

```

8
9 #include <stdio.h>
10 #include <string.h>
11 int F(char symbol)
12 {
13     switch(symbol)
14     {
15         case '+':
16         case '-': return 2;
17         case '*':
18         case '/': return 4;
19         case '^':
20         case '$': return 5;
21         case '(': return 0;
22         case '#': return -1;
23         default : return 8;
24     }
25 }
26 int G(char symbol)
27 {
28     switch(symbol)
29     {
30         case '+':
31         case '-': return 1;
32         case '*':
33         case '/': return 3;
34         case '^':
35         case '$': return 6;
36         case '(': return 9;
37         case ')': return 0;
38         default : return 7;
39     }
40 }
41 void infix_to_postfix(char infix[], char postfix[])
42 {
43     int top, i, j;
44     char s[30], symbol;
45     top = -1;
46     s[++top] = '#';
47     j = 0;
48     for(i = 0; i < strlen(infix); i++)
49     {
50         symbol = infix[i];
51         while(F(s[top]) > G(symbol))
52         {
53             postfix[j] = s[top--];

```

```

30     case '+':
31     case '-':return 1;|
32     case '*':
33     case '/':return 3;
34     case '^':
35     case '$': return 6;
36     case '(': return 9;
37     case ')': return 0;
38     default : return 7;
39 }
40 }
41 void infix_to_postfix(char infix[],char postfix[])
42 {
43     int top,i,j;
44     char s[30],symbol;
45     top=-1;
46     s[++top]='#';
47     j=0;
48     for(i=0;i< strlen(infix);i++)
49     {
50         symbol = infix[i];
51         while(F(s[top]) > G(symbol))
52         {
53             postfix[j] = s[top--];
54             j++;
55         }
56         if(F(s[top])!= G(symbol))
57             s[++top]=symbol;
58         else
59             top--;
60     }
61     while(s[top] != '#')
62     {
63         postfix[j++] = s[top--];
64     }
65     postfix[j] = '\0';
66 }
67 void main()
68 {
69     char infix[20];
70     char postfix[20];
71     printf("Enter valid infix equation\n");
72     scanf("%s",infix);
73     infix_to_postfix(infix,postfix);
74     printf("postfix equation is \n");
75     printf("%s\n",postfix);
76 }
77

```



Enter valid infix equation

$((a+(b-c)*d)^E+F)$

postfix equation is

$abc-d*+E^F+$

...Program finished with exit code 0

Press ENTER to exit console.

```
40 }  
41 void infix_to_postfix(char infix[],char postfix[])  
42 {  
43     int top,i,j;  
44     char s[30],symbol;  
45     top=-1;  
46     s[++top]='#';  
47     j=0;  
48     for(i=0;i< strlen(infix);i++)
```

Enter valid infix equation

X^Y^Z-M+N+P/Q

postfix equation is

XYZ^^M-N+PQ/+

...Program finished with exit code 0

Press ENTER to exit console.

Enter valid infix equation

$a^b*c-d+e/f/(g+h)$

postfix equation is

$ab^c*d-ef/gh+/+$

...Program finished with exit code 0

Press ENTER to exit console.