LAB-9

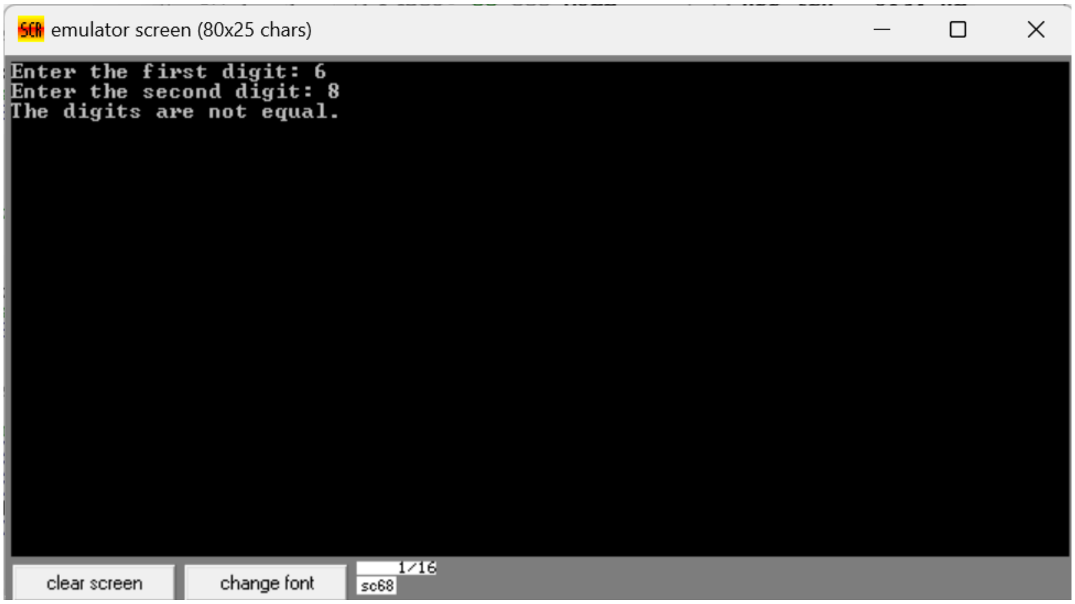
1.Write a program in assembly language to take two single-digit numbers as input and display whether they are equal or not.

CODE:

```
ORG 100h

; Display the message "Enter the first digit: "
MOV DX, OFFSET msg_input1
MOV AH, 09h
INT 21h

; Read the first digit from the user
MOV AH, 01h
INT 21h
MOV BL, AL          ; Store the first digit in BL

; Check if the first input is a digit
CMP BL, '0'         ; Compare with '0'
JL invalid_input    ; Jump if less than '0'
CMP BL, '9'         ; Compare with '9'
JG invalid_input    ; Jump if greater than '9'

; Display the message "Enter the second digit: "
MOV DX, OFFSET msg_input2
MOV AH, 09h
INT 21h

; Read the second digit from the user
MOV AH, 01h
INT 21h
MOV CL, AL          ; Store the second digit in CL

; Check if the second input is a digit
CMP CL, '0'         ; Compare with '0'
JL invalid_input    ; Jump if less than '0'
CMP CL, '9'         ; Compare with '9'
JG invalid_input    ; Jump if greater than '9'

; Compare the two digits
CMP BL, CL          ; Compare the two digits
JE digits_equal      ; Jump if equal
```

```
; Display message for not equal
MOV DX, OFFSET msg_not_equal

MOV AH, 09h
INT 21h
JMP end_program

digits_equal:
; Display message for equal
MOV DX, OFFSET msg_equal
MOV AH, 09h
INT 21h

end_program:
; Terminate the program
MOV AH, 4Ch
INT 21h

invalid_input:
; Display message for invalid input
MOV DX, OFFSET msg_invalid
MOV AH, 09h
INT 21h
JMP end_program

; Data section with messages
msg_input1  DB 'Enter the first digit: $'
msg_input2  DB 0Dh, 0Ah, 'Enter the second digit: $'
msg_equal   DB 0Dh, 0Ah, 'The digits are equal.$'
msg_not_equal DB 0Dh, 0Ah, 'The digits are not equal.$'
msg_invalid DB 0Dh, 0Ah, 'Invalid input! Please enter digits only.$'

END
```

OUTPUT:



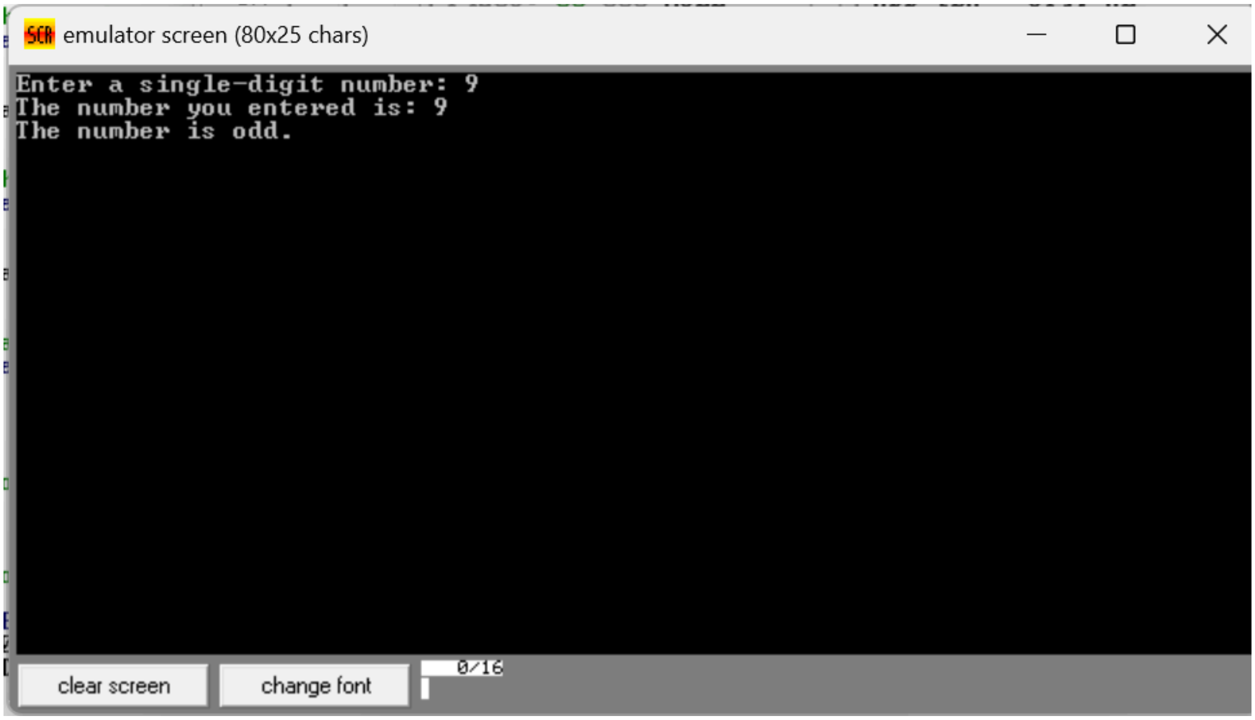
```
Enter the first digit: 6
Enter the second digit: 8
The digits are not equal.
```

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

```
CODE:
ORG 100h

; Prompt for the single-digit number
mov dx, offset msg_input
mov ah, 09h
int 21h

; Get the single digit
mov ah, 01h
int 21h
mov bl, al               ; Store the input in BL
cmp al, '0'              ; Check if it's a valid digit (ASCII '0' =
48)
jl NotDigit              ; If less than '0', it's not a digit
cmp al, '9'              ; Check if it's greater than '9' (ASCII '9' =
57)
jg NotDigit              ; If greater than '9', it's not a digit

; Display the input digit
mov dx, offset msg_output
mov ah, 09h
int 21h
mov dl, bl
mov ah, 02h
```

```
int 21h

; Convert the digit from ASCII to numeric value
sub bl, '0'

; Check if the number is odd or even using bitwise AND
mov al, bl               ; Move the number to AL for bitwise operation
and al, 1                ; AND with 1 to check the least significant
bit
jz Even                  ; If zero, the number is even
jmp Odd                  ; If not zero, the number is odd

Even:
; Display "The number is even"
mov dx, offset msg_even
mov ah, 09h
int 21h
jmp EndProgram

Odd:
; Display "The number is odd"
mov dx, offset msg_odd
mov ah, 09h
int 21h
jmp EndProgram

NotDigit:
; Handle invalid input
mov dx, offset msg_error
mov ah, 09h
int 21h

EndProgram:
; End the program
mov ah, 4Ch
int 21h

; Data section
msg_input DB "Enter a single-digit number: $"
msg_output DB 0Dh, 0Ah, "The number you entered is: $"
msg_even DB 0Dh, 0Ah, "The number is even.$"
msg_odd DB 0Dh, 0Ah, "The number is odd.$"
msg_error DB 0Dh, 0Ah, "Error: Not a digit!$"
```

OUTPUT:



```
Enter a single-digit number: 9
The number you entered is: 9
The number is odd.
```