**4a - DETECTION AND ELIMINATION OF LEFT RECURSIVE GRAMMAR :**

#include <iostream>

#include <string>

using namespace std;

int main()

{

int n, j, l, i, k;

int length[10] = {};

string d, a, b, flag;

char c;

cout<<"Enter Parent Non-Terminal: ";

cin >> c;

d.push\_back(c);

a += d + "\'->";

d += "->";

b += d;

cout<<"Enter productions: ";

cin >> n;

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

{

cout<<"Enter Production ";

cout<<i + 1<<" :";

cin >> flag;

length[i] = flag.size();

d += flag;

if (i != n - 1)

{

d += "|";

}

}

cout<<"The Production Rule is: ";

cout<<d<<endl;

for (i = 0, k = 3; i < n; i++)

{

if (d[0] != d[k])

{

cout<<"Production: "<< i + 1;

cout<<" does not have left recursion.";

cout<<endl;

if (d[k] == '#')

{

b.push\_back(d[0]);

b += "\'";

}

else

{

for (j = k; j < k + length[i]; j++)

{

b.push\_back(d[j]);

}

k = j + 1;

b.push\_back(d[0]);

b += "\'|";

}

}

else

{

cout<<"Production: "<< i + 1 ;

cout<< " has left recursion";

cout<< endl;

if (d[k] != '#')

{

for (l = k + 1; l < k + length[i]; l++)

{

a.push\_back(d[l]);

}

k = l + 1;

a.push\_back(d[0]);

a += "\'|";

}

}

}

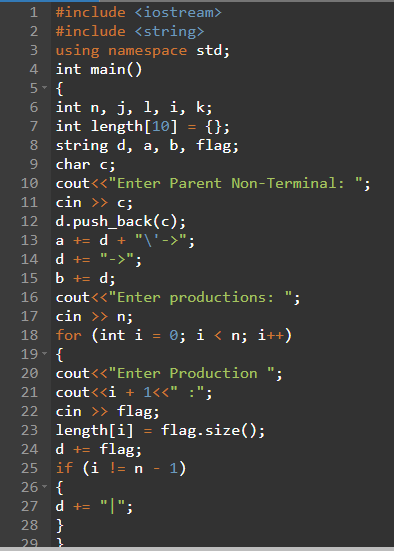
a += "#";

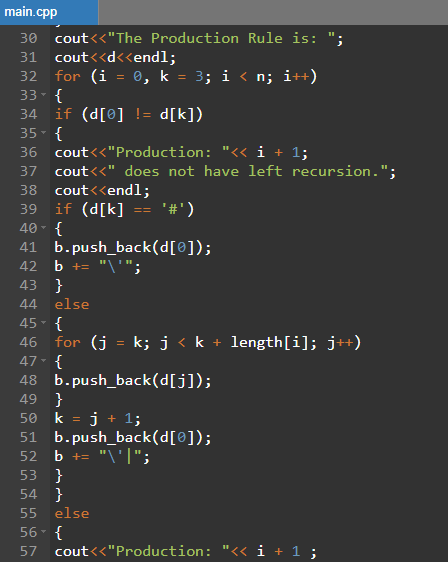
cout << b << endl;

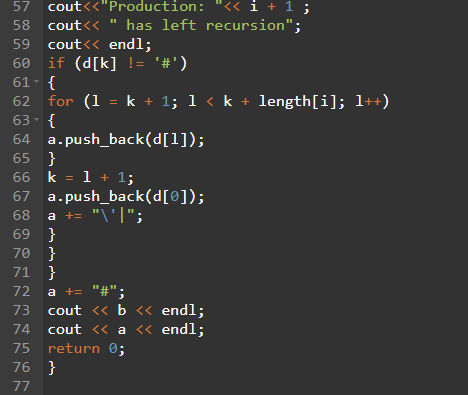
cout << a << endl;

return 0;

}







**Output:-**

