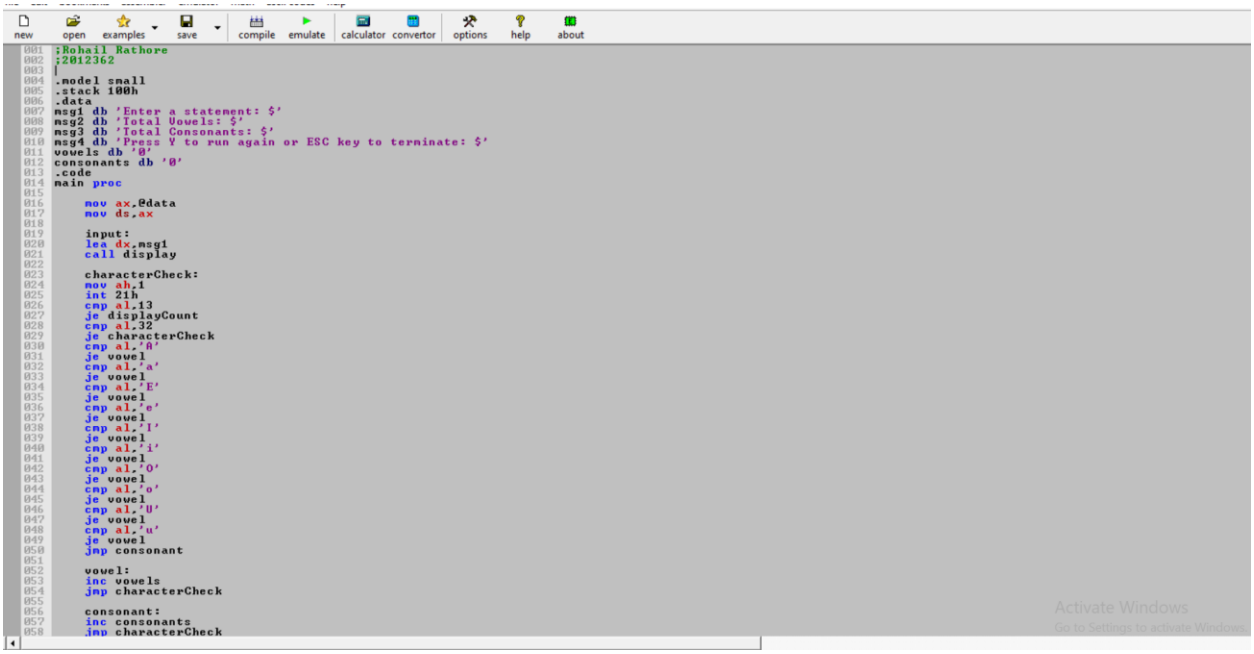

Task-31:



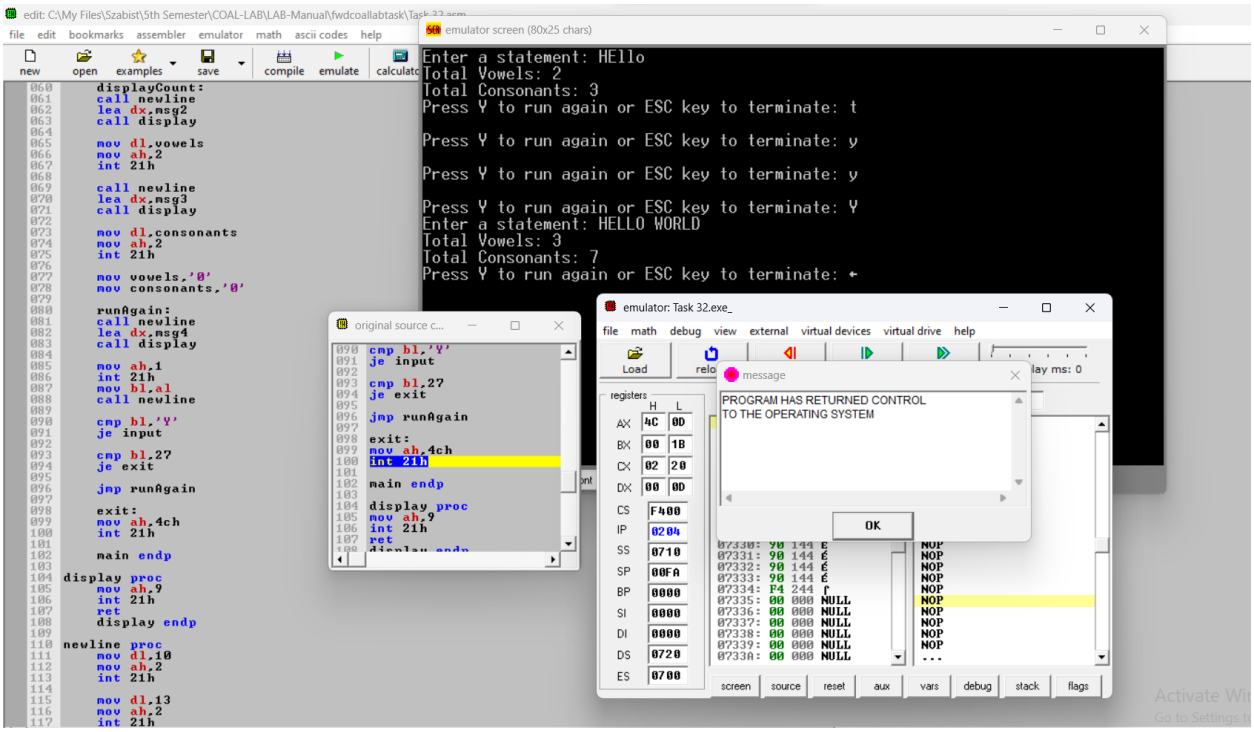
```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data

name1 db 'Name: Rohail Rathore$'
reg db 'Reg No. 2012362$'
ins db 'Institute: SZABIST$'
email db 'Email: Rohail@skiff.com$'

.code
main proc

    mov ax,@data
    mov ds,ax

    lea dx,name1
    call display

    lea dx,reg
    call display

    lea dx,ins
    call display

    lea dx,email
    call display

    mov ah,4ch
    int 21h

    main endp

display proc
    mov ah,9
    int 21h

    call newline

    ret
    display endp

newline proc
    mov dl,10
    mov ah,2
    int 21h

    mov dl,13
    mov ah,2
    int 21h

    ret
    newline endp
end main
```

```
Name: Rohail Rathore
Reg No. 2012362
Institute: SZABIST
Email: Rohail@skiff.com
```

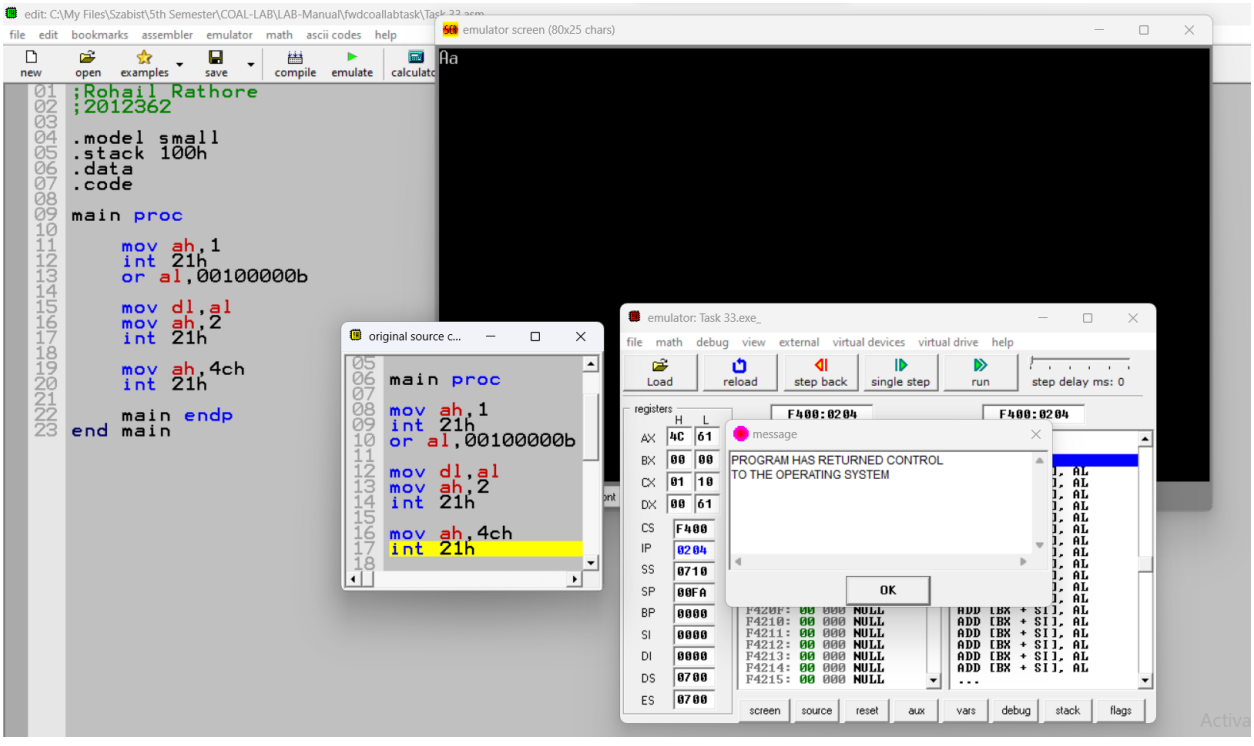
Task-32:



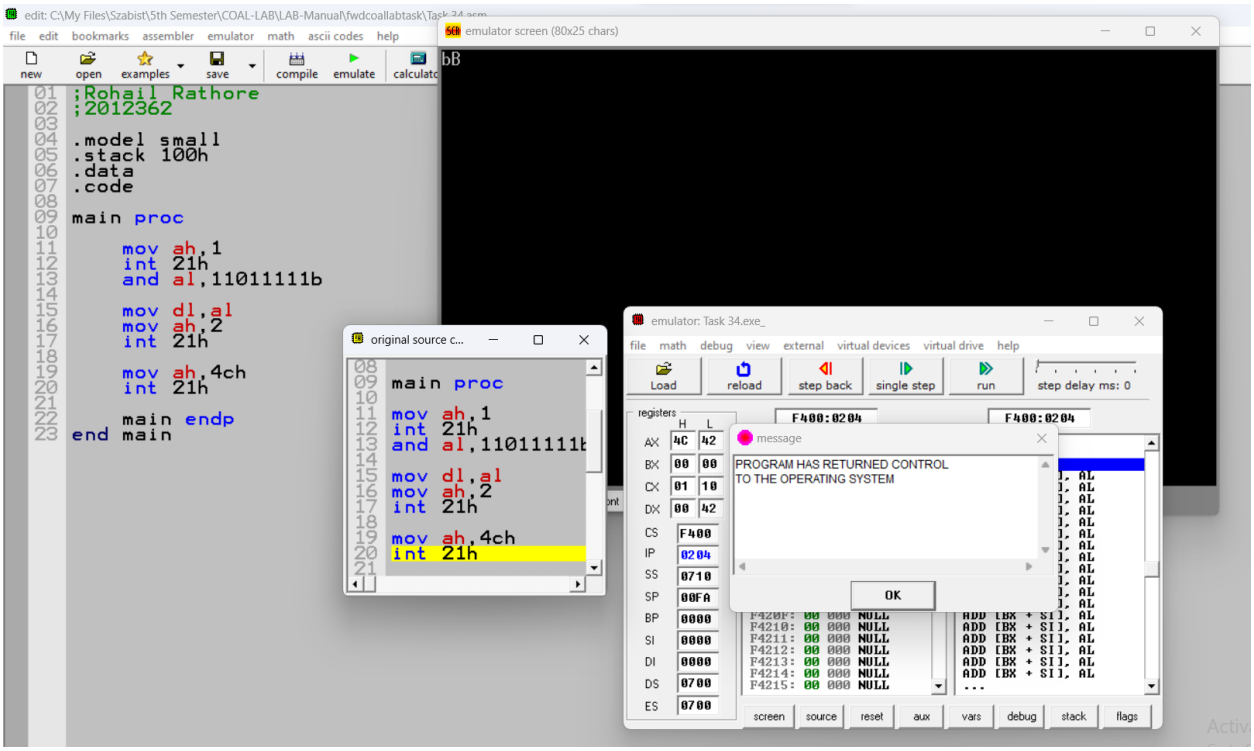
```
;Rohail Rathore
;2012362

.model small
.stack 100h
.data
msg1 db 'Enter a statement: $'
msg2 db 'Total Vowels: $'
msg3 db 'Total Consonants: $'
msg4 db 'Press Y to run again or ESC key to terminate: $'
vowels db '0'
consonants db '0'
.code
main proc

    mov ax,@data
    mov ds,ax

    input:
    lea dx,msg1
    call display

    characterCheck:
    mov ah,1
    int 21h
    cmp al,13
    je displayCount
    cmp al,32
    je characterCheck
    cmp al,'A'
    je vowel
    cmp al,'a'
    je vowel
    cmp al,'E'
    je vowel
    cmp al,'e'
    je vowel
    cmp al,'I'
    je vowel
    cmp al,'i'
    je vowel
    cmp al,'O'
    je vowel
    cmp al,'o'
    je vowel
    cmp al,'U'
    je vowel
    cmp al,'u'
    je vowel
    jmp consonant

    vowel:
    inc vowels
    jmp characterCheck

    consonant:
    inc consonants
    jmp characterCheck
```



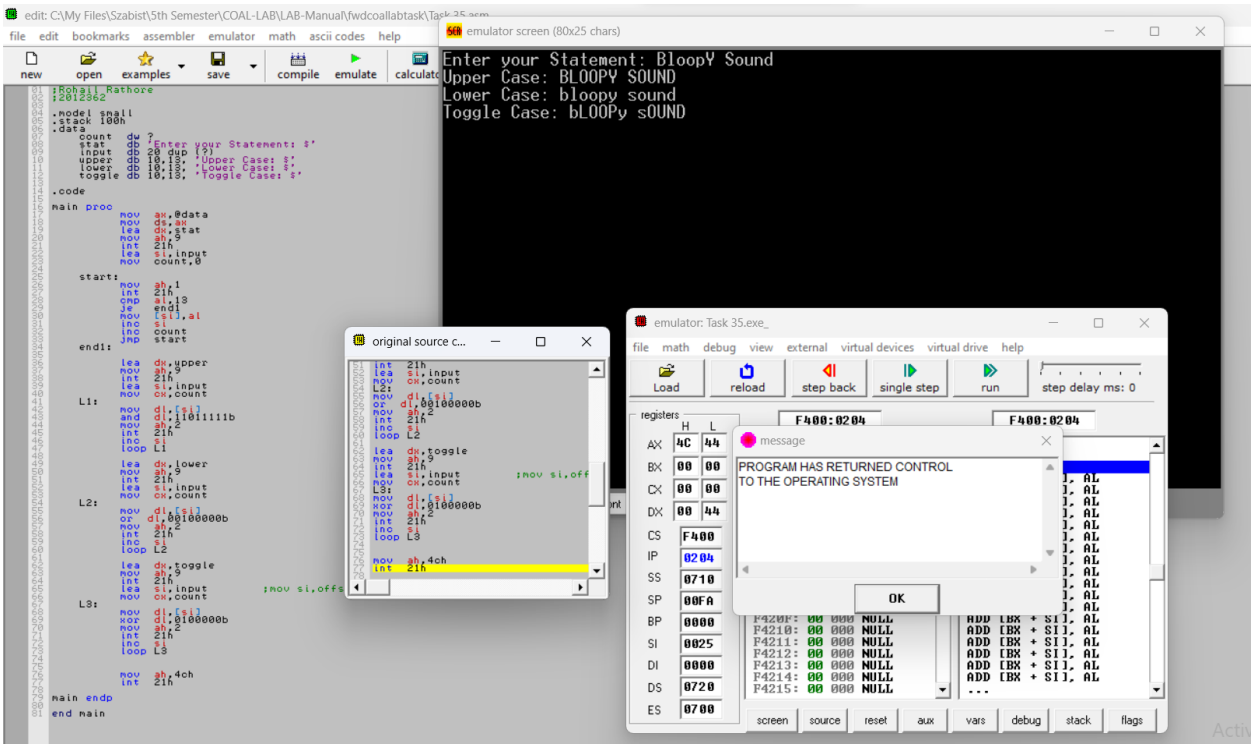
Task-33:



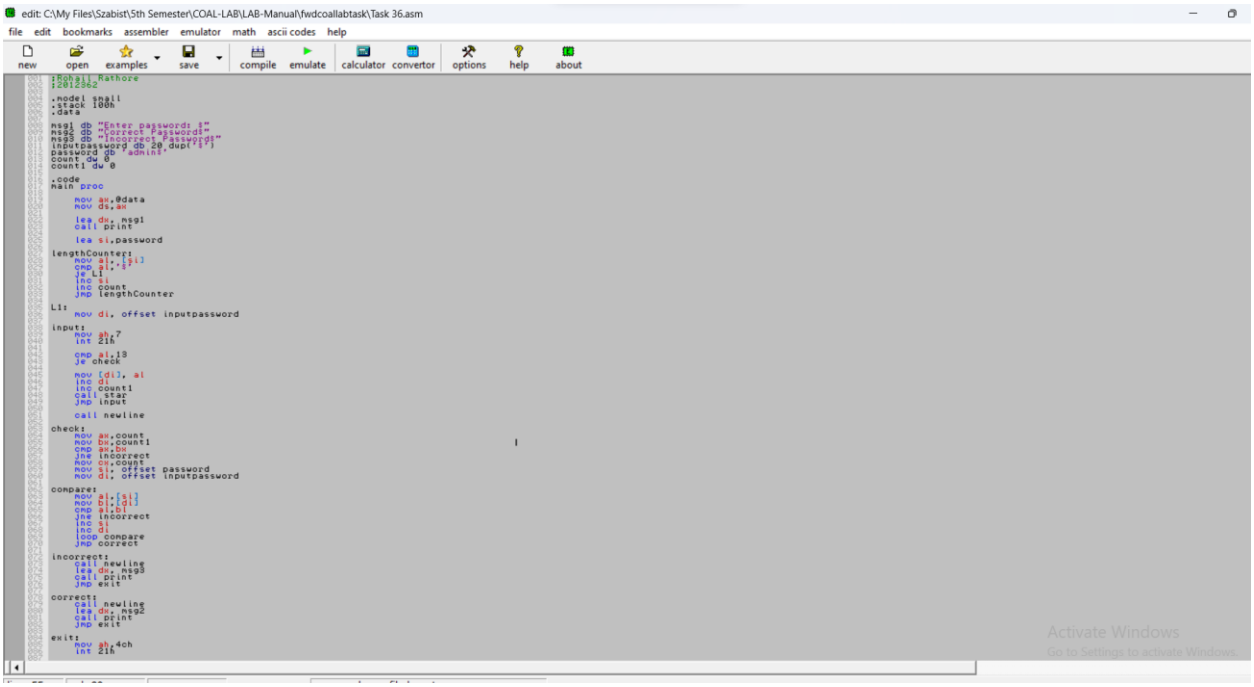
Task-34:

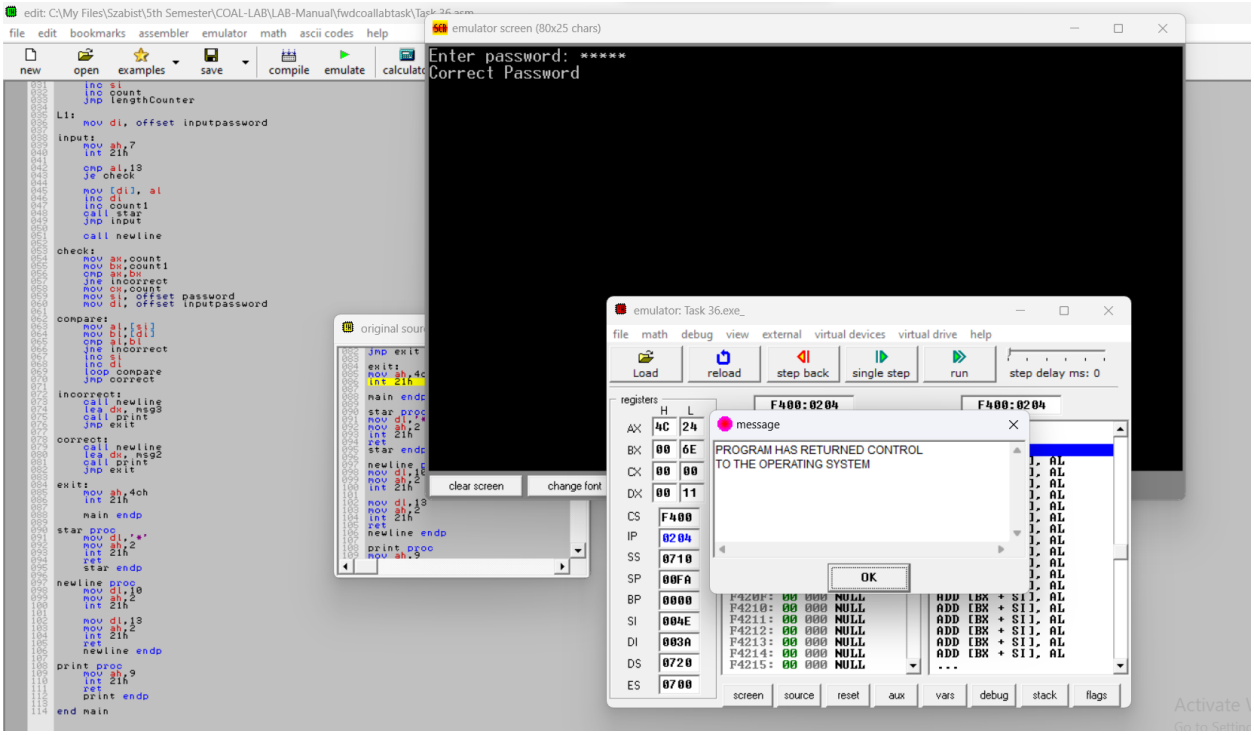


Task-35:

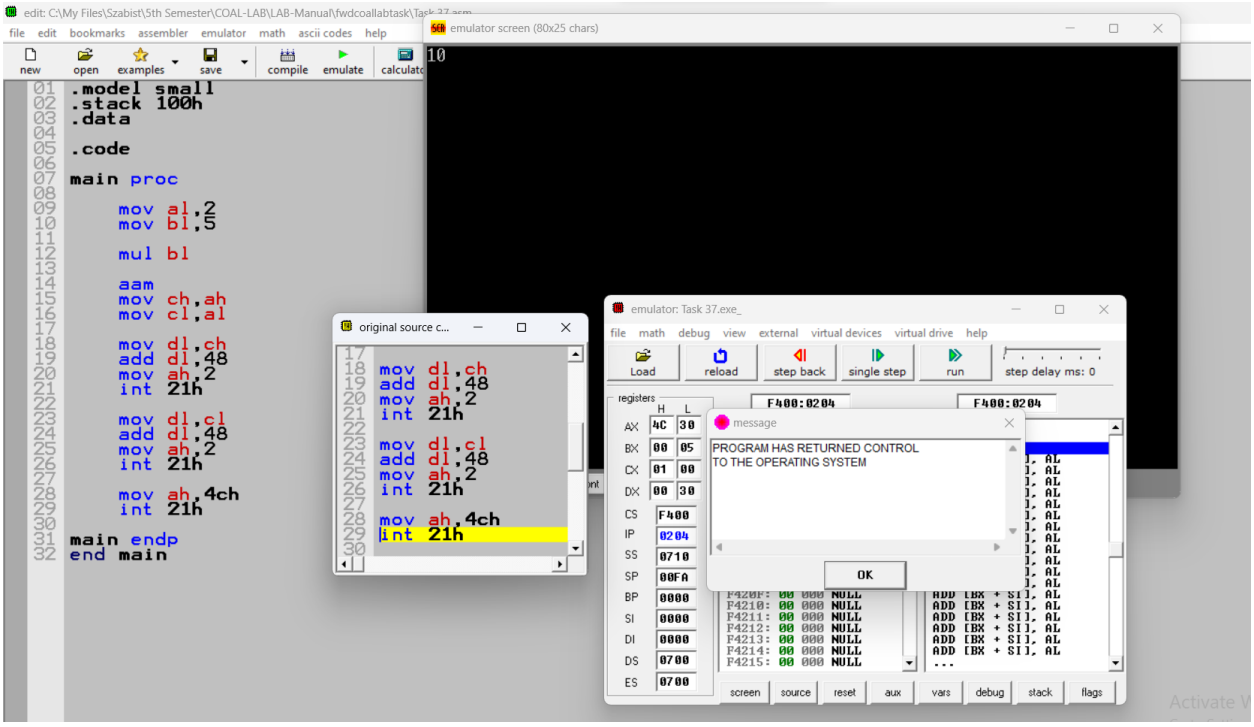


Task-36:


edit: C:\My Files\Szabist\5th Semester\COAL-LAB\LAB-Manual\fwdcoallabtask\Task 36.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
;Rahul Rathore
;2012262
.model small
.stack 100h
.data
msg1 db "Enter password: $"
msg2 db "Correct Password"
msg3 db "Incorrect Password$"
inputpassword db 20 dup('$')
password db 'admin$'
count dw 0
count1 dw 0
.code
main proc
mov ax,@data
mov ds,ax
lea dx, msg1
call print
lea si,password
lengthCounter:
mov al,[si]
cmp al,'$'
je L1
inc si
inc count
jmp lengthCounter
L1:
mov di, offset inputpassword
input:
mov ah,7
int 21h
cmp al,13
je check
mov [di], al
inc di
inc count1
call star
jmp input
call newline
check:
mov ax,count
mov bx,count1
cmp ax,bx
jne incorrect
mov cx,count
mov si, offset password
mov di, offset inputpassword
compare:
mov al,[si]
mov bl,[di]
cmp al,bl
jne incorrect
inc si
inc di
loop compare
jmp correct
incorrect:
call newline
lea dx, msg3
call print
jmp exit
correct:
call newline
lea dx, msg2
call print
jmp exit
exit:
mov ah,4ch
int 21h

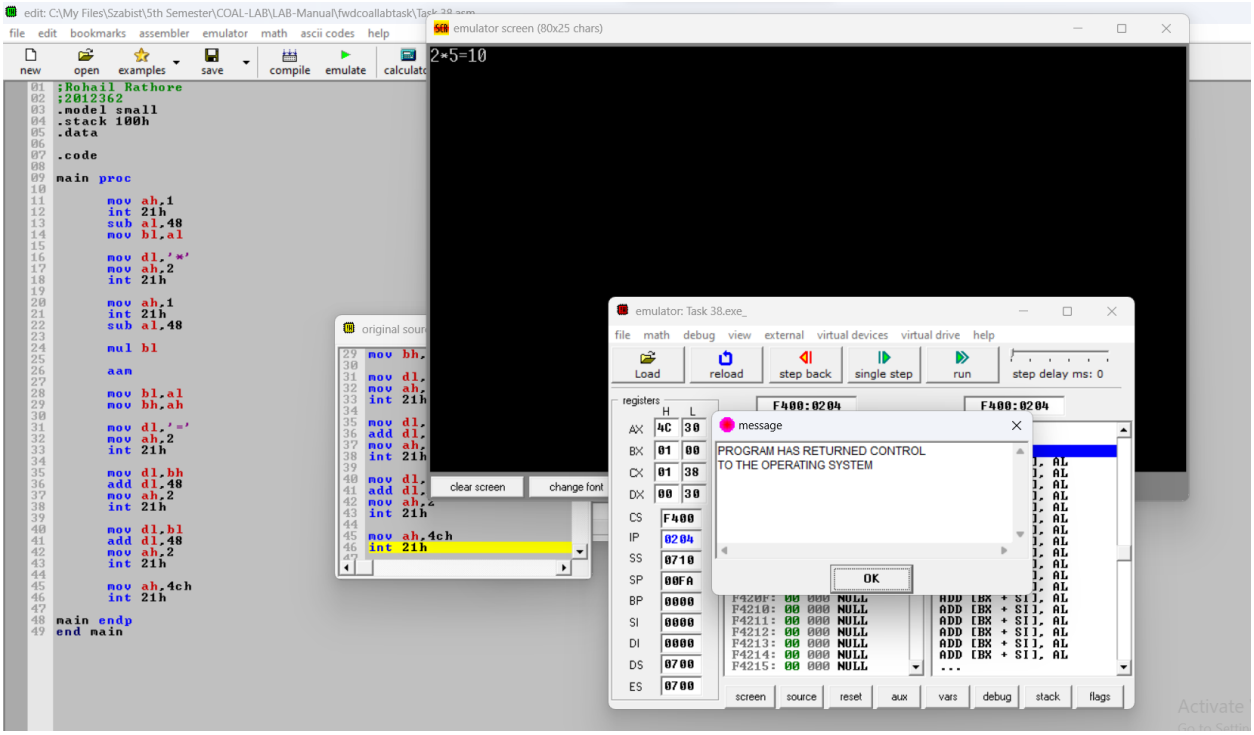

edit: C:\My Files\Szabist\5th Semester\COAL-LAB\LAB-Manual\fwdcoallabtask\Task 36.asm
emulator screen (80x25 chars)
Enter password: *****
Correct Password
clear screen
change font
emulator: Task 36.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
registers
AX 4C 24
BX 00 6E
CX 00 00
DX 00 11
CS F400
IP 0204
SS 0710
SP 00FA
BP 0000
SI 004E
DI 003A
DS 0720
ES 0700
F400:0204
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
screen source reset aux vars debug stack flags

Task-37:



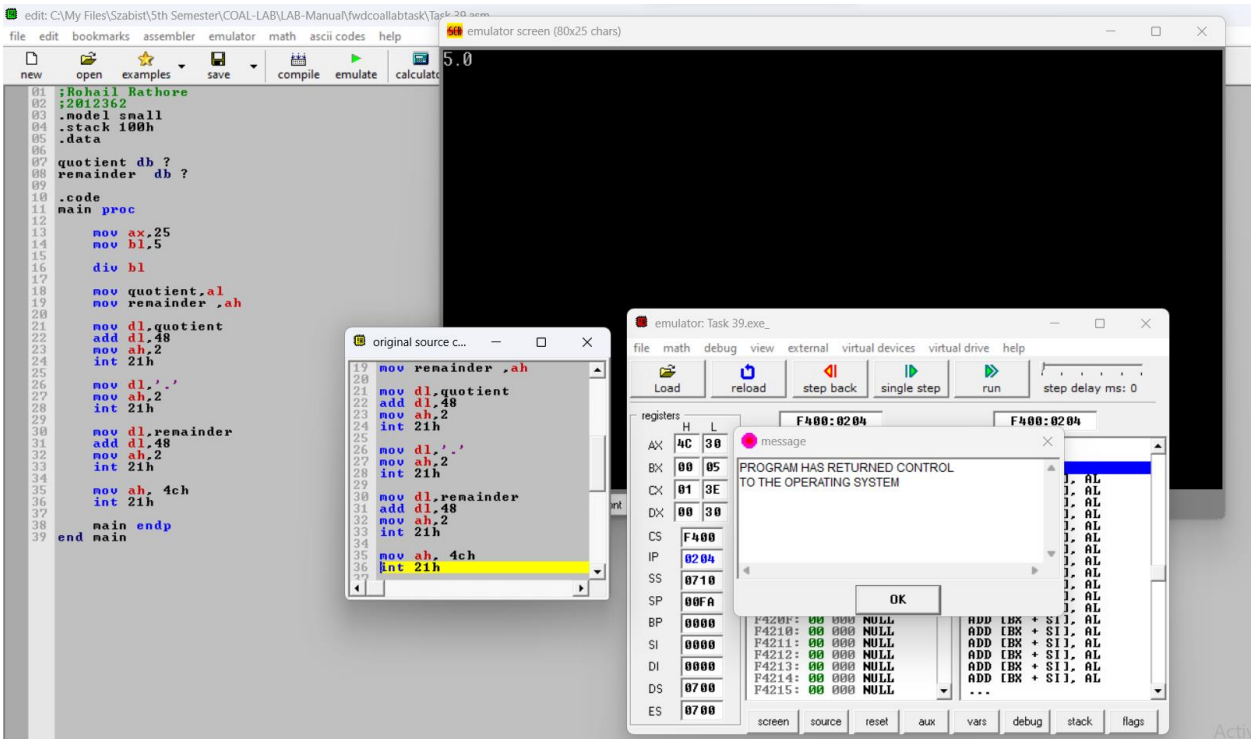

```
.model small
.stack 100h
.data

.code

main proc

      mov al,2
      mov bl,5

      mul bl

      aam
      mov ch,ah
      mov cl,al

      mov dl,ch
      add dl,48
      mov ah,2
      int 21h

      mov dl,cl
      add dl,48
      mov ah,2
      int 21h

      mov ah,4ch
      int 21h

main endp
end main
```

Task-38:

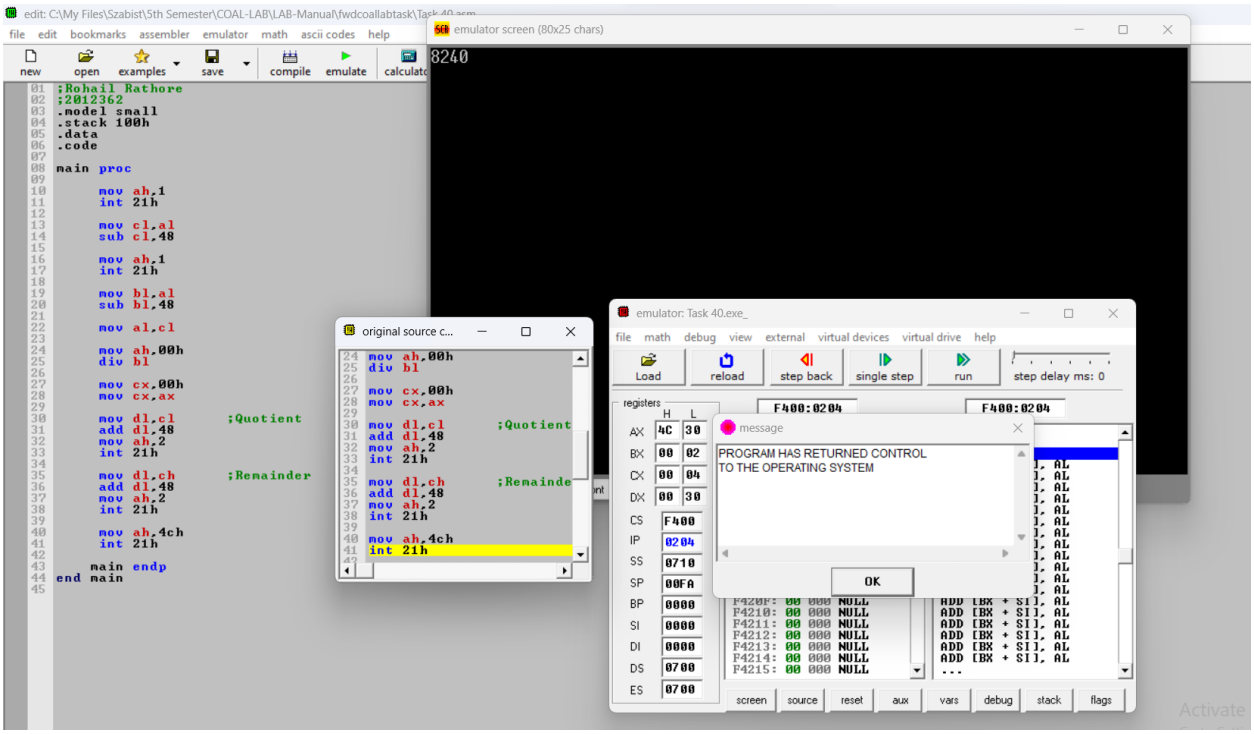



```
;Rohail Rathore
;2012362
.model small
.stack 100h
.data

.code

main proc

      mov ah,1
      int 21h
      sub al,48
      mov bl,al

      mov dl,'*'
      mov ah,2
      int 21h

      mov ah,1
      int 21h
      sub al,48

      mul bl

      aam

      mov bl,al
      mov bh,ah

      mov dl,'='
      mov ah,2
      int 21h

      mov dl,bh
      add dl,48
      mov ah,2
      int 21h

      mov dl,bl
      add dl,48
      mov ah,2
      int 21h

      mov ah,4ch
      int 21h

main endp
end main
```

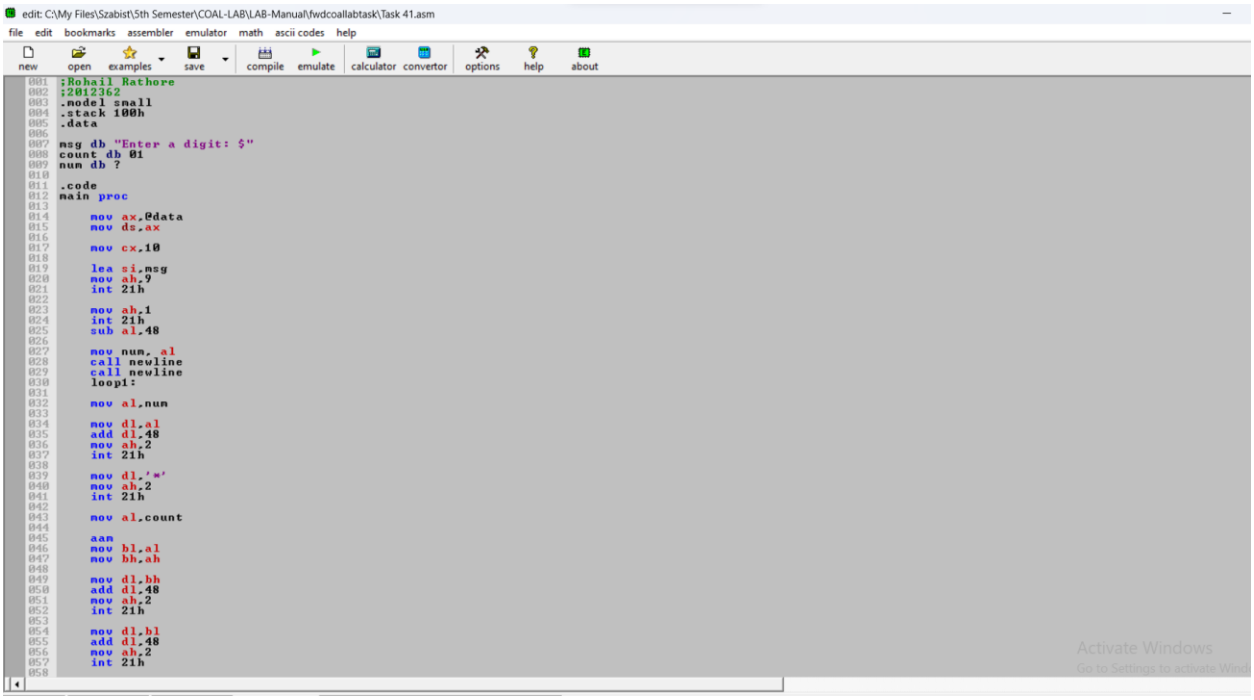
Task-39:

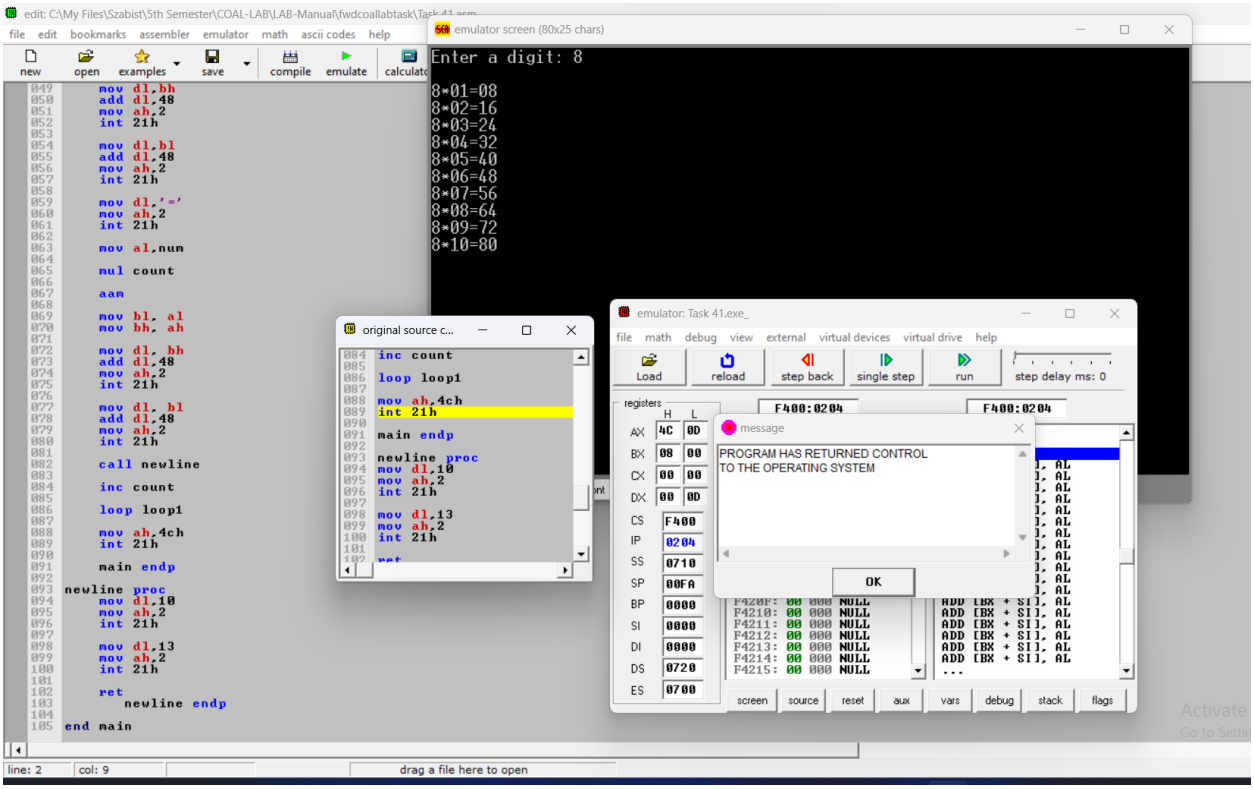


Task-40:

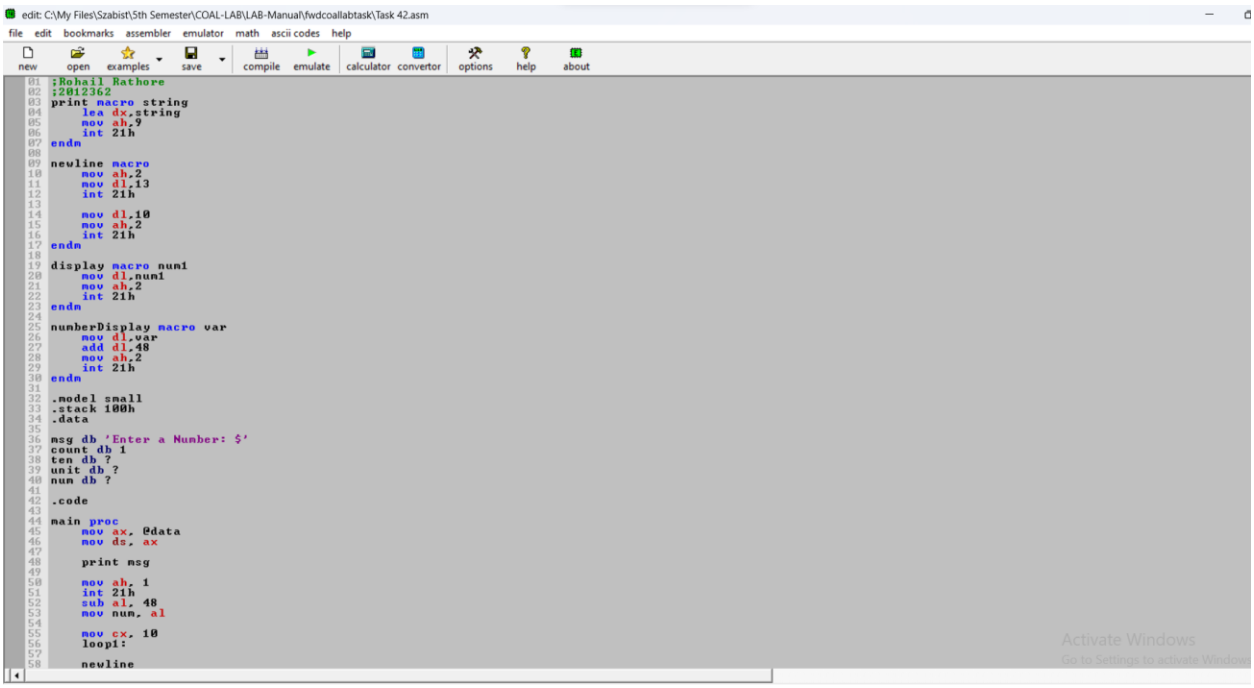


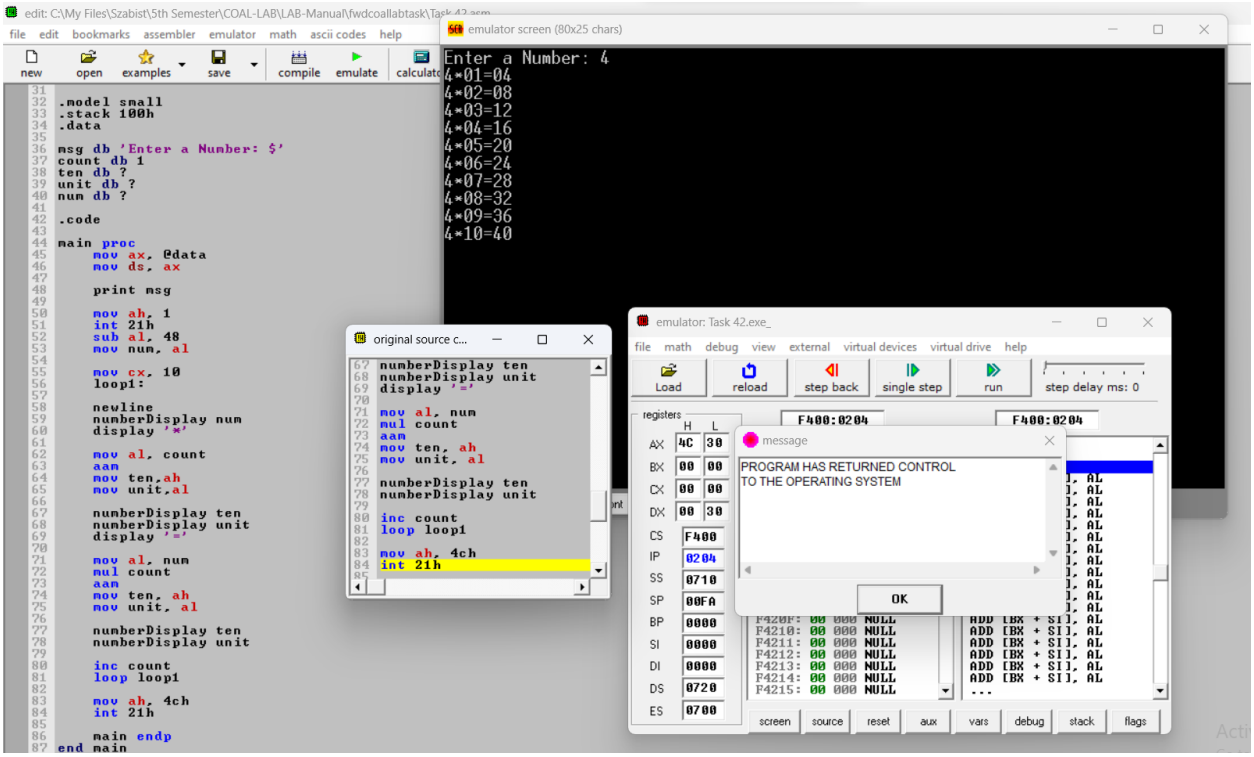
Task-41:


edit: C:\My Files\Szabist\5th Semester\COAL-LAB\LAB-Manual\fwdcoallabtask\Task 41.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
:Rohail Rathore
:20012362
.model small
.stack 100h
.data
msg db "Enter a digit: $"
count db 01
num db ?
.code
main proc
mov ax,@data
mov ds,ax
mov cx,10
lea si,msg
mov ah,9
int 21h
mov ah,1
int 21h
sub al,48
mov num, al
call newline
call newline
loop1:
mov al,num
mov dl,al
add dl,48
mov ah,2
int 21h
mov dl,'='
mov ah,2
int 21h
mov al,count
aam
mov bl,al
mov bh,ah
mov dl,bh
add dl,48
mov ah,2
int 21h
mov dl,bl
add dl,48
mov ah,2
int 21h

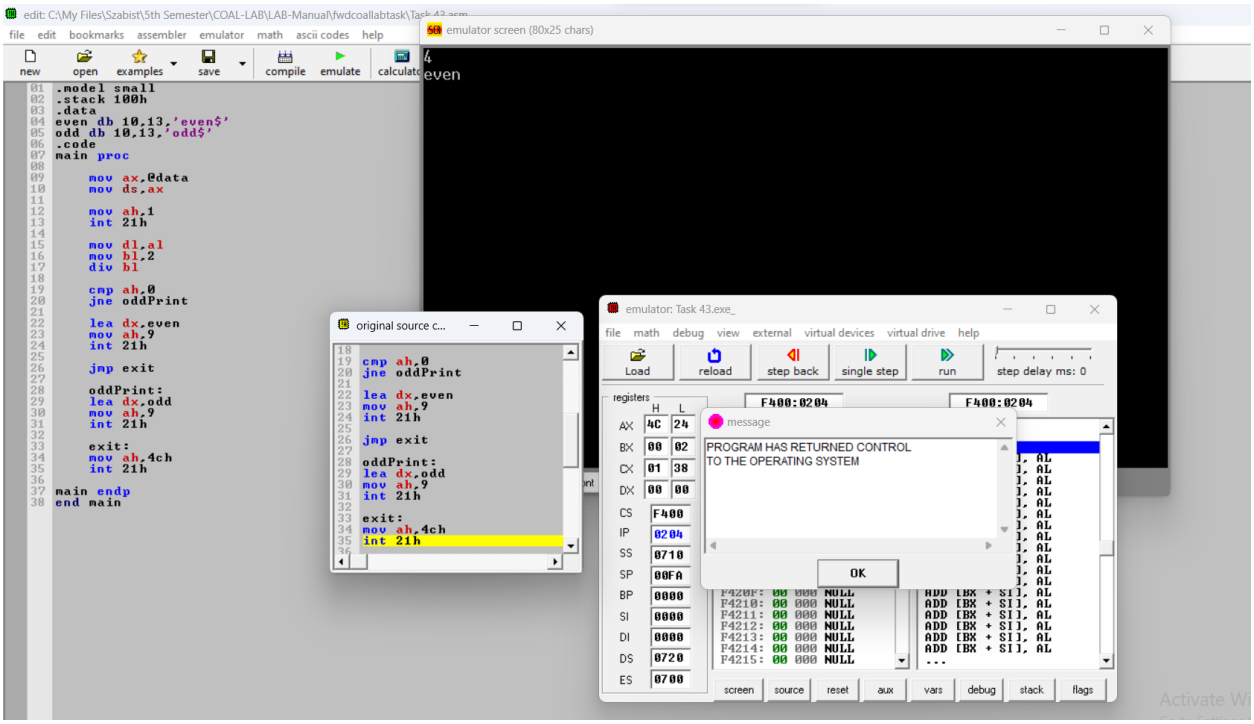

edit: C:\My Files\Szabist\5th Semester\COAL-LAB\LAB-Manual\fwdcoallabtask\Task 41.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate
emulator screen (80x25 chars)
Enter a digit: 8
8*01=08
8*02=16
8*03=24
8*04=32
8*05=40
8*06=48
8*07=56
8*08=64
8*09=72
8*10=80
original source c...
inc count
loop loop1
mov ah,4ch
int 21h
main endp
newline proc
mov dl,10
mov ah,2
int 21h
mov dl,13
mov ah,2
int 21h
ret
emulator: Task 41.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
registers
AX 4C 0D
BX 08 00
CX 00 00
DX 00 0D
CS F400
IP 0204
SS 0710
SP 00FA
BP 0000
SI 0000
DI 0000
DS 0720
ES 0700
F400:0204
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
screen source reset aux vars debug stack flags
line: 2 col: 9
drag a file here to open

Task-42:

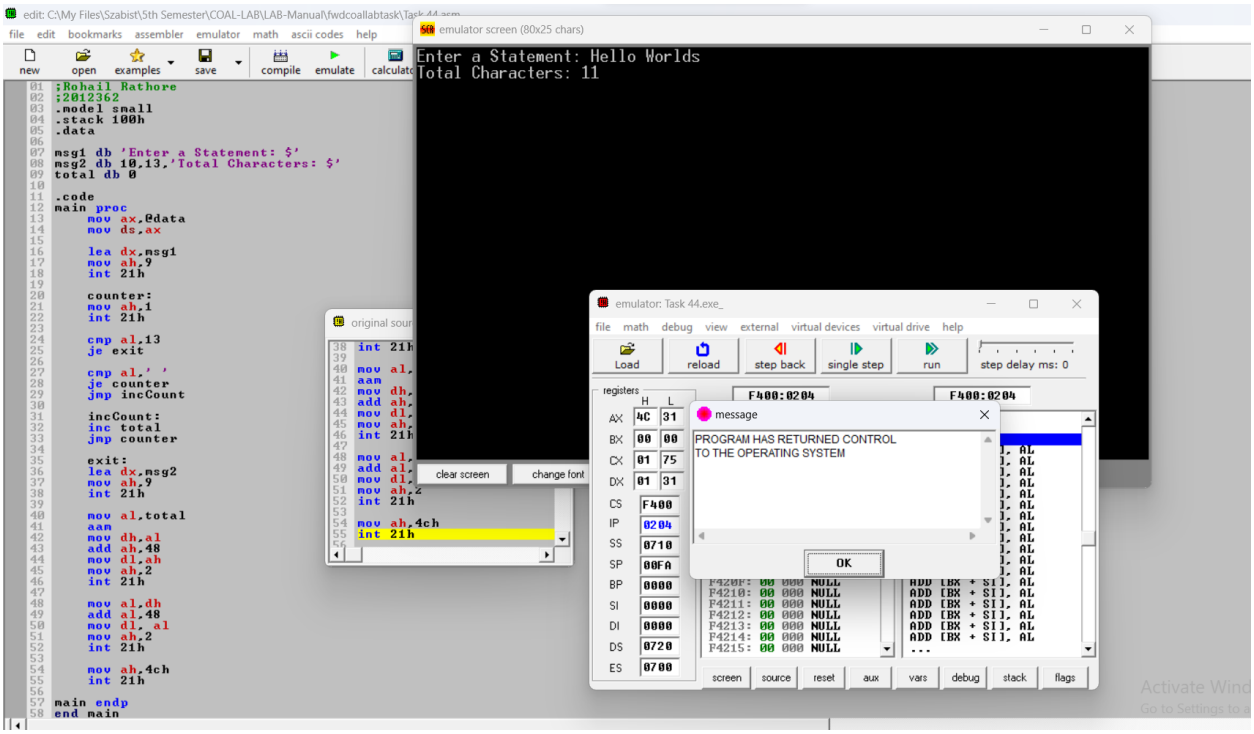

edit: C:\My Files\Szabist\5th Semester\COAL-LAB\LAB-Manual\fwdcoallabtask\Task 42.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
;Rohail Rathore
;2012362
print macro string
lea dx,string
mov ah,9
int 21h
endm
newline macro
mov ah,2
mov dl,13
int 21h
mov dl,10
mov ah,2
int 21h
endm
display macro num1
mov dl,num1
mov ah,2
int 21h
endm
numberDisplay macro var
mov dl,var
add dl,48
mov ah,2
int 21h
endm
.model small
.stack 100h
.data
msg db 'Enter a Number: $'
count db 1
ten db ?
unit db ?
num db ?
.code
main proc
mov ax, @data
mov ds, ax
print msg
mov ah, 1
int 21h
sub al, 48
mov num, al
mov cx, 10
loop1:
newline
Activate Windows
Go to Settings to activate Windows


edit: C:\My Files\Szabist\5th Semester\COAL-LAB\LAB-Manual\fwdcoallabtask\Task 42.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate
emulator screen (80x25 chars)
Enter a Number: 4
4*01=04
4*02=08
4*03=12
4*04=16
4*05=20
4*06=24
4*07=28
4*08=32
4*09=36
4*10=40
original source c...
numberDisplay ten
numberDisplay unit
display '='
mov al, num
mul count
aam
mov ten, ah
mov unit, al
numberDisplay ten
numberDisplay unit
inc count
loop loop1
mov ah, 4ch
int 21h
emulator: Task 42.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
registers
AX 4C 30
BX 00 00
CX 00 00
DX 00 30
CS F400
IP 0204
SS 0710
SP 00FA
BP 0000
SI 0000
DI 0000
DS 0720
ES 0700
F400:0204
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
screen source reset aux vars debug stack flags

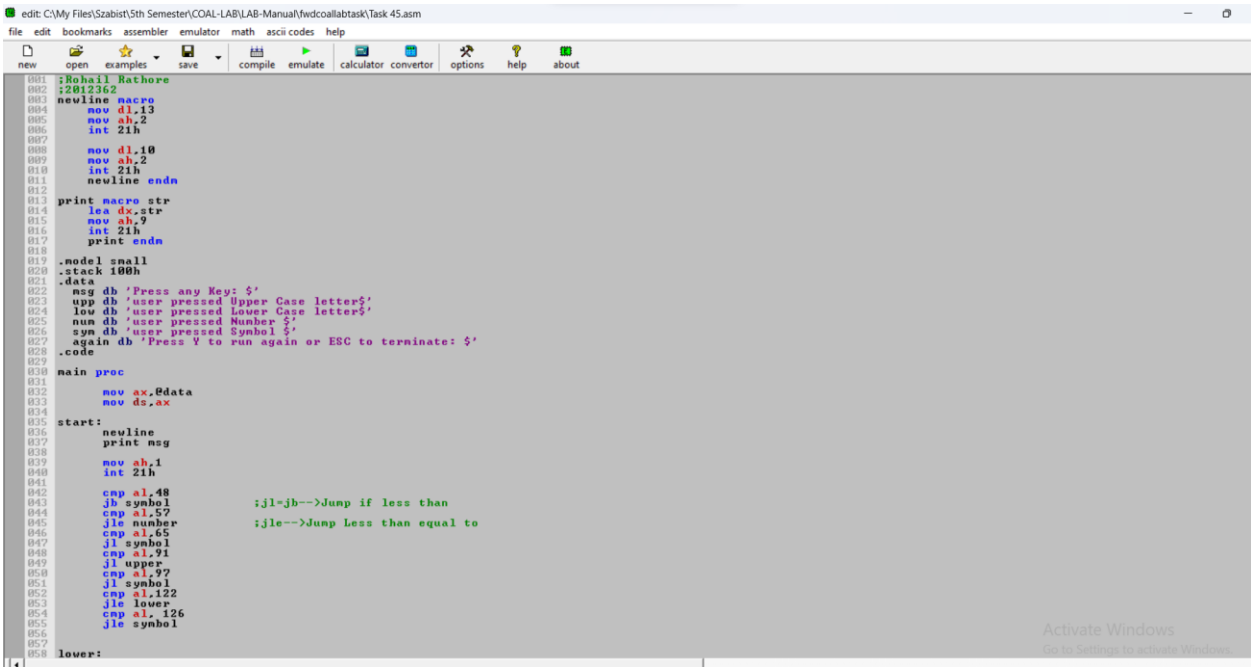
Task-43:



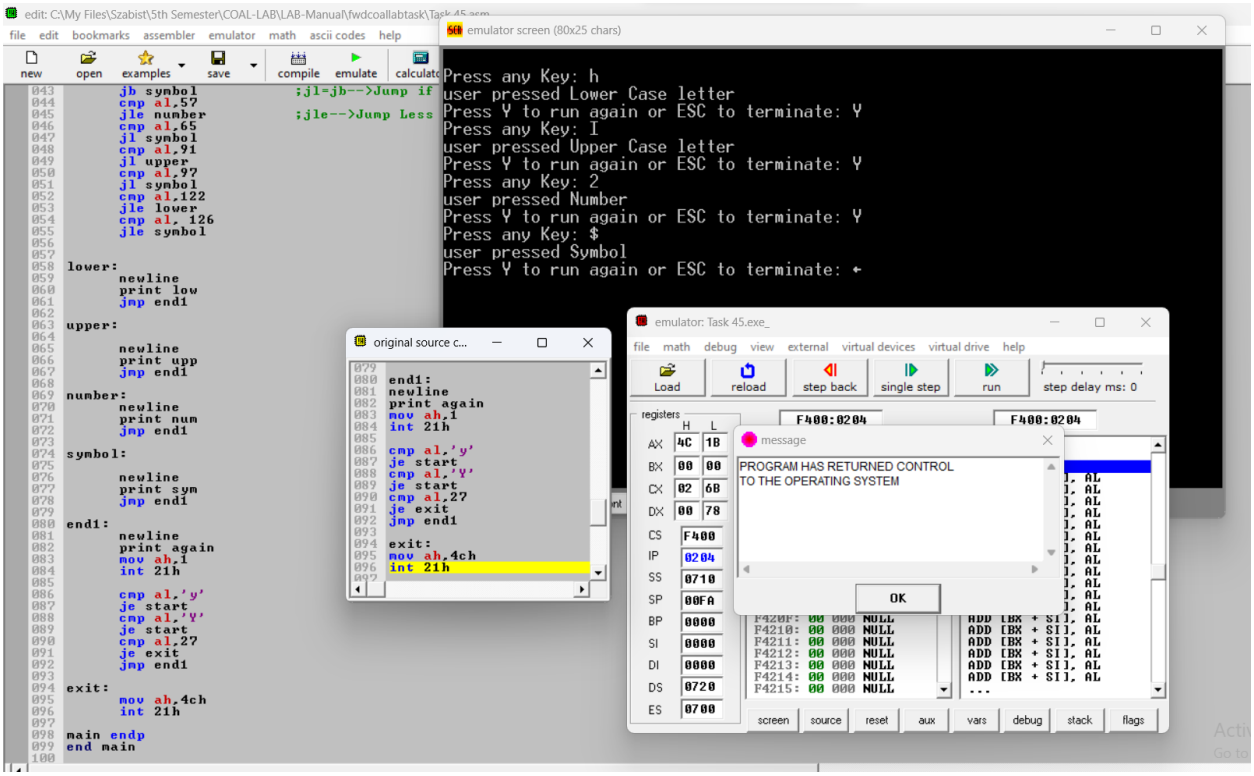
Task-44:



Task-45:



```
;Rohail Rathore
;2012362
newline macro
    mov dl,13
    mov ah,2
    int 21h

    mov dl,10
    mov ah,2
    int 21h
    newline endm

print macro str
    lea dx,str
    mov ah,9
    int 21h
    print endm

.model small
.stack 100h
.data
  msg db 'Press any Key: $'
  upp db 'user pressed Upper Case letter$'
  low db 'user pressed Lower Case letter$'
  num db 'user pressed Number $'
  sym db 'user pressed Symbol $'
  again db 'Press Y to run again or ESC to terminate: $'
.code

main proc

      mov ax,@data
      mov ds,ax

start:
      newline
      print msg

      mov ah,1
      int 21h

      cmp al,48
      jb symbol           ;jl=jb-->Jump if less than
      cmp al,57
      jle number          ;jle-->Jump Less than equal to
      cmp al,65
      jl symbol
      cmp al,91
      jl upper
      cmp al,97
      jl symbol
      cmp al,122
      jle lower
      cmp al, 126
      jle symbol


lower:
```



```
      jb symbol           ;jl=jb-->Jump if
      cmp al,57
      jle number          ;jle-->Jump Less
      cmp al,65
      jl symbol
      cmp al,91
      jl upper
      cmp al,97
      jl symbol
      cmp al,122
      jle lower
      cmp al, 126
      jle symbol


lower:
      newline
      print low
      jmp end1

upper:

      newline
      print upp
      jmp end1

number:
      newline
      print num
      jmp end1

symbol:

      newline
      print sym
      jmp end1

end1:
      newline
      print again
      mov ah,1
      int 21h

      cmp al,'y'
      je start
      cmp al,'Y'
      je start
      cmp al,27
      je exit
      jmp end1

exit:
      mov ah,4ch
      int 21h

main endp
end main
```

```
Press any Key: h
user pressed Lower Case letter
Press Y to run again or ESC to terminate: Y
Press any Key: I
user pressed Upper Case letter
Press Y to run again or ESC to terminate: Y
Press any Key: 2
user pressed Number
Press Y to run again or ESC to terminate: Y
Press any Key: $
user pressed Symbol
Press Y to run again or ESC to terminate: ←
```

```
end1:
newline
print again
mov ah,1
int 21h

cmp al,'y'
je start
cmp al,'Y'
je start
cmp al,27
je exit
jmp end1

exit:
mov ah,4ch
int 21h
```