**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

\_start:

; Display message: "Enter the first digit (0-9): "

MOV DX, OFFSET msg\_input1 ; Load the address of the first input prompt into DX

MOV AH, 09h ; DOS interrupt for displaying a string

INT 21h ; Call interrupt to print the message

; Read first digit from the user

MOV AH, 01h ; DOS interrupt to read a single character from input

INT 21h ; Store the character in AL (ASCII format)

SUB AL, '0' ; Convert ASCII to numerical value by subtracting '0'

MOV BL, AL ; Store the numerical value in BL (first digit)

; Move to a new line (CR LF)

MOV DL, 0Dh ; Carriage return (CR)

MOV AH, 02h ; DOS interrupt to output a character

INT 21h ; Print CR

MOV DL, 0Ah ; Line feed (LF)

INT 21h ; Print LF to move to the next line

; Display message: "Enter the second digit (0-9): "

MOV DX, OFFSET msg\_input2 ; Load the address of the second input prompt into DX

MOV AH, 09h ; DOS interrupt for displaying a string

INT 21h ; Call interrupt to print the message

; Read second digit from the user

MOV AH, 01h ; DOS interrupt to read a single character from input

INT 21h ; Store the character in AL (ASCII format)

SUB AL, '0' ; Convert ASCII to numerical value by subtracting '0'

MOV BH, AL ; Store the numerical value in BH (second digit)

; Move to a new line (CR LF)

MOV DL, 0Dh ; Carriage return (CR)

MOV AH, 02h ; DOS interrupt to output a character

INT 21h ; Print CR

MOV DL, 0Ah ; Line feed (LF)

INT 21h ; Print LF to move to the next line

; Compare the two digits

CMP BL, BH ; Compare the first digit (BL) with the second digit (BH)

JE equal ; If they are equal, jump to the 'equal' label

; If digits are not equal, display message: "The digits are not equal."

MOV DX, OFFSET msg\_not\_equal ; Load the address of the not equal message into DX

MOV AH, 09h ; DOS interrupt for displaying a string

INT 21h ; Call interrupt to print the message

JMP move\_to\_next\_line ; Jump to the move\_to\_next\_line label

equal:

; If digits are equal, display message: "The digits are equal."

MOV DX, OFFSET msg\_equal ; Load the address of the equal message into DX

MOV AH, 09h ; DOS interrupt for displaying a string

INT 21h ; Call interrupt to print the message

move\_to\_next\_line:

; Move to a new line (CR LF)

MOV DL, 0Dh ; Carriage return (CR)

MOV AH, 02h ; DOS interrupt to output a character

INT 21h ; Print CR

MOV DL, 0Ah ; Line feed (LF)

INT 21h ; Print LF to move to the next line

exit:

; Exit the program

MOV AH, 4Ch ; DOS interrupt for program termination

INT 21h ; Call interrupt to exit

msg\_input1 DB 'Enter the first digit (0-9): $'

msg\_input2 DB 'Enter the second digit (0-9): $'

msg\_equal DB 'The digits are equal.$'

msg\_not\_equal DB 'The digits are not equal.$'

END \_start



**Practice set:**

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

**even.**

**CODE:**

ORG 100h

\_start:

; Display message "Enter a single-digit number: "

MOV DX, OFFSET msg\_input

MOV AH, 09h

INT 21h

; Read a single digit from the user

MOV AH, 01h ; DOS function to read a character

INT 21h ; Store the input in AL

SUB AL, '0' ; Convert ASCII to integer

MOV BL, AL ; Store the input number in BL for later use

; Move to a new line before checking and printing odd/even result

CALL newline

; Check if the number is even or odd using bitwise AND

MOV AL, BL

AND AL, 1 ; Check the least significant bit

JZ even ; If the LSB is 0, the number is even

odd:

; Display "The number is odd."

MOV DX, OFFSET msg\_odd

MOV AH, 09h

INT 21h

JMP done

even:

; Display "The number is even."

MOV DX, OFFSET msg\_even

MOV AH, 09h

INT 21h

done:

; Move to a new line

CALL newline

; Terminate the program

MOV AH, 4Ch

INT 21h

newline:

; Move to a new line

MOV DL, 0Dh ; Carriage return

MOV AH, 02h

INT 21h

MOV DL, 0Ah ; Line feed

INT 21h

RET

msg\_input DB 'Enter a single-digit number: $'

msg\_odd DB 'The number is odd.$'

msg\_even DB 'The number is even.$'

END

