1. Write a program in assembly language to take two single-digit numbers as input and

display whether they are equal or not.

**Ans:**

**Code:**

.model small

.stack 100h

.data

prompt1 db 'Enter first single-digit number: $'

prompt2 db 0dh, 0ah, 'Enter second single-digit number: $'

equal\_msg db 0dh, 0ah, 'The numbers are equal.$'

not\_equal\_msg db 0dh, 0ah, 'The numbers are not equal.$'

.code

main proc

mov ax, @data

mov ds, ax

lea dx, prompt1

mov ah, 09h

int 21h

mov ah, 01h

int 21h

sub al, '0'

mov bl, al

lea dx, prompt2

mov ah, 09h

int 21h

mov ah, 01h

int 21h

sub al, '0'

cmp bl, al

je equal

lea dx, not\_equal\_msg

mov ah, 09h

int 21h

jmp exit

equal:

lea dx, equal\_msg

mov ah, 09h

int 21h

exit:

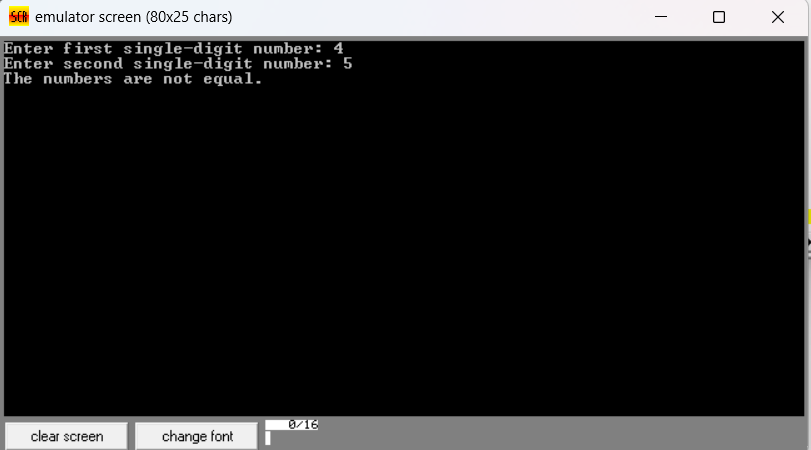
mov ah, 4ch

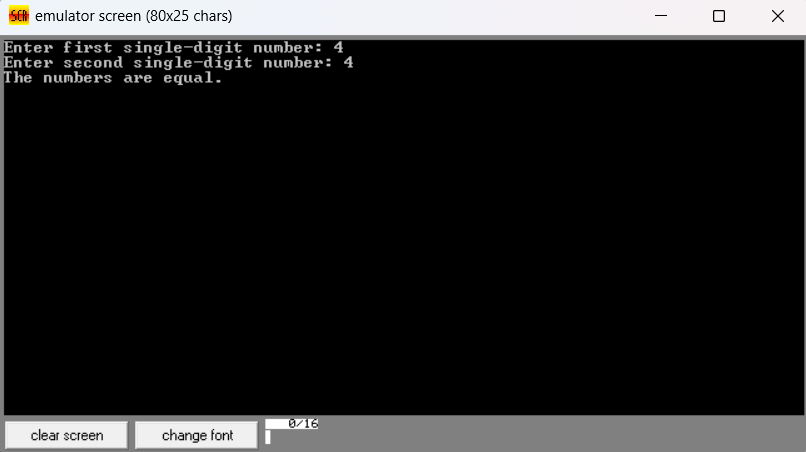
int 21h

main endp

end main

**Output:**

****

****

2. Write a program in assembly language to check whether a single-digit number is odd or

even.

**Ans:**

**Code:**

ORG 100h

start:

mov ah, 09h

lea dx, prompt\_msg

int 21h

mov ah, 01h

int 21h

sub al, 30h

mov bl, al

mov ah, 00h

mov al, bl

mov cl, 02h

div cl

cmp ah, 00h

je even

odd:

mov ah, 09h

lea dx, odd\_msg

int 21h

jmp done

even:

mov ah, 09h

lea dx, even\_msg

int 21h

done:

mov ah, 4Ch

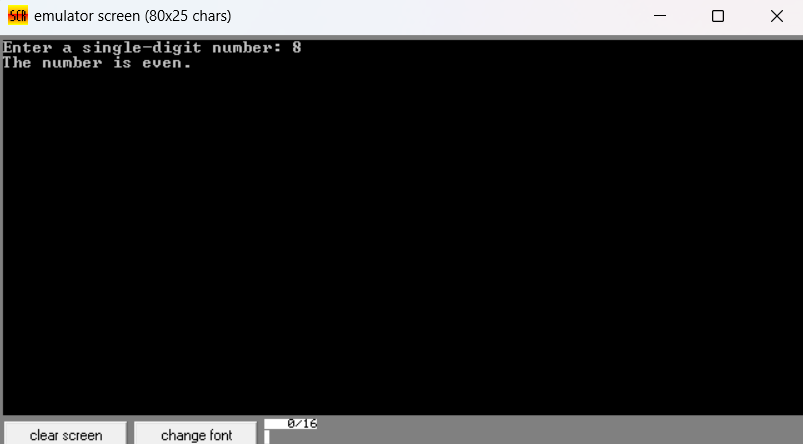
int 21h

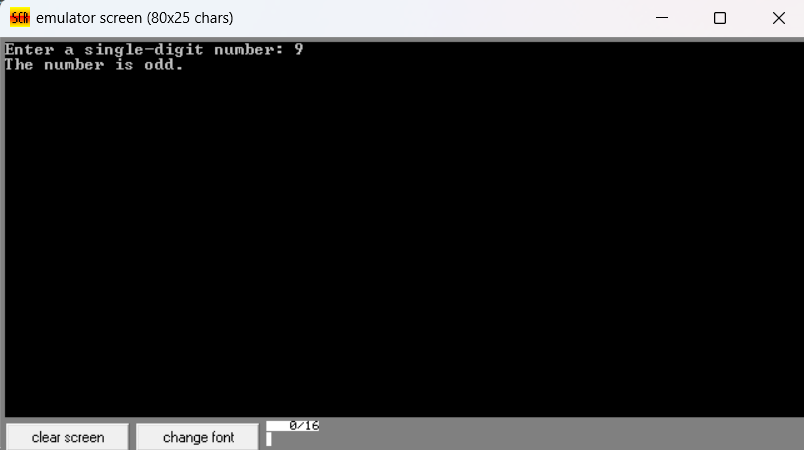
prompt\_msg db 'Enter a single-digit number: $'

even\_msg db 0Dh, 0Ah, 'The number is even.$'

odd\_msg db 0Dh, 0Ah, 'The number is odd.$'

**Output:**

****

****