

# 18CSC304J/ Compiler Design

Submitted By:- ANANNYA P. NEOG (RA1911003010367)

## Exp-2: Conversion from Regular Expression to NFA

**Aim:-** To convert a Regular Function into NFA

### **Code:-**

```
#include<stdio.h>
#include<string.h>
int main()
{
    char reg[20];
    int q[20][3],i,j,len,a,b;
    for(a=0;a<20;a++)
    {
        for(b=0;b<3;b++)
        {
            q[a][b]=0;
        }
    }
    printf("Enter Regular Expression:\n");
    scanf("%s",reg);
    len=strlen(reg);
    i=0;
    j=1;
    while(i<len)
    {
        if(reg[i]=='a'&&reg[i+1]!='|'&&reg[i+1]!='*')
        {
            q[j][0]=j+1;
            j++;
        }
        if(reg[i]=='b'&&reg[i+1]!='|'&&reg[i+1]!='*')
        {
            q[j][1]=j+1;
            j++;
        }
        if(reg[i]=='e'&&reg[i+1]!='|'&&reg[i+1]!='*')
        {
            q[j][2]=j+1;
            j++;
        }
        if(reg[i]=='a'&&reg[i+1]=='|'&&reg[i+2]=='b')
        {
            q[j][2]=((j+1)*10)+(j+3);
            j++;
            q[j][0]=j+1;
        }
    }
}
```

```

        j++;
        q[j][2]=j+3;
        j++;
        q[j][1]=j+1;
        j++;
        q[j][2]=j+1;
        j++;
        i=i+2;
    }
    if(reg[i]=='b'&&reg[i+1]=='|'&&reg[i+2]=='a')
    {
        q[j][2]=((j+1)*10)+(j+3);
        j++;
        q[j][1]=j+1;
        j++;
        q[j][2]=j+3;
        j++;
        q[j][0]=j+1;
        j++;
        q[j][2]=j+1;
        j++;
        i=i+2;
    }
    if(reg[i]=='a'&&reg[i+1]=='*')
    {
        q[j][2]=((j+1)*10)+(j+3);
        j++;
        q[j][0]=j+1;
        j++;
        q[j][2]=((j+1)*10)+(j-1);
        j++;
    }
    if(reg[i]=='b'&&reg[i+1]=='*')
    {
        q[j][2]=((j+1)*10)+(j+3);
        j++;
        q[j][1]=j+1;
        j++;
        q[j][2]=((j+1)*10)+(j-1);
        j++;
    }
    if(reg[i]=='')&&reg[i+1]=='*')
    {
        q[0][2]=((j+1)*10)+1;
        q[j][2]=((j+1)*10)+1;
        j++;
    }
    i++;
}

```

```

printf("Transition function \n");
for(i=0;i<=j;i++)
{
    if(q[i][0]!=0)
        printf("\n q[%d,a]-->%d",i,q[i][0]);
    if(q[i][1]!=0)
        printf("\n q[%d,b]-->%d",i,q[i][1]);
    if(q[i][2]!=0)
    {
        if(q[i][2]<10)
            printf("\n q[%d,e]-->%d",i,q[i][2]);
        else
            printf("\n q[%d,e]-->%d & %d",i,q[i][2]/10,q[i][2]%10);
    }
}
return 0;
}

```

```

1  #include<stdio.h>
2  #include<string.h>
3  int main()
4  {
5      char reg[20];
6      int q[20][3],i,j,len,a,b;
7      for(a=0;a<20;a++)
8      {
9          for(b=0;b<3;b++)
10         {
11             q[a][b]=0;
12         }
13     }
14     printf("Enter Regular Expression:\n");
15     scanf("%s",reg);
16     len=strlen(reg);
17     i=0;
18     j=1;
19     while(i<len)
20     {
21         if(reg[i]=='a'&&reg[i+1]!='|'&&reg[i+1]!='*')
22         {
23             q[j][0]=j+1;
24             j++;
25         }
26         if(reg[i]=='b'&&reg[i+1]!='|'&&reg[i+1]!='*')
27         {
28             q[j][1]=j+1;
29             j++;
30         }
31         if(reg[i]=='e'&&reg[i+1]!='|'&&reg[i+1]!='*')
32         {
33             q[j][2]=j+1;
34             j++;
35         }
36     }
37 }

```

```
mpiler
Run Debug Stop Share Save {} Beautify
Language C
main.c
36 if(reg[i]=='a'&&reg[i+1]=='|'&&reg[i+2]=='b')
37 {
38     q[j][2]=((j+1)*10)+(j+3);
39     j++;
40     q[j][0]=j+1;
41     j++;
42     q[j][2]=j+3;
43     j++;
44     q[j][1]=j+1;
45     j++;
46     q[j][2]=j+1;
47     j++;
48     i=i+2;
49 }
50 if(reg[i]=='b'&&reg[i+1]=='|'&&reg[i+2]=='a')
51 {
52     q[j][2]=((j+1)*10)+(j+3);
53     j++;
54     q[j][1]=j+1;
55     j++;
56     q[j][2]=j+3;
57     j++;
58     q[j][0]=j+1;
59     j++;
60     q[j][2]=j+1;
61     j++;
62     i=i+2;
63 }
64 if(reg[i]=='a'&&reg[i+1]=='*')
65 {
66     q[j][2]=((j+1)*10)+(j+3);
67     j++;
68     q[j][0]=j+1;
69     j++;
70     q[j][2]=((j+1)*10)+(j-1);
71     j++;
72 }
```

```
mpiler
Run Debug Stop Share Save {} Beautify
Language C
main.c
73 if(reg[i]=='b'&&reg[i+1]=='*')
74 {
75     q[j][2]=((j+1)*10)+(j+3);
76     j++;
77     q[j][1]=j+1;
78     j++;
79     q[j][2]=((j+1)*10)+(j-1);
80     j++;
81 }
82 if(reg[i]=='|'&&reg[i+1]=='*')
83 {
84     q[0][2]=((j+1)*10)+1;
85     q[j][2]=((j+1)*10)+1;
86     j++;
87 }
88 i++;
89 }
90 printf("Transition function \n");
91 for(i=0;i<=j;i++)
92 {
93     if(q[i][0]!=0)
94         printf("\n q[%d,a]-->%d",i,q[i][0]);
95     if(q[i][1]!=0)
96         printf("\n q[%d,b]-->%d",i,q[i][1]);
97     if(q[i][2]!=0)
98     {
99         if(q[i][2]<10)
100             printf("\n q[%d,e]-->%d",i,q[i][2]);
101         else
102             printf("\n q[%d,e]-->%d & %d",i,q[i][2]/10,q[i][2]%10);
103     }
104 }
105 return 0;
106 }
```

## Output:-

### (Output 1)

```
input
Enter Regular Expression:
a*
Transition function

q[1,e]-->2 & 4
q[2,a]-->3
q[3,e]-->4 & 2

...Program finished with exit code 0
Press ENTER to exit console.
```

### (Output 2)

```
input
Enter Regular Expression:
a*ab
Transition function

q[1,e]-->2 & 4
q[2,a]-->3
q[3,e]-->4 & 2
q[4,a]-->5
q[5,b]-->6

...Program finished with exit code 0
Press ENTER to exit console.
```