

Compiler Design Lab

Assignment-3

Elimination of Immediate Left Recursion using C

Name : Vikraman S

Reg No. : 185001195

Code:

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>

void main()
{
    char input[20][50];
    char update[20][50];
    int i,j,count,ncount,flag=0;
    printf("Enter the number of productions : ");
    scanf(" %d",&count);

    for(i=0;i<count;i++)
    {
        printf("\n %d : ",i+1);
        scanf(" %s",input[i]);
    }

    printf("\n\nGrammar:\n");
    for(i=0;i<count;i++)
    {
        printf(" %d : %s\n",i+1,input[i]);
        if(input[i][0]==input[i][3])
            flag=1;
    }

    if(flag==1)
    {
        printf("\nLeft Recursive\n");
        printf("\nEliminating Left recursion\n");
    }

    else
        printf("\nNo Left Recursion\n");

    char nt,old_nt,alpha,beta;
    int index=3;
    for(i=0,j=0;i<count;i++)
    {
        nt=input[i][0];
        if(nt==input[i][index])
        {
            alpha=input[i][index+1];
            beta=input[i][index+2];
            sprintf(update[j],"%c->%c%c'",nt,beta,nt);
            j++;

            sprintf(update[j],"%c\''->%c%c%c'",nt,alpha,beta,nt);
            j++;
        }
    }
}
```

```

        sprintf(update[j], "%c\''->eps", nt);
        j++;
        old_nt=input[i][0];
    }
    else if(old_nt==nt)
        continue;
    else
    {
        strcpy(update[j], input[i]);
        j++;
    }

    ncount=j;
}

for(i=0; i<ncount; i++)
{
    printf(" %d : %s\n", i+1, update[i]);
}
}

```

Output:

```

viki@viki:~/Desktop/CD Lab/Ex3$ gcc left_rec3.c -o a
viki@viki:~/Desktop/CD Lab/Ex3$ ./a
Enter the number of productions : 5

1 : E->E+T
2 : E->T
3 : T->T*F
4 : T->F
5 : F->i

Grammar:
1 : E->E+T
2 : E->T
3 : T->T*F
4 : T->F
5 : F->i

Left Recursive

Eliminating Left recursion
1 : E->TE'
2 : E' ->+TE'
3 : E' ->eps
4 : T->FT'
5 : T' ->*FT'
6 : T' ->eps
7 : F->i
viki@viki:~/Desktop/CD Lab/Ex3$ █

```