# Institute of Mathematics and Computer Science
Assembly Language

**ASSIGNMENT**

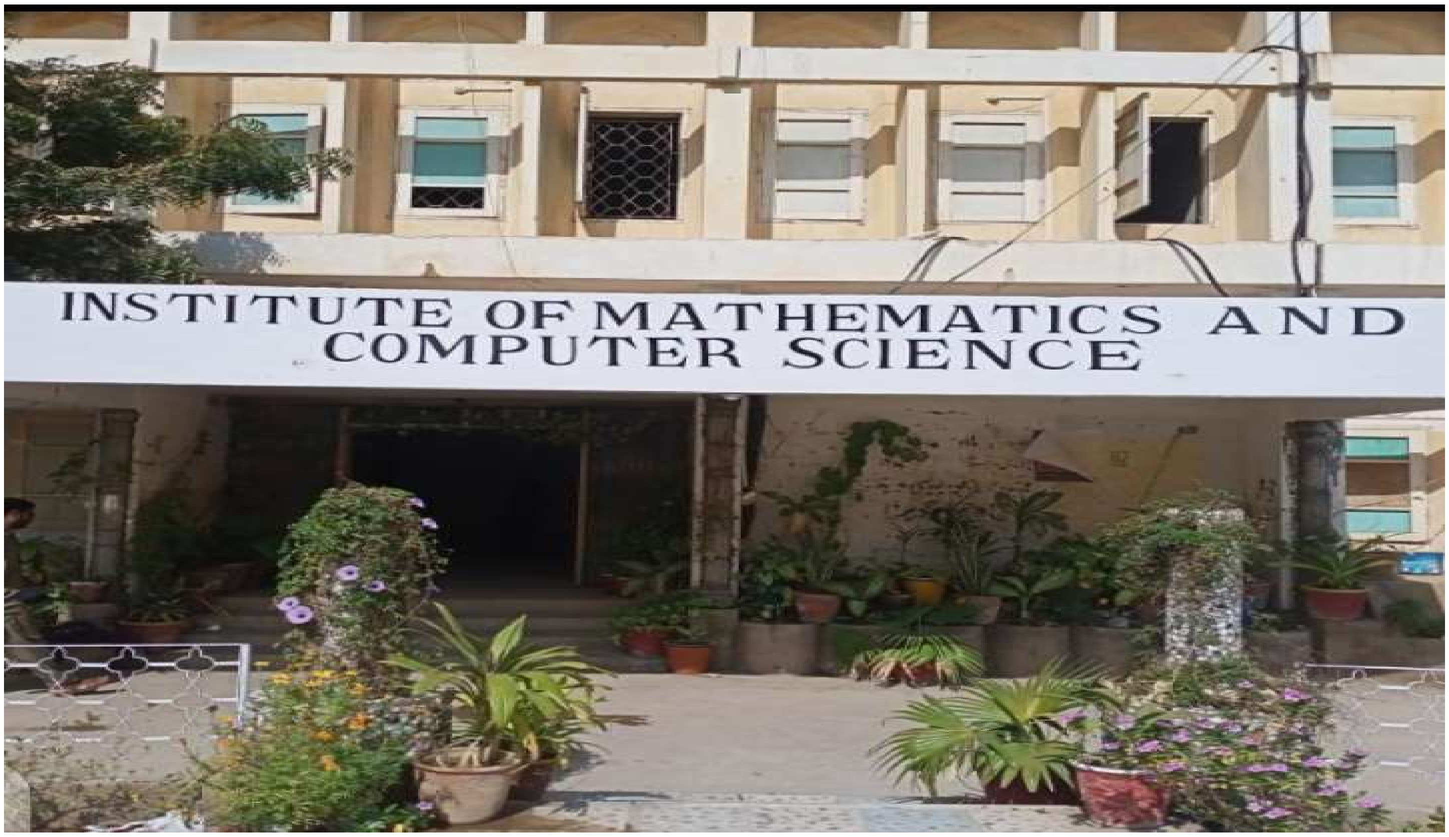
**STUDENT NAME:** **IRFAN ALI**

**ROLL NO:** **2K22/CSE/48**

**SUBJECT:** **COAL** (COMPUTER ORGANIZATION AND ASSEMBLY **LANGUAGE**)

**SUBMITTED TO:** **PRO: SIR IMTIAZ ALI KOREJO**

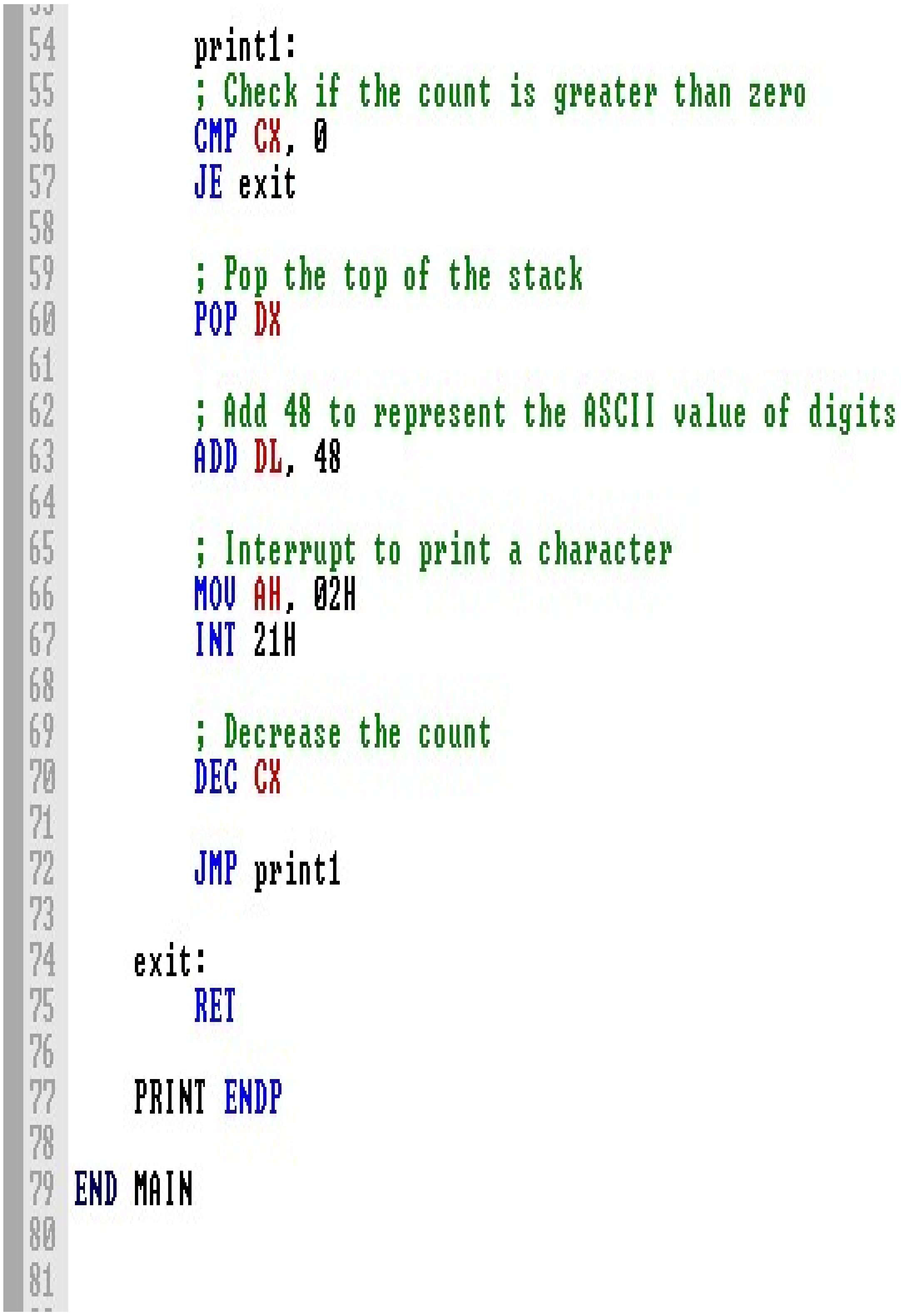
**DEPARTMENT:** **INSTITUE OF MATHEMATICS AND COMPUTER SCIENCE**



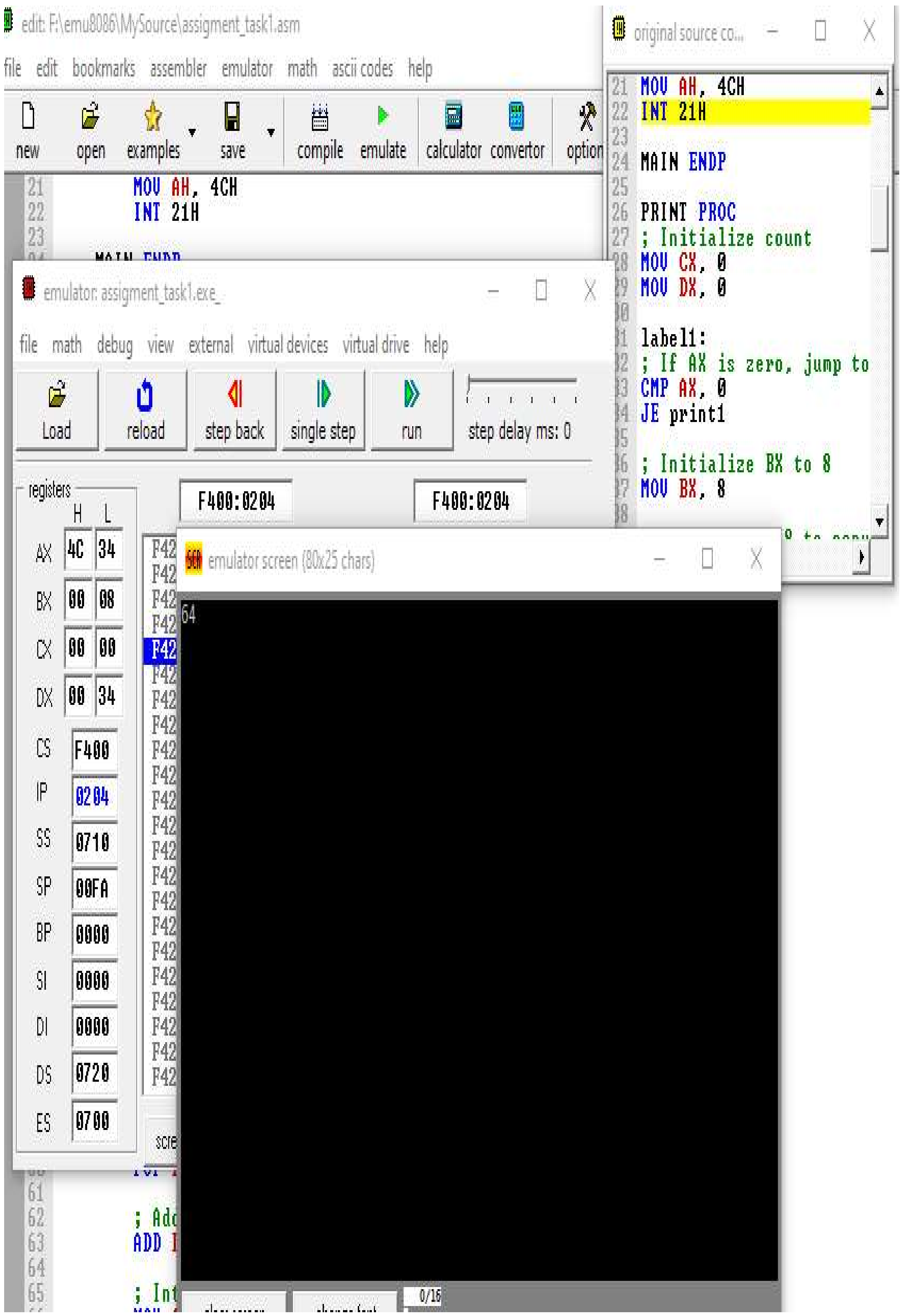
# WRITE A CODE ON 16-BIT ARCHITECTURE

QNO: 01**:- WRITE A PROGRAM FOR CONVERTING A DECIMAL TO OCTAL.**

```
.MODEL SMALL
.STACK 100H
.DATA
    d1 DW 52
.CODE
    MAIN PROC
        MOV AX, @DATA
        MOV DS, AX

        ; Load the value stored in variable d1
        MOV AX, d1

        ; Convert the value to octal and print the value
        CALL PRINT

        ; Interrupt to exit
        MOV AH, 4CH
        INT 21H

    MAIN ENDP

    PRINT PROC
        ; Initialize count
        MOV CX, 0
        MOV DX, 0

    label1:
        ; If AX is zero, jump to print1
        CMP AX, 0
        JE print1

        ; Initialize BX to 8
        MOV BX, 8

        ; Divide AX by 8 to convert it to octal
        DIV BX

        ; Push the result in the stack
        PUSH DX

        ; Increment the count
        INC CX

        ; Set DX to 0
        XOR DX, DX

        JMP label1
```



```
        print1:
        ; Check if the count is greater than zero
        CMP CX, 0
        JE exit

        ; Pop the top of the stack
        POP DX

        ; Add 48 to represent the ASCII value of digits
        ADD DL, 48

        ; Interrupt to print a character
        MOV AH, 02H
        INT 21H

        ; Decrease the count
        DEC CX

        JMP print1

    exit:
        RET

    PRINT ENDP

END MAIN
```


edit: F:\emu8086\MySource\assigment_task1.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor option
emulator: assigment_task1.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
original source co...
emulator screen (80x25 chars)

```
        MOV AH, 4CH
        INT 21H

    MAIN ENDP
```

```
MOV AH, 4CH
INT 21H

MAIN ENDP

PRINT PROC
; Initialize count
MOV CX, 0
MOV DX, 0

label1:
; If AX is zero, jump to
CMP AX, 0
JE print1

; Initialize BX to 8
MOV BX, 8
```

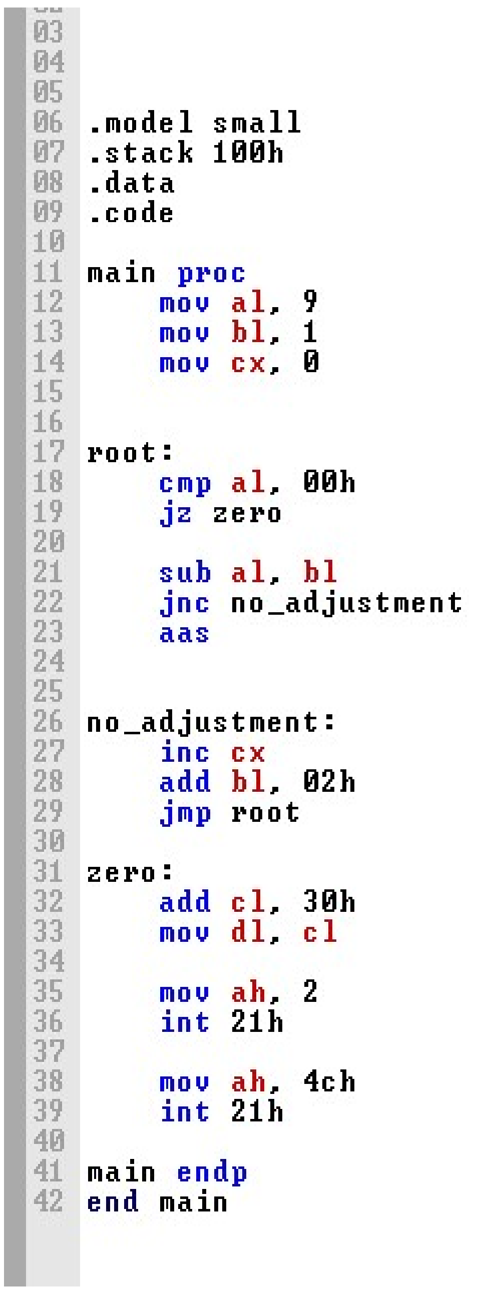
F400:0204 F400:0204

| registers | H | L |
|---|---|---|
| AX | 4C | 34 |
| BX | 00 | 08 |
| CX | 00 | 00 |
| DX | 00 | 34 |
| CS | F400 | |
| IP | 0204 | |
| SS | 0710 | |
| SP | 00FA | |
| BP | 0000 | |
| SI | 0000 | |
| DI | 0000 | |
| DS | 0720 | |
| ES | 0700 | |

```
64
```

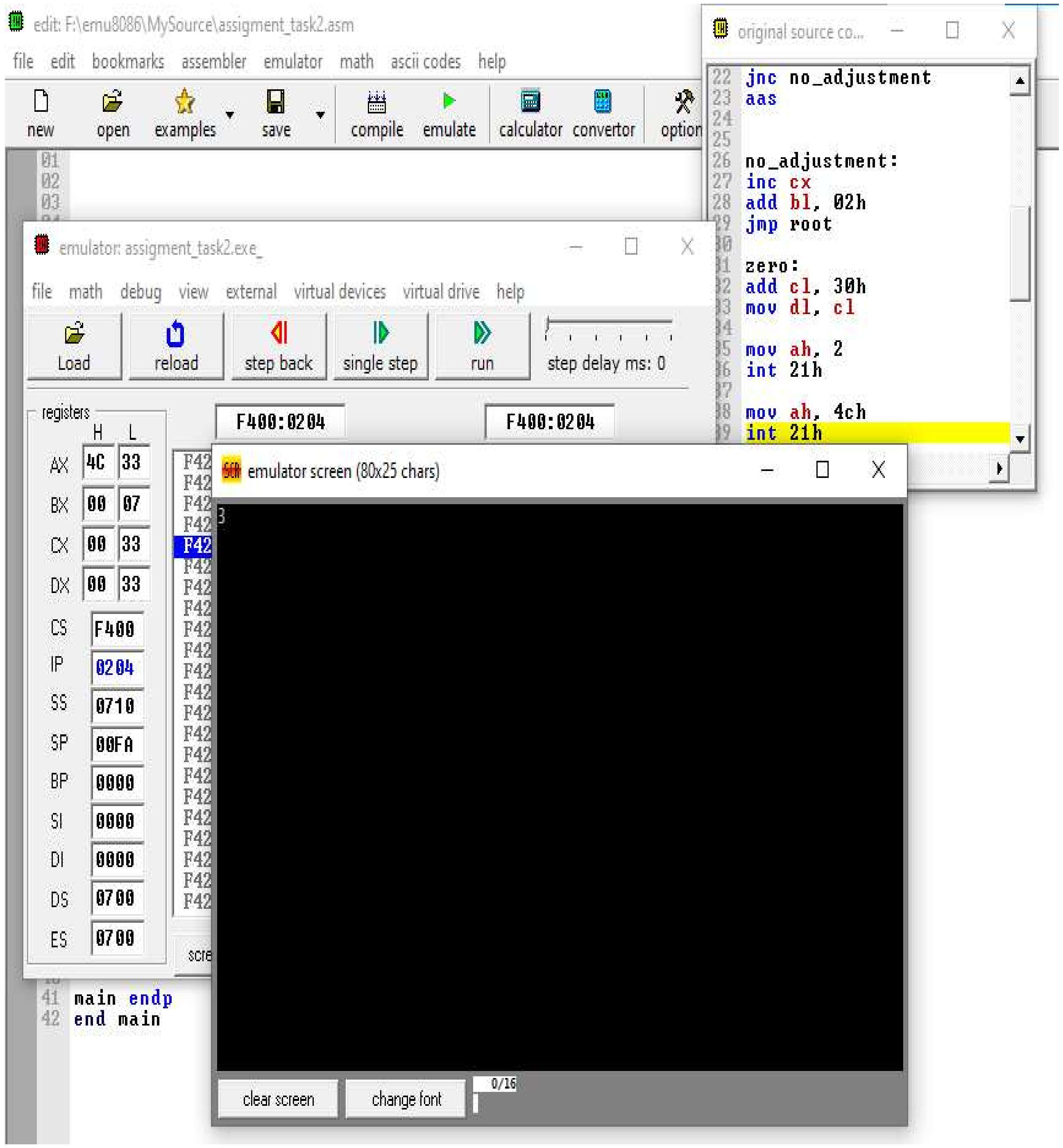
QNO: 02:-

**FIND A SQUARE ROOT OF A PERFECT SQUARE ROOT NUMBER**



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

main proc
     mov al, 9
     mov bl, 1
     mov cx, 0


root:
     cmp al, 00h
     jz zero

     sub al, bl
     jnc no_adjustment
     aas


no_adjustment:
     inc cx
     add bl, 02h
     jmp root

zero:
     add cl, 30h
     mov dl, cl

     mov ah, 2
     int 21h

     mov ah, 4ch
     int 21h

main endp
end main
```

**RESULT:**


edit: F:\emu8086\MySource\assigment_task2.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor option
emulator: assigment_task2.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
original source co...
emulator screen (80x25 chars)
clear screen change font

```
jnc no_adjustment
aas

no_adjustment:
inc cx
add bl, 02h
jmp root

zero:
add cl, 30h
mov dl, cl

mov ah, 2
int 21h

mov ah, 4ch
int 21h
```

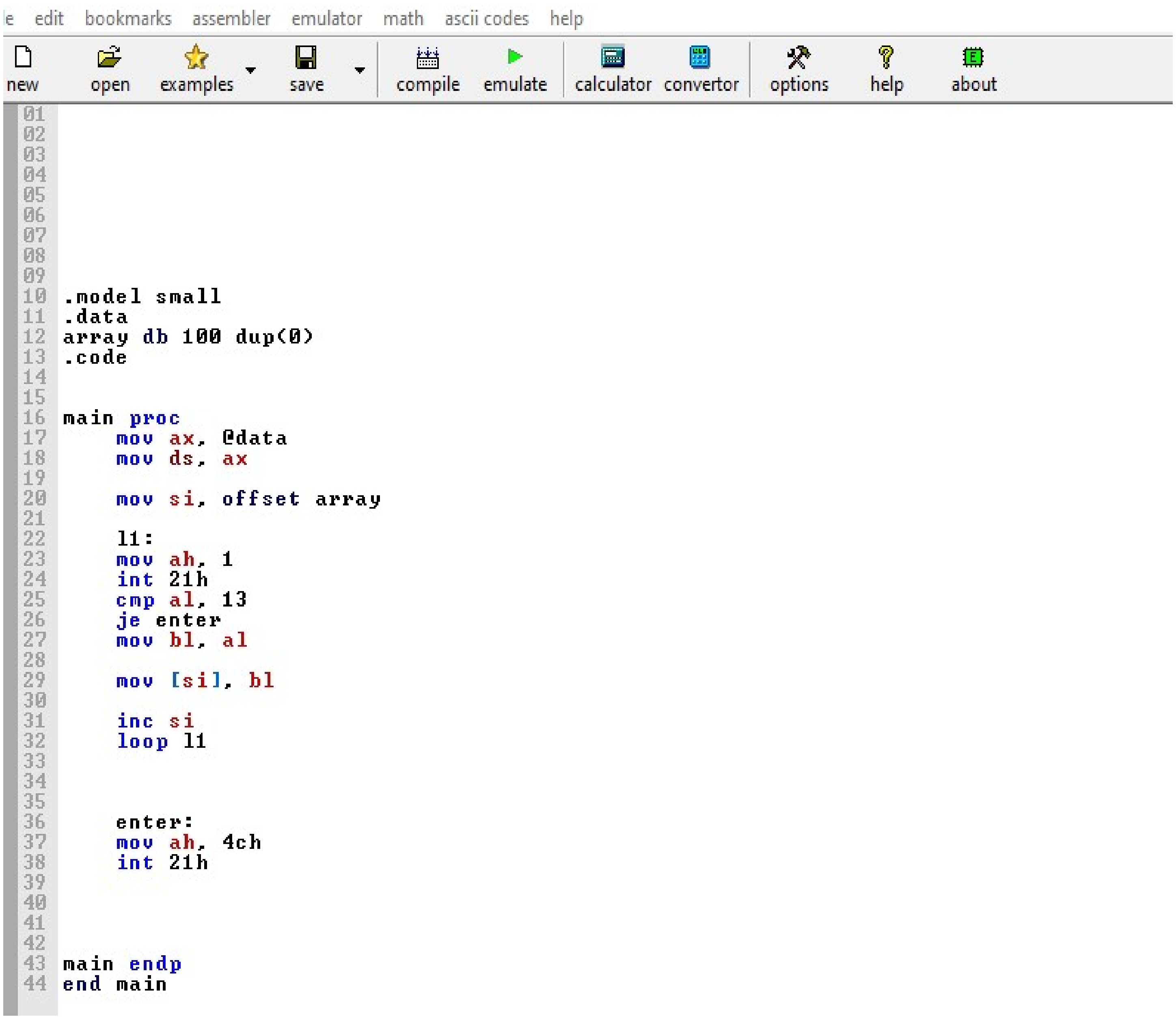
```
main endp
end main
```

```
registers
      H   L
AX   4C  33
BX   00  07
CX   00  33
DX   00  33
CS   F400
IP   0204
SS   0710
SP   00FA
BP   0000
SI   0000
DI   0000
DS   0700
ES   0700

F400:0204      F400:0204
```

```
3
```

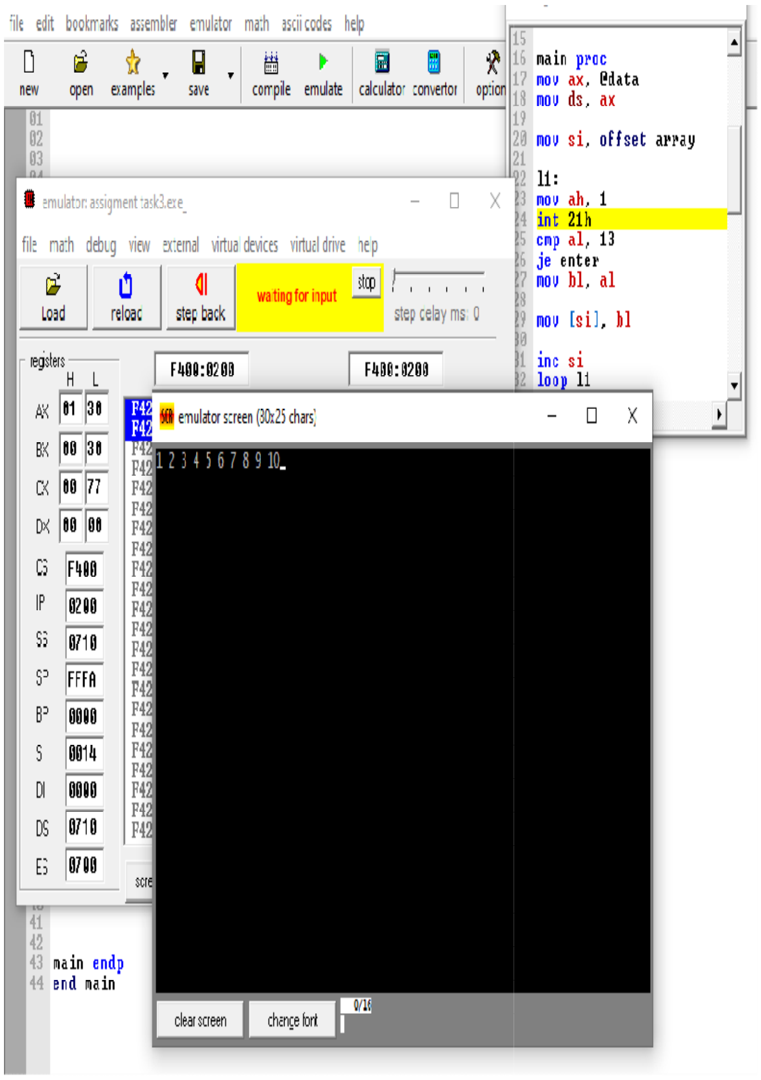
## QNO: 03:

**INSERT ELEMENTS INTO AN ARRAY USING DUP OPERATOR.**



e edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about

```
.model small
.data
array db 100 dup(0)
.code


main proc
    mov ax, @data
    mov ds, ax

    mov si, offset array

    l1:
    mov ah, 1
    int 21h
    cmp al, 13
    je enter
    mov bl, al

    mov [si], bl

    inc si
    loop l1



    enter:
    mov ah, 4ch
    int 21h




main endp
end main
```


file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options
emulator: assigment task3.exe_
file math debug view external virtual devices virtual drive help
Load reload step back waiting for input stop step delay ms: 0
emulator screen (80x25 chars)
clear screen change font

```
main proc
mov ax, @data
mov ds, ax

mov si, offset array

l1:
mov ah, 1
int 21h
cmp al, 13
je enter
mov bl, al

mov [si], bl

inc si
loop l1
```

```
main endp
end main
```

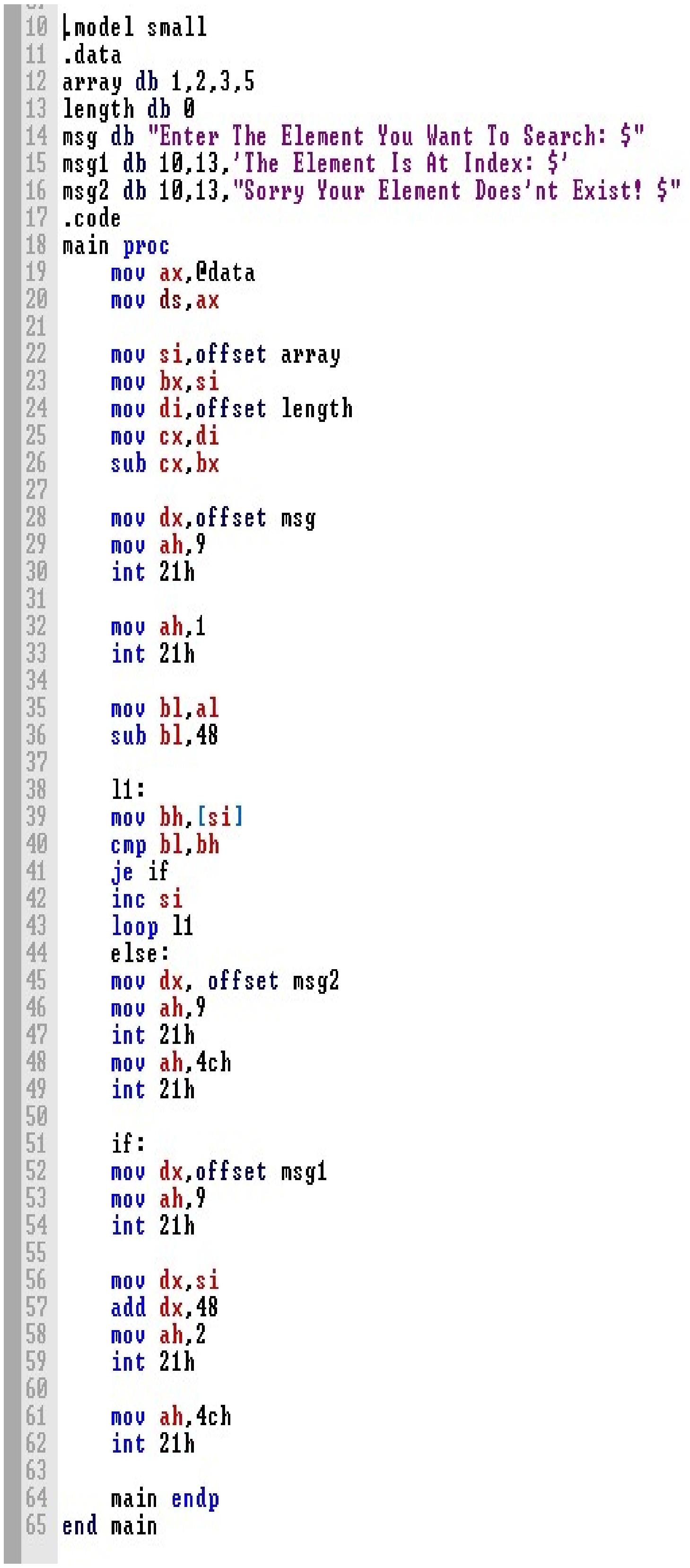
```
1 2 3 4 5 6 7 8 9 10_
```

```
registers
     H  L
AX  01 30
BX  00 30
CX  00 77
DX  00 00
CS  F400
IP  0200
SS  0710
SP  FFFA
BP  0000
SI  0014
DI  0000
DS  0710
ES  0700
```

F400:0200 F400:0200

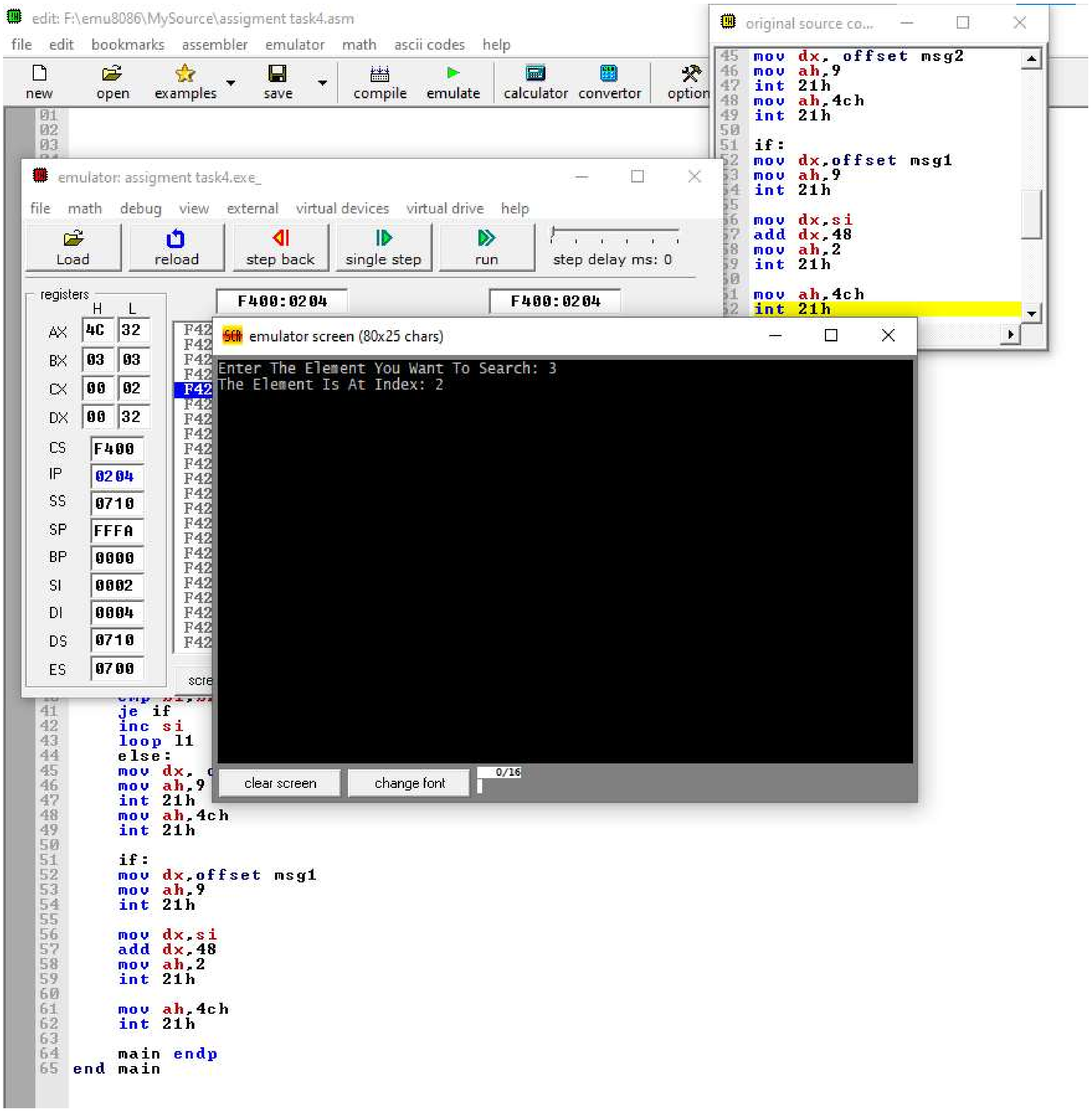
QNO: 04:

## SEARCH ELEMENT IN AN ARRAY.



```
.model small
.data
array db 1,2,3,5
length db 0
msg db "Enter The Element You Want To Search: $"
msg1 db 10,13,'The Element Is At Index: $'
msg2 db 10,13,"Sorry Your Element Does'nt Exist! $"
.code
main proc
    mov ax,@data
    mov ds,ax

    mov si,offset array
    mov bx,si
    mov di,offset length
    mov cx,di
    sub cx,bx

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

    mov ah,1
    int 21h

    mov bl,al
    sub bl,48

    l1:
    mov bh,[si]
    cmp bl,bh
    je if
    inc si
    loop l1
    else:
    mov dx, offset msg2
    mov ah,9
    int 21h
    mov ah,4ch
    int 21h

    if:
    mov dx,offset msg1
    mov ah,9
    int 21h

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

    mov ah,4ch
    int 21h

    main endp
end main
```

RESULT:


edit: F:\emu8086\MySource\assigment task4.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor option
emulator: assigment task4.exe_
file math debug view external virtual devices virtual drive help
Load reload step back single step run step delay ms: 0
F400:0204 F400:0204
emulator screen (80x25 chars)
clear screen change font 0/16
original source co...

```
registers
     H   L
AX  4C  32
BX  03  03
CX  00  02
DX  00  32
CS  F400
IP  0204
SS  0710
SP  FFFA
BP  0000
SI  0002
DI  0004
DS  0710
ES  0700
```

```
Enter The Element You Want To Search: 3
The Element Is At Index: 2
```

```
       je if
       inc si
       loop l1
       else:
       mov dx, o
       mov ah,9
       int 21h
       mov ah,4ch
       int 21h

       if:
       mov dx,offset msg1
       mov ah,9
       int 21h

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

       mov ah,4ch
       int 21h

       main endp
end main
```

```
mov dx, offset msg2
mov ah,9
int 21h
mov ah,4ch
int 21h

if:
mov dx,offset msg1
mov ah,9
int 21h

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

mov ah,4ch
int 21h
```

sir other programs will be send you soon with given output .