

ASSEMBLY PROGRAMMING LANGUAGE LAB

Password Checking

Contents

1	Objectives	2
2	Check And Match User Password	2
2.1	Code	2
2.2	Output	2

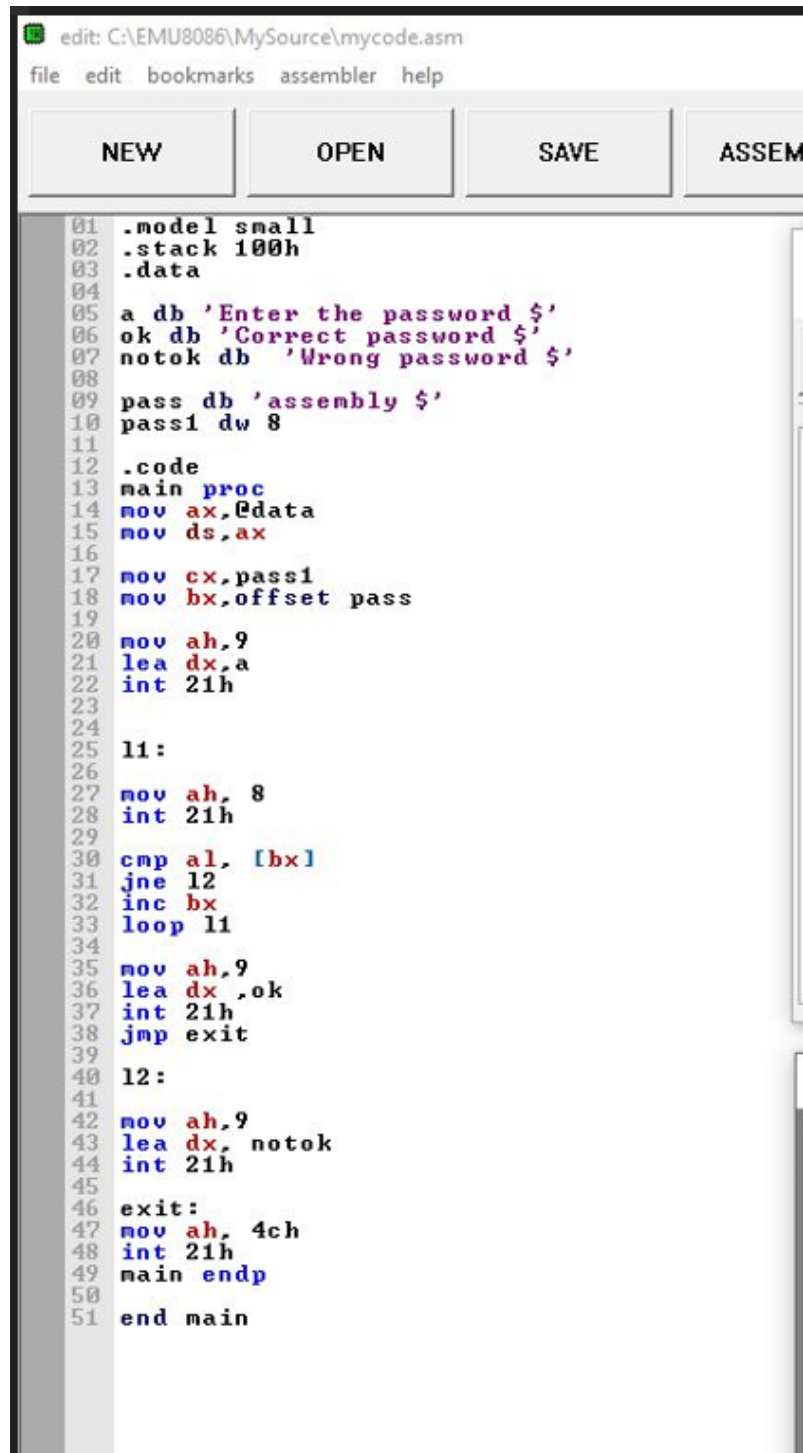
1 Objectives

In this lab I'm going to learn program how check password or matching in assembly programming language.

2 Check And Match User Password

2.1 Code

2.2 Output

The image shows a screenshot of an assembly code editor window. The title bar reads "edit: C:\EMU8086\MySource\mycode.asm". The menu bar includes "file", "edit", "bookmarks", "assembler", and "help". Below the menu bar are four buttons: "NEW", "OPEN", "SAVE", and "ASSEMBLE". The main text area contains assembly code for a password checker. The code starts with directives for model, stack, and data segments. It defines three strings: "a" (prompt), "ok" (correct password), and "notok" (wrong password). A password "assembly" is stored in memory. The main procedure begins by setting up registers and pointers. It enters a loop where it prompts the user to enter a password (8 characters). It then compares the entered password with the stored password. If they match, it displays "Correct password" and exits. If not, it displays "Wrong password" and loops back to prompt the user again. Finally, it exits the program with a return code of 4ch.

```
01 .model small
02 .stack 100h
03 .data
04
05 a db 'Enter the password $'
06 ok db 'Correct password $'
07 notok db 'Wrong password $'
08
09 pass db 'assembly $'
10 pass1 dw 8
11
12 .code
13 main proc
14 mov ax,@data
15 mov ds,ax
16
17 mov cx,pass1
18 mov bx,offset pass
19
20 mov ah,9
21 lea dx,a
22 int 21h
23
24
25 l1:
26
27 mov ah, 8
28 int 21h
29
30 cmp al, [bx]
31 jne l2
32 inc bx
33 loop l1
34
35 mov ah,9
36 lea dx,ok
37 int 21h
38 jmp exit
39
40 l2:
41
42 mov ah,9
43 lea dx,notok
44 int 21h
45
46 exit:
47 mov ah, 4ch
48 int 21h
49 main endp
50
51 end main
```

Figure 1: Check and match user password

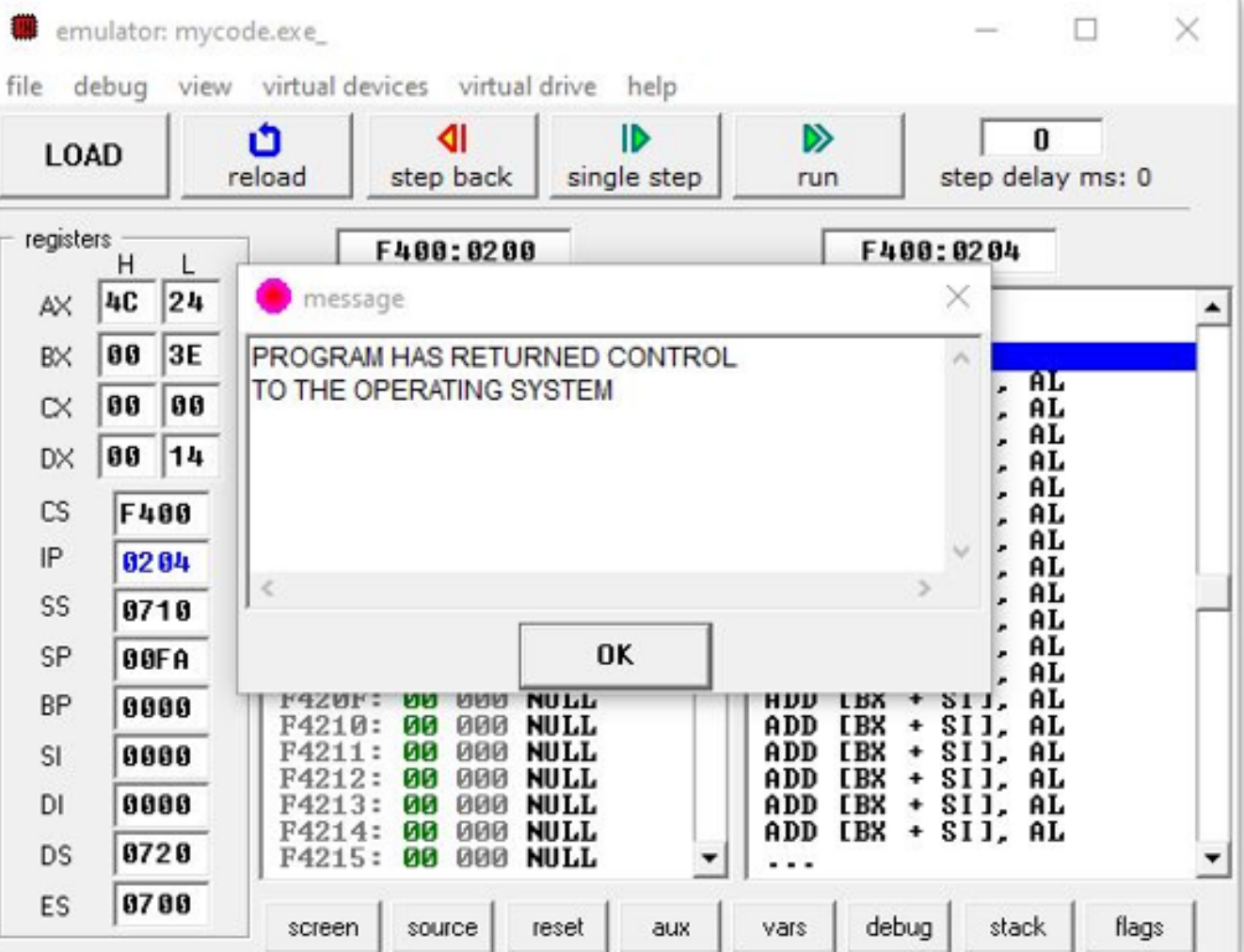
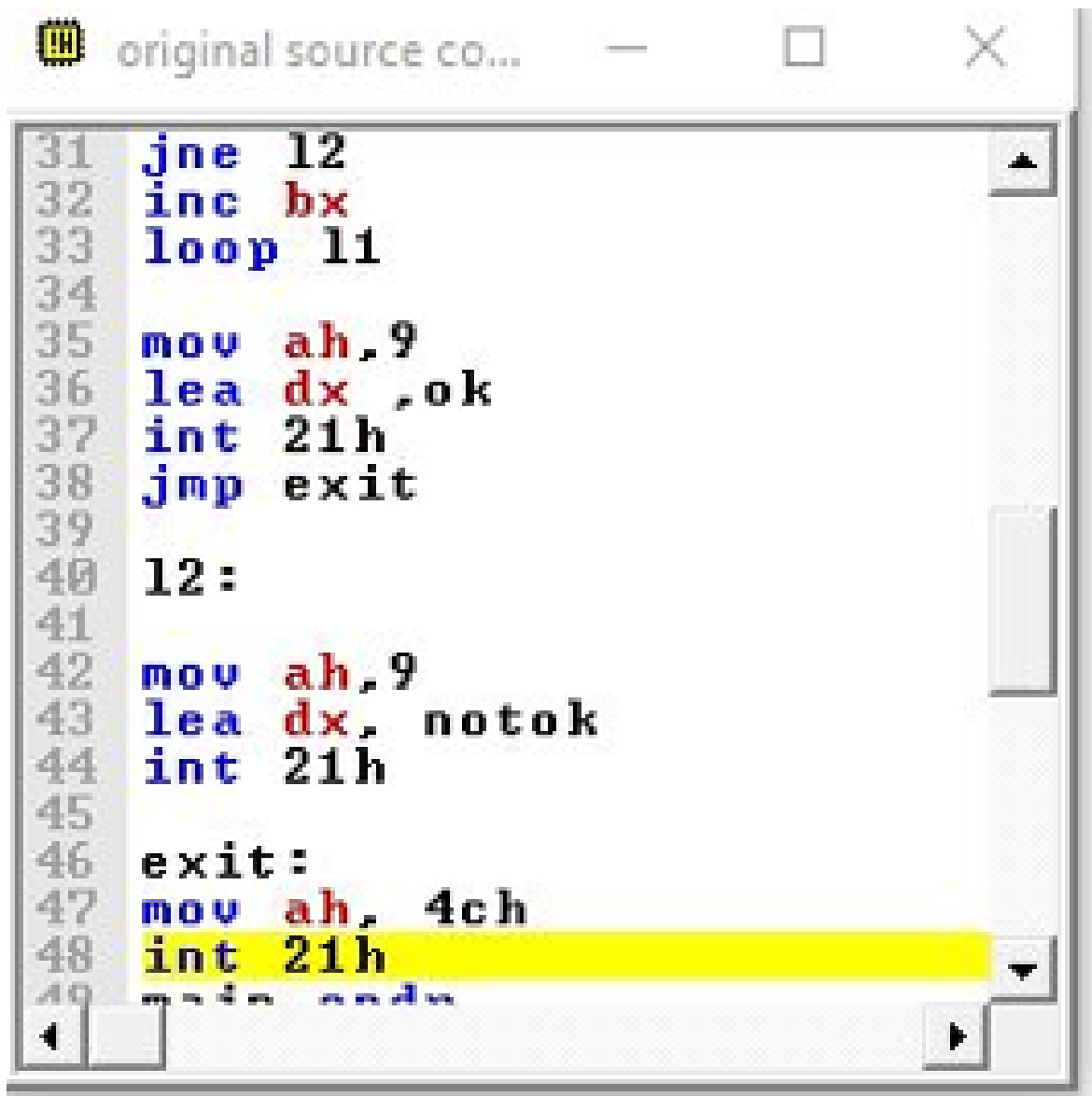


Figure 2: Counter loop



```
31 jne 12
32 inc bx
33 loop 11
34
35 mov ah,9
36 lea dx,ok
37 int 21h
38 jmp exit
39
40 12:
41
42 mov ah,9
43 lea dx, notok
44 int 21h
45
46 exit:
47 mov ah, 4ch
48 int 21h
49
```

Figure 3: Check and match user password

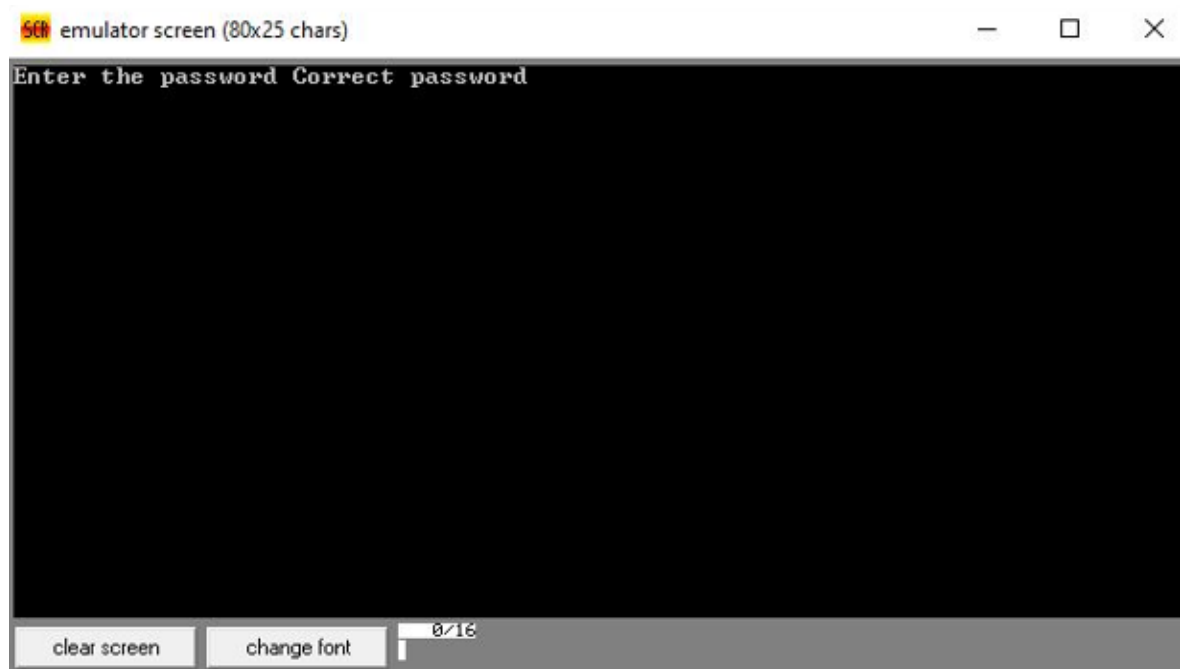


Figure 4: Output