

# EXPERIMENT-9

## AIM:

Implement a C program to eliminate left recursion from a given CFG.

## PROGRAM:

The screenshot shows the Dev-C++ IDE interface. The main window displays the source code for EXP\_1.cpp. The code reads a non-terminal symbol and a production rule from the user, then processes the production rule to eliminate left recursion by creating two arrays, alpha and beta, and filling them with the tokens of the production rule. The code then checks if the alpha array is empty, which would indicate no left recursion was found. Below the code editor is the Compiler Log window, which shows the compilation results, including the output filename (EXP\_1.exe), size (129.2705078125 KiB), and time (0.47s). There are 0 errors and 0 warnings.

```
C:\Users\Haritha\OneDrive\Documents\EXP_1.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
Project Classes Debug EXP_1.cpp
1 #include <stdio.h>
2 #include <string.h>
3 int main() {
4     char nonTerminal, production[20], alpha[20], beta[20];
5     int i, index = 0, alphaIndex = 0, betaIndex = 0;
6     printf("Enter Non-Terminal (Example: A): ");
7     scanf(" %c", &nonTerminal);
8     printf("Enter Production (Example: A=Aa|b): ");
9     scanf("%s", production);
10    i = 2;
11    while (i < strlen(production)) {
12        if (production[i] == nonTerminal) {
13            i++;
14            while (i < strlen(production) && production[i] != '|')
15                alpha[alphaIndex++] = production[i++];
16            alpha[alphaIndex] = '\0';
17        }
18        else {
19            while (i < strlen(production) && production[i] != '|')
20                beta[betaIndex++] = production[i++];
21            beta[betaIndex] = '\0';
22        }
23        i++;
24    }
25    if (strlen(alpha) == 0) {
26        printf("No left recursion found.");
27    }
}
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Haritha\OneDrive\Documents\EXP_1.exe
- Output Size: 129.2705078125 KiB
- Compilation Time: 0.47s
```

## OUTPUT:

The screenshot shows a terminal window displaying the execution of the program. The user enters the non-terminal symbol 'A' and the production rule 'A=Aa|b'. The program then outputs the result of eliminating left recursion, showing the new rules 'A -> bA'' and 'A' -> aA' | e'. Finally, it prints a standard C-style exit message: 'Process exited after 83.91 seconds with return value 0' and 'Press any key to continue . . .'. The terminal window title bar shows the path 'C:\Users\Haritha\OneDrive\Documents\EXP\_1.exe'.

```
C:\Users\Haritha\OneDrive\Documents\EXP_1.exe + v
Enter Non-Terminal (Example: A): A
Enter Production (Example: A=Aa|b): A=Aa|b

After Eliminating Left Recursion:
A -> bA'
A' -> aA' | e

-----
Process exited after 83.91 seconds with return value 0
Press any key to continue . . .
```