PROGRAM 6:-

#include <stdio.h>

#include <string.h>

#define MAX 100

// Function to eliminate left recursion

void eliminateLeftRecursion(char nonTerminal, char alpha[], char beta[]) {

char newNonTerminal = nonTerminal + '\''; // Create new non-terminal A'

printf("Grammar after removing left recursion:\n");

printf("%c -> %s%c'\n", nonTerminal, beta, newNonTerminal); // A -> βA'

printf("%c' -> %s%c' | ε\n", newNonTerminal, alpha, newNonTerminal); // A' -> αA' | ε

}

int main() {

char nonTerminal, alpha[MAX], beta[MAX];

// Input format: A -> Aα | β

printf("Enter the left-recursive grammar (Format: A Aα β):\n");

scanf(" %c %s %s", &nonTerminal, alpha, beta);

if (nonTerminal == alpha[0]) { // Check if left recursion exists

eliminateLeftRecursion(nonTerminal, alpha + 1, beta);

} else {

printf("No left recursion found.\n");

}

return 0;

}

**OUTPUT:-**

