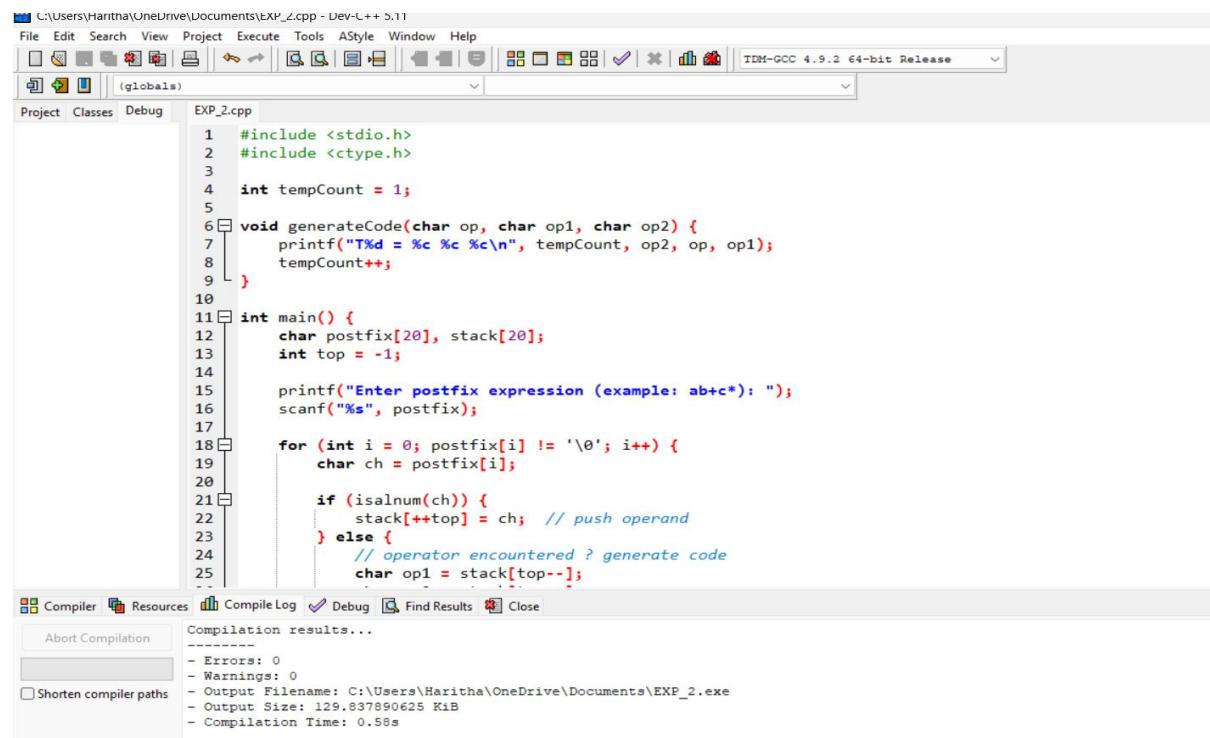


# EXPERIMENT-18

## AIM:

Write a C program to implement the back end of the compiler.

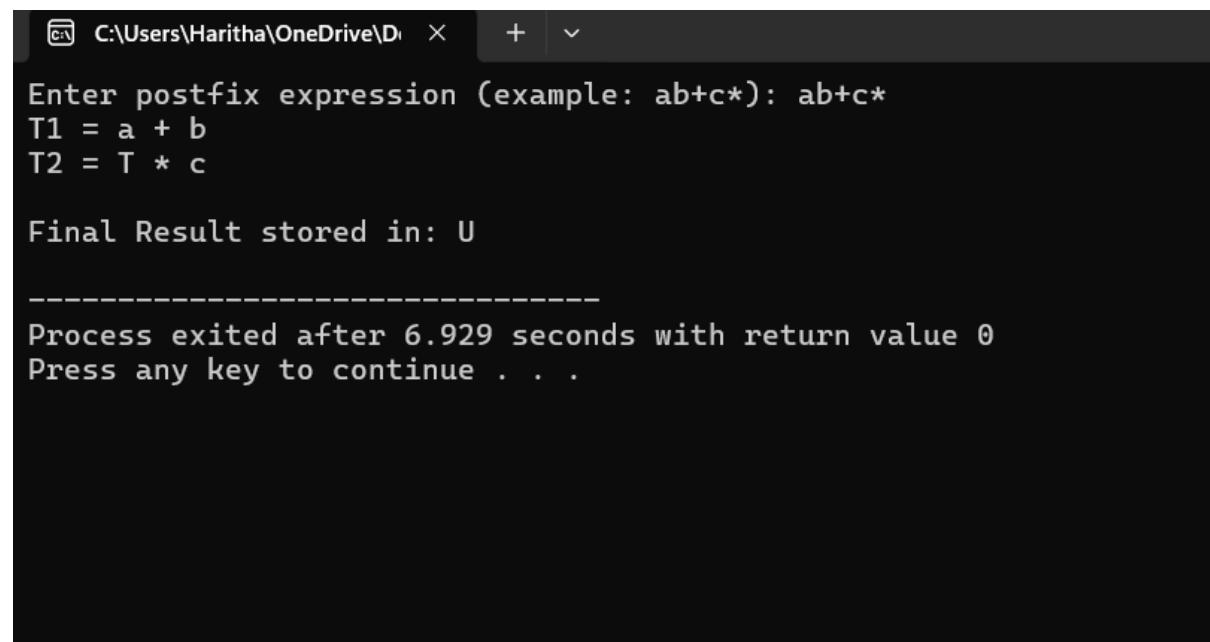
## PROGRAM:



The screenshot shows the Dev-C++ IDE interface. The project is named "EXP\_2.cpp". The code implements a simple postfix expression evaluator. It uses a stack to store operands and a temporary counter to generate unique identifiers for operators. The user is prompted to enter a postfix expression like "ab+c\*". The program then processes the expression and stores the result in variable 'U'.

```
1 #include <stdio.h>
2 #include <ctype.h>
3
4 int tempCount = 1;
5
6 void generateCode(char op, char op1, char op2) {
7     printf("T%d = %c %c %c\n", tempCount, op2, op, op1);
8     tempCount++;
9 }
10
11 int main() {
12     char postfix[20], stack[20];
13     int top = -1;
14
15     printf("Enter postfix expression (example: ab+c*): ");
16     scanf("%s", postfix);
17
18     for (int i = 0; postfix[i] != '\0'; i++) {
19         char ch = postfix[i];
20
21         if (isalnum(ch)) {
22             stack[++top] = ch; // push operand
23         } else {
24             // operator encountered ? generate code
25             char op1 = stack[top--];
```

## OUTPUT:



The terminal window shows the execution of the program. The user enters the postfix expression "ab+c\*". The program calculates intermediate results T1 and T2, and finally stores the result in variable 'U'. The process exits after 6.929 seconds.

```
C:\Users\Haritha\OneDrive\Documents> Enter postfix expression (example: ab+c*): ab+c*
T1 = a + b
T2 = T * c

Final Result stored in: U

-----
Process exited after 6.929 seconds with return value 0
Press any key to continue . . .
```