Lab 8 : Compiler Design

Anirudh Sathish : CS20B1125

Code:

// @ Author : Anirudh Sathish

// @ Roll\_No : CS20B1125

/\* Lab 8 : Write a C program to simulate lexical analyzer

 for validating operators\*/

// Defining header files

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int main()

{

    char input[6];

    printf("\n -------------------------------------- \n");

    printf("Validating expression \n ");

    printf("Enter the expression to be checked :  \n ");

    scanf("%s",input);

    // Checking for the options

    switch(input[0])

    {

        case '+':

            printf("This is the addition operator \n");

            break;

        case '-':

            printf("This is the subtraction operator \n");

            break;

        case '\*':

            printf("This is the multiplication operator \n");

            break;

        case '/':

            printf("This is the division operator \n");

            break;

        case '%':

            printf("This is the modulo operator \n");

            break;

        case '=':

            if(input[1] == '=')

            {

                printf("This is the equality check operator \n");

            }

            else

            {

                printf("This is the assignment operator \n ");

            }

            break;

        case '!':

            if(input[1] == '=')

            {

                printf("This is the Not Equal To operator \n ");

            }

            else

            {

                printf("This is the Bit NOT operator \n ");

            }

            break;

        case '&':

            if(input[1] == '&')

            {

                printf("This is the Logical AND To operator \n ");

            }

            else

            {

                printf("This is the Bitwise AND operator \n ");

            }

            break;

        case '|':

            if(input[1] == '|')

            {

                printf("This is the Logical OR operator \n ");

            }

            else

            {

                printf("This is the Bitwise OR operator \n ");

            }

            break;

        case '>':

            if(input[1] == '=')

            {

                printf("This is the Greater than equal to operator \n ");

            }

            else

            {

                printf("This is the Greater Than operator \n ");

            }

            break;

        case '<':

            if(input[1] == '=')

            {

                printf("This is the Lessr than equal to operator \n ");

            }

            else

            {

                printf("This is the Lesser Than operator \n ");

            }

            break;

        default:

            printf("The given option is not an expression \n");

            break;

    }

}

**Output:**

