



VIT[®]

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Microprocessors and Interfacing

(CSE – 3002)

LAB EXPERIMENT- 6

Name: **Vibhu Kumar Singh**

Reg. No: **19BCE0215**

Teacher: **Mr. Konguvel E.**

Q1) Write and execute 8086 ALP to verify the entered password. (Save a string as password and cross check with it to print "Entered Password is Correct or Entered Password is Incorrect")

Ans1)

ALP:

Q1) ALP:

data segment

password db 'chelsea1224'

len equ <\$-password>

msg1 db 10,13,'Please enter your password: \$'

msg2 db 10,13,'Welcome to my Program! \$'

msg3 db 10,13,'Incorrect Password! \$'

new db 10,13,'\$'

inst db 10 dup(0)

data ends

code segment

assume cs:code,ds:data

start:

mov ax,data

mov ds,ax

lea dx,msg1

mov ah,09h

int 21h

mov si,00

up1:

mov ah,08h

int 21h

cmp al,0dh

je down

mov [inst+si],al

mov dl,'\$'

mov ah,02h

int 21h

inc si

jmp up1

down:

mov bx, 00
mov cx, len

check:

mov al, [inst + bx]
mov dl, [password + bx]
cmp al, dl
jne fail
inc bx
loop check
lea dx, msg2
mov ah, 09h
int 21h
jmp finish

fail:

lea dx, msg3
mov ah, 09h
int 21h

finish:

mov ah, 04ch
int 21h

code ends

end start

end.

Screenshot of ALP:

edit: C:\Users\Vibhu\OneDrive - vit.ac.in\Desktop\Fall Semester 21-22\Micro\ELA\Task-6\password.asm

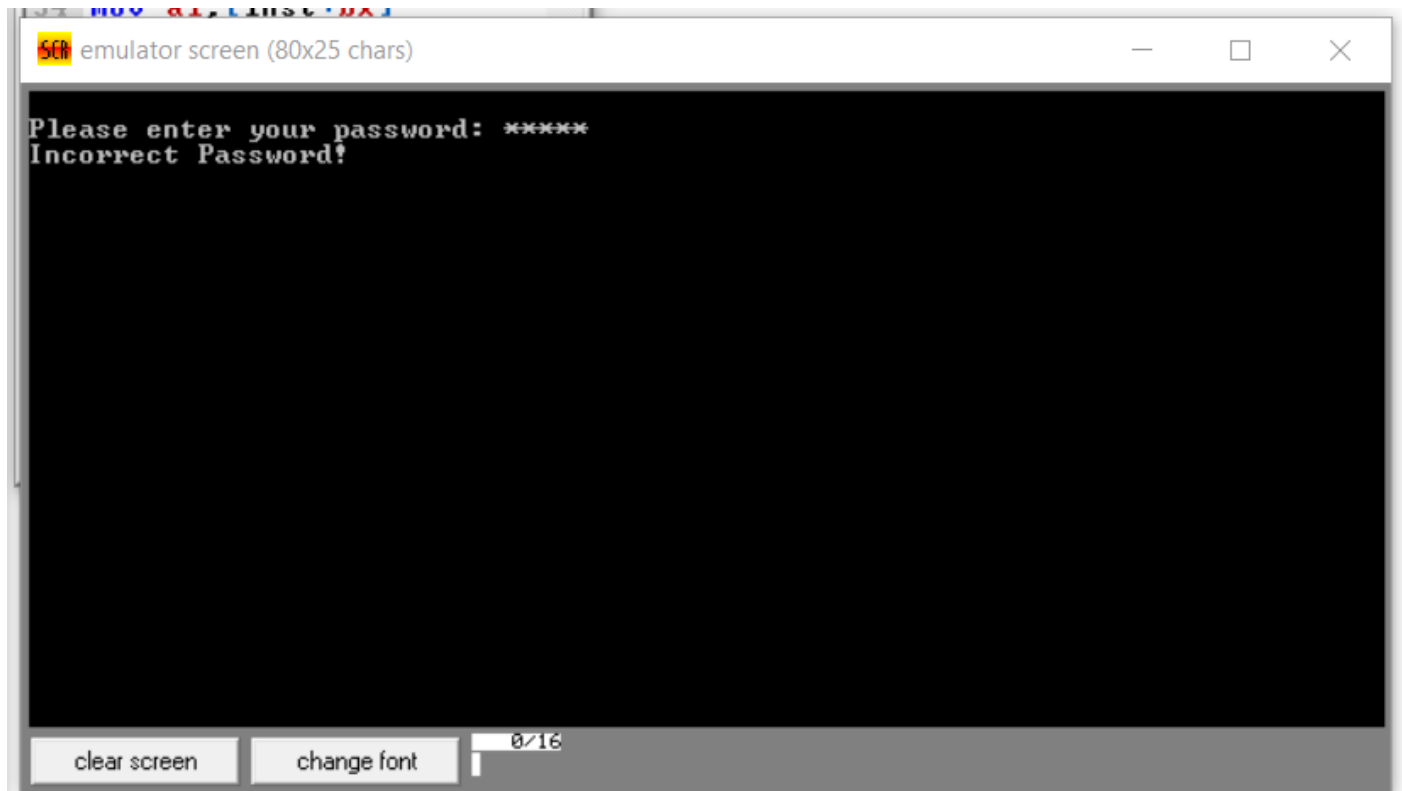
file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options help about

```
01 data segment
02     password db 'chelsea1224'
03     len equ (<$-password>)
04     msg1 db 10,13,'Please enter your password: $'
05     msg2 db 10,13,'Welcome to my Program!$'
06     msg3 db 10,13,'Incorrect Password!$'
07     new db 10,13,'$'
08     inst db 10 dup(0)
09 data ends
10 code segment
11     assume cs:code,ds:data
12     start:
13         mov ax,data
14         mov ds,ax
15         lea dx,msg1
16         mov ah,09h
17         int 21h
18         mov si,00
19     up1:
20         mov ah,08h
21         int 21h
22         cmp al,0dh
23         je down
24         mov [inst+si],al
25         mov dl,'*'
26         mov ah,02h
27         int 21h
28         inc si
29         jmp up1
30     down:
31         mov bx,00
32         mov cx,len
33     check:
34         mov al,[inst+bx]
35         mov dl,[password+bx]
36         cmp al,dl
37         jne fail
38         inc bx
39         loop check
40         lea dx,msg2
41         mov ah,09h
42         int 21h
43         jmp finish
44     fail:
45         lea dx,msg3
46         mov ah,009h
47         int 21h
48     finish:
49         mov ah,04ch
50         int 21h
51 code ends
52 end start
53 end
```

Output:

Incorrect Password



Correct Password

