

AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY  
DHAKA-1208, BANGLADESH.



Department of Computer Science and Engineering  
Spring 2019

Program: Bachelor of Science in Computer Science and Engineering

Course No: CSE 4130

Course Title: Formal Languages and Compilers Lab

Session: Spring 2019

Assignment No: 06

Date of Submission: 18/10/2019

Submitted to      Md. Aminur Rahman

Lecturer, Department of CSE, AUST.

Shoeb Mohammad Shahriar

Lecturer, Department of CSE, AUST.

Submitted By

Name            : Robiul Hasan Nowshad

Student ID   : 16.01.04.061

Lab Group   : B1

## Source Code:

```
1. #include<stdio.h>
2. #include<conio.h>
3. #include<string.h>
4. int main()
5. {
6.     char fin[10][20],st[10][20],ft[20][20],fol[20][20];
7.     int a=0,e,i,t,b,c,n,k,l=0,j,s,m,p;
8.     clrscr();
9.     printf("enter the no. of coordinates\n");
10.    scanf("%d",&n);
11.    printf("enter the productions in a grammar\n");
12.    for(i=0;i<n;i++)
13.        scanf("%s",st[i]);
14.    for(i=0;i<n;i++)
15.        fol[i][0]='\0';
16.    for(s=0;s<n;s++)
17.    {
18.        for(i=0;i<n;i++)
19.        {
20.            j=3;
21.            l=0;
22.            a=0;
23.            l1:if(!((st[i][j]>64)&&(st[i][j]<91)))
24.            {
25.                for(m=0;m<l;m++)
26.                {
27.                    if(ft[i][m]==st[i][j])
28.                        goto s1;
29.                }
30.                ft[i][l]=st[i][j];
31.                l=l+1;
32.                s1:j=j+1;
33.            }
34.            else
35.            {
36.                if(s>0)
37.                {
38.                    while(st[i][j]!=st[a][0])
39.                    {
40.                        a++;
41.                    }
42.                    b=0;
43.                    while(ft[a][b]!='\0')
44.                    {
45.                        for(m=0;m<l;m++)
46.                        {
47.                            if(ft[i][m]==ft[a][b])
48.                                goto s2;
49.                        }
50.                        ft[i][l]=ft[a][b];
51.                        l=l+1;
52.                        s2:b=b+1;
53.                    }
54.                }
55.            }
56.            while(st[i][j]!='\0')
57.            {
58.                if(st[i][j]=='|')
```

```

59.         {
60.             j=j+1;
61.             goto l1;
62.         }
63.         j=j+1;
64.     }
65.     ft[i][1]='\0';
66. }
67. }
68. printf("\n\n");
69. for(i=0;i<n;i++){
70.     printf("FIRST[%c]={ %c",st[i][0],ft[i][0]);
71.     int g=1;
72.     while(ft[i][g]!='\0'){
73.         printf(" ,%c", ft[i][g]);
74.         g++;
75.     }
76.     printf(" }\n");
77. }
78. fol[0][0]='$';
79. for(i=0;i<n;i++)
80. {
81.     k=0;
82.     j=3;
83.     if(i==0)
84.         l=1;
85.     else
86.         l=0;
87.     k1:while((st[i][0]!=st[k][j])&&(k<n))
88.     {
89.         if(st[k][j]!='\0')
90.         {
91.             k++;
92.             j=2;
93.         }
94.         j++;
95.     }
96.     j=j+1;
97.     if(st[i][0]==st[k][j-1])
98.     {
99.         if((st[k][j]!='|')&&(st[k][j]!='\0'))
100.        {
101.            a=0;
102.            if(!((st[k][j]>64)&&(st[k][j]<91)))
103.            {
104.                for(m=0;m<l;m++)
105.                {
106.                    if(fol[i][m]==st[k][j])
107.                        goto q3;
108.                }
109.                fol[i][l]=st[k][j];
110.                l++;
111.                q3: int iiii=0;
112.            }
113.            else
114.            {
115.                while(st[k][j]!=st[a][0])
116.                {
117.                    a++;
118.                }
119.                p=0;

```

```

120.         while(ft[a][p]!='\0')
121.         {
122.             if(ft[a][p]!='@')
123.             {
124.                 for(m=0;m<l;m++)
125.                 {
126.                     if(fol[i][m]==ft[a][p])
127.                         goto q2;
128.                 }
129.                 fol[i][l]=ft[a][p];
130.                 l=l+1;
131.             }
132.             else
133.                 e=1;
134.             q2:p++;
135.         }
136.         if(e==1)
137.         {
138.             e=0;
139.             goto a1;
140.         }
141.     }
142. }
143. else
144. {
145.     a1:c=0;
146.     a=0;
147.     while(st[k][0]!=st[a][0])
148.     {
149.         a++;
150.     }
151.     while((fol[a][c]!='\0')&&(st[a][0]!=st[i][0]))
152.     {
153.         for(m=0;m<l;m++)
154.         {
155.             if(fol[i][m]==fol[a][c])
156.                 goto q1;
157.         }
158.         fol[i][l]=fol[a][c];
159.         l++;
160.         q1:c++;
161.     }
162. }
163. goto k1;
164. }
165. fol[i][l]='\0';
166. }
167. printf("\n\n");
168. for(i=0;i<n;i++){
169.     printf("FOLLOW[%c]={ %c",st[i][0],fol[i][0]);
170.     int g=1;
171.     while(ft[i][g]!='\0'){
172.         printf(" ,%c", fol[i][g]);
173.         g++;
174.     }
175.     printf(" }\n");
176. }
177. printf("\n");
178. s=0;
179. printf("PREDICTIVE PARSER TABLE\n");
180. for(i=0;i<n;i++)

```

```

181.         {
182.             j=3;
183.             while(st[i][j]!='\0')
184.             {
185.                 if((st[i][j-1]=='|')||(j==3))
186.                 {
187.                     for(p=0;p<=2;p++)
188.                     {
189.                         fin[s][p]=st[i][p];
190.                     }
191.                     t=j;
192.                     for(p=3;((st[i][j]!='|')&&(st[i][j]!='\0'));p++)
193.                     {
194.                         fin[s][p]=st[i][j];
195.                         j++;
196.                     }
197.                     fin[s][p]='\0';
198.                     if(st[i][k]=='@')
199.                     {
200.                         b=0;
201.                         a=0;
202.                         while(st[a][0]!=st[i][0])
203.                         {
204.                             a++;
205.                         }
206.                         while(fol[a][b]!='\0')
207.                         {
208.                             printf("M[%c,%c] = %s\n",st[i][0],fol[a][b],fin[s]);
209.                             b++;
210.                         }
211.                     }
212.                     else if(!((st[i][t]>64)&&(st[i][t]<91)))
213.                     printf("M[%c,%c] = %s\n",st[i][0],st[i][t],fin[s]);
214.                     else
215.                     {
216.                         b=0;
217.                         a=0;
218.                         while(st[a][0]!=st[i][3])
219.                         {
220.                             a++;
221.                         }
222.                         while(ft[a][b]!='\0')
223.                         {
224.                             printf("M[%c,%c] = %s\n",st[i][0],ft[a][b],fin[s]);
225.                             b++;
226.                         }
227.                     }
228.                     s++;
229.                 }
230.                 if(st[i][j]=='|')
231.                     j++;
232.             }
233.         }
234.         getch();
235.     }

```

## Output:

```
G:\AUST4.1\CompilerLab\Lab6\test1.exe
enter the no. of coordinates
4
enter the productions in a grammar
S->aXd
X->YZ
Y->b|#
Z->cX|#

FIRST[S]={ a }
FIRST[X]={ b ,# }
FIRST[Y]={ b ,# }
FIRST[Z]={ c ,# }

FOLLOW[S]={ $ }
FOLLOW[X]={ d , }
FOLLOW[Y]={ c ,# }
FOLLOW[Z]={ d , }

PREDICTIVE PARSER TABLE
M[S,a] = S->aXd
M[X,b] = X->YZ
M[X,#] = X->YZ
M[Y,b] = Y->b
M[Y,#] = Y->#
M[Z,c] = Z->cX
M[Z,#] = Z->#
```