ASSIGNMENT -4

a) Program Name – Write a C program to check whether the given Grammar is left Recursive or not.

Objective - To determine whether the input is left recursive or not.

Resources- Online Compiler To simulate the program and test the output

Result

Program CODE-

```
#include<stdio.h>
#include<string.h>
#define SIZE 10
int main () {
        char non_terminal;
        char beta,alpha;
        int num;
        char production[10][SIZE];
        int index=3;
        printf("Enter Number of Production : ");
        scanf("%d",&num);
        printf("Enter the grammar as E->E-A :\n");
        for(int i=0;i<num;i++){
                scanf("%s",production[i]);
        }
        for(int i=0;i<num;i++){
```

```
                printf("\nGRAMMAR : : : %s",production[i]);
                non_terminal=production[i][0];
                if(non_terminal==production[i][index]) {
                        alpha=production[i][index+1];
                        printf(" is left recursive.\n");
                        while(production[i][index]!=0 && production[i][index]!='|')
                        index++;
                        if(production[i][index]!=0) {
                                beta=production[i][index+1];
                                printf("Grammar without left recursion:\n");
                                printf("%c->%c%c\'",non_terminal,beta,non_terminal);
                                printf("\n%c\'->%c%c\'|E\n",non_terminal,alpha,non_terminal);
                        }
                        else
                        printf(" can't be reduced\n");
                }
                else
                printf(" is not left recursive.\n");
                index=3;
        }
        }
```

Output

```
/tmp/oOEI8WxJT2.o
Enter Number of Production : 1
Enter the grammar as E->E-A :
T->T*F|E
GRAMMAR : : : T->T*F|E is left recursive.
Grammar without left recursion:
T->ET'
T'->*T'|E
```

b) Program Name - Write a C program to check whether the given Grammar is Left Factoring or not.

Objective- To determine whether the input is left factoring or not

Resources- Online Compiler To simulate the program and test the output Result

PROGRAM CODE-

```
#include<stdio.h>

#include<string.h>

int main()

{

   char gram[20],part1[20],part2[20],modifiedGram[20],newGram[20],tempGram[20];

   int i,j=0,k=0,l=0,pos;

   printf("Enter Production : A->");
```

```
gets(gram);
for(i=0;gram[i]!='|';i++,j++)
   part1[j]=gram[i];
part1[j]='\0';
for(j=++i,i=0;gram[j]!='\0';j++,i++)
   part2[i]=gram[j];
part2[i]='\0';
for(i=0;i<strlen(part1)||i<strlen(part2);i++){
   if(part1[i]==part2[i]){
      modifiedGram[k]=part1[i];
      k++;
      pos=i+1;
   }
}
for(i=pos,j=0;part1[i]!='\0';i++,j++){
   newGram[j]=part1[i];
}
newGram[j++]='|';
for(i=pos;part2[i]!='\0';i++,j++){
   newGram[j]=part2[i];
}
modifiedGram[k]='X';
modifiedGram[++k]='\0';
newGram[j]='\0';
```

```
    printf("\nGrammar Without Left Factoring : : \n");
    printf(" A->%s",modifiedGram);
    printf("\n X->%s\n",newGram);
}
```

OUTPUT-

```
Output

/tmp/oOEI8WxJT2.o
Enter Production : A->aA|a
Grammar Without Left Factoring : :
A->aX
 X->A|
```