

CSB 353: Compiler Design

LAB 5

Submitted By:

Name: PREM KUMAR

Roll No: 191210037

Branch: CSE

Semester: 6 th

Submitted To: Dr. Shelly Sachdeva

Department of Computer Science and Engineering



NATIONAL INSTITUTE OF TECHNOLOGY DELHI

2019-2023

Ques 1. Write a C program that accepts a grammar and removes left recursion from it.

Code:

```
1  √ #include <stdio.h>
2    #include <stdlib.h>
3    #include <string.h>
4  √ char *substr(const char *src, int m, int n)
5  {
6      int len = n - m;
7
8      char *dest = (char *)malloc(sizeof(char) * (len + 1));
9
10 √   for (int i = m; i < n && (*(src + i) != '\0'); i++)
11     {
12         *dest = *(src + i);
13         dest++;
14     }
15
16     *dest = '\0';
17
18     return dest - len;
19 }
20 √ int main()
21 {
22
23     int n;
24     printf("Prem Kumar\n191210037");
25     printf("\nEnter number of non terminals: ");
26     scanf("%d", &n);
27
28     printf("\nEnter non terminals one by one: ");
29
30     int i;
31
32     char nonter[10][10];
33
34     int leftrecr[10];
35     for (int i = 0; i < 10; i++)
36     {
37         leftrecr[i] = 0;
```

```
38
39     for (i = 0; i < n; ++i)
40     {
41
42         printf("\nNon terminal %d :", i + 1);
43
44         scanf("%s", nonter[i]);
45     }
46
47     char prod[10][10][10];
48     int prodsizes[10];
49     char dup[10];
50
51     printf("\nEnter '^' for null/epsilon");
52
53     for (i = 0; i < n; ++i)
54     {
55
56         printf("\nNumber of %s productions: ", nonter[i]);
57
58         int k;
59
60         scanf("%d", &k);
61         prodsizes[i] = k;
62         int j;
63
64         printf("\nOne by one enter all %s productions: ", nonter[i]);
65
66         for (j = 0; j < k; ++j)
67         {
68
69             printf("\nRHS of production %d : ", j + 1);
70
71             scanf("%s", prod[i][j]);
72
```

```

73         if (strlen(nonter[i]) <= strlen(prod[i][j]) && strcmp(nonter[i], substr(prod[i][j], 0, strlen(nonter[i]))) == 0)
74         {
75             leftrecr[i] = 1;
76         }
77     }
78
79     for (i = 0; i < n; ++i)
80     {
81
82         printf("%d", leftrecr[i]);
83     }
84     int tn = n, pn = n;
85     for (i = 0; i < n; ++i)
86     {
87
88         if (leftrecr[i] == 0)
89             continue;
90
91         int j;
92         strcpy(dup, nonter[i]);
93         strcat(dup, "");
94         strcpy(nonter[tn], dup);
95         tn++;
96
97         char temp[10][10];
98         int kt = 0;
99
100        for (j = 0; j < prodsizes[i]; ++j)
101        {
102
103            if (strlen(nonter[i]) <= strlen(prod[i][j]) && strcmp(nonter[i], substr(prod[i][j], 0, strlen(nonter[i]))) == 0)
104            {
105                char dup2[100];
106                strcpy(dup2, substr(prod[i][j], strlen(nonter[i]), strlen(prod[i][j]) - strlen(nonter[i])));
107                strcpy(temp[kt], strcat(dup2, dup));
108                kt++;
109                strcpy(prod[i], memmove(&prod[i][j], &prod[i][j + 1], strlen(prod[i]) - j));

```

```

110
111         --j;
112     }
113
114     else
115     {
116         char dup2[100];
117         strcpy(dup2, prod[i][j]);
118         strcat(dup2, dup);
119         strcpy(prod[i][j], dup2);
120     }
121 }
122 strcpy(temp[kt], "^");
123 strcpy(prod[pn], temp);
124 prodsizes[pn] = kt;
125 pn++;
126 }
127 n = tn;
128
129 printf("\n\n");
130
131 printf("\nNew set of non-terminals: ");
132
133 for (i = 0; i < tn; ++i)
134
135     printf("%s ", nonter[i]);
136
137 printf("\n\nNew set of productions: ");
138
139 for (i = 0; i < tn; ++i)
140 {
141
142     int j;
143
144     for (j = 0; j < prodsizes[i]; ++j)
145     {
146
147         printf("\n%s -> %s", nonter[i], prod[i][j]);
148     }
149 }
150
151 return 0;
152 }

```

Output:

```
Prem Kumar
191210037
Enter number of non terminals: 3

Enter non terminals one by one:
Non terminal 1 : E

Non terminal 2 : T

Non terminal 3 : F

Enter '^' for null
Number of E productions: 2

One by one enter all E productions
RHS of production 1: E+T

Number of F productions: 2

One by one enter all F productions
RHS of production 1: (E)

RHS of production 2: id
110

New set of non-terminals: E T F E' T'

New set of productions:
E -> TE'
T -> FT'
F -> (E)
F -> id
E' -> +TE'
E' -> ^
T' -> *FT'
T' -> ^
PS C:\Users\Prem\Desktop\6thSem\CSB353\lab5> □
```