```c
#include<stdio.h>
#include<string.h>
int F(char symbol)
{
    switch(symbol)
    {
        case '+':
        case '-':return 2;
        case '*':
        case '/':return 4;
        case '^':
        case '$':return 5;
        case '(':return 0;
        case '#': return -1;
        default : return 8;
    }
}
int G(char symbol)
{
    switch(symbol)
    {
    case '+':
    case '-':return 1;
    case '*':
    case '/':return 3;
    case '^':
    case '&':return 6;
    case '(':return 9;
    case ')':return 0;
    default:return 7;
    }
}
void infix_postfix(char infix[],char postfix[])
{
        int top,j,i;
        char Stack[30],symbol;
        top=-1;
        Stack[++top]='#';
```

```c
31            }
32      }
33    void infix_postfix(char infix[],char postfix[])
34    {
35            int top,j,i;
36            char Stack[30],symbol;
37            top=-1;
38            Stack[++top]='#';
39            j=0;
40            for(i=0;i<strlen(infix);i++)
41            {
42              symbol=infix[i];
43              while(F(Stack[top])>G(symbol))
44              {
45                postfix[j]=Stack[top--];
46                j++;
47              }
48              if(F(Stack[top])!=G(symbol))
49              Stack[++top]=symbol;
50              else
51              top--;
52            }
53            while(Stack[top]!='#')
54            {
55                postfix[j++]=Stack[top--];
56            }
57            postfix[j]='\0';
58    }
59    int main()
60    {
61        char infix[20];
62        char postfix[20];
63        printf("Enter the infix expression from user: \n");
64        scanf("%s",infix);
65        infix_postfix(infix,postfix);
66        printf("The postfix expression is : \n");
67        printf("%s\n",postfix);
68        return 0;
```

```c
{
    int top,j,i;
    char Stack[30],symbol;
    top=-1;
    Stack[++top]='#';
    j=0;
    for(i=0;i<strlen(infix);i++)
    {
        symbol=infix[i];
        while(F(Stack[top])>G(symbol))
        {
            postfix[j]=Stack[top--];
            j++;
        }
        if(F(Stack[top])!=G(symbol))
        Stack[++top]=symbol;
        else
        top--;
    }
    while(Stack[top]!='#')
    {
        postfix[j++]=Stack[top--];
    }
    postfix[j]='\0';
}
int main()
{
    char infix[20];
    char postfix[20];
    printf("Enter the infix expression from user: \n");
    scanf("%s",infix);
    infix_postfix(infix,postfix);
    printf("The postfix expression is : \n");
    printf("%s\n",postfix);
    return 0;
}
```

```
Enter the infix expression from user:
a^b*c-d+e/f/(g+h)
The postfix expression is :
ab^c*d-ef/gh+/+


------------------
(program exited with code: 0)

Press any key to continue . . .
```