CODE:  
#include <stdio.h>

#include <string.h>

void eliminateLeftRecursion(char \*nonTerminal, char productions[][20], int n) {

char alpha[10][10], beta[10][10];

int alphaCount = 0, betaCount = 0;

for (int i = 0; i < n; i++) {

if (productions[i][0] == \*nonTerminal) {

strcpy(alpha[alphaCount++], productions[i] + 1);

} else {

strcpy(beta[betaCount++], productions[i]);

}

}

if (alphaCount == 0) {

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

return;

}

printf("%c -> ", \*nonTerminal);

for (int i = 0; i < betaCount; i++) {

printf("%s%c'", beta[i], \*nonTerminal);

if (i < betaCount - 1) printf(" | ");

}

printf("\n%c' -> ", \*nonTerminal);

for (int i = 0; i < alphaCount; i++) {

printf("%s%c'", alpha[i], \*nonTerminal);

if (i < alphaCount - 1) printf(" | ");

}

printf(" | ε\n");

}

int main() {

char nonTerminal = 'A';

char productions[][20] = {"Aa", "b"};

int n = 2;

eliminateLeftRecursion(&nonTerminal, productions, n);

return 0;

}

OUTPUT:

