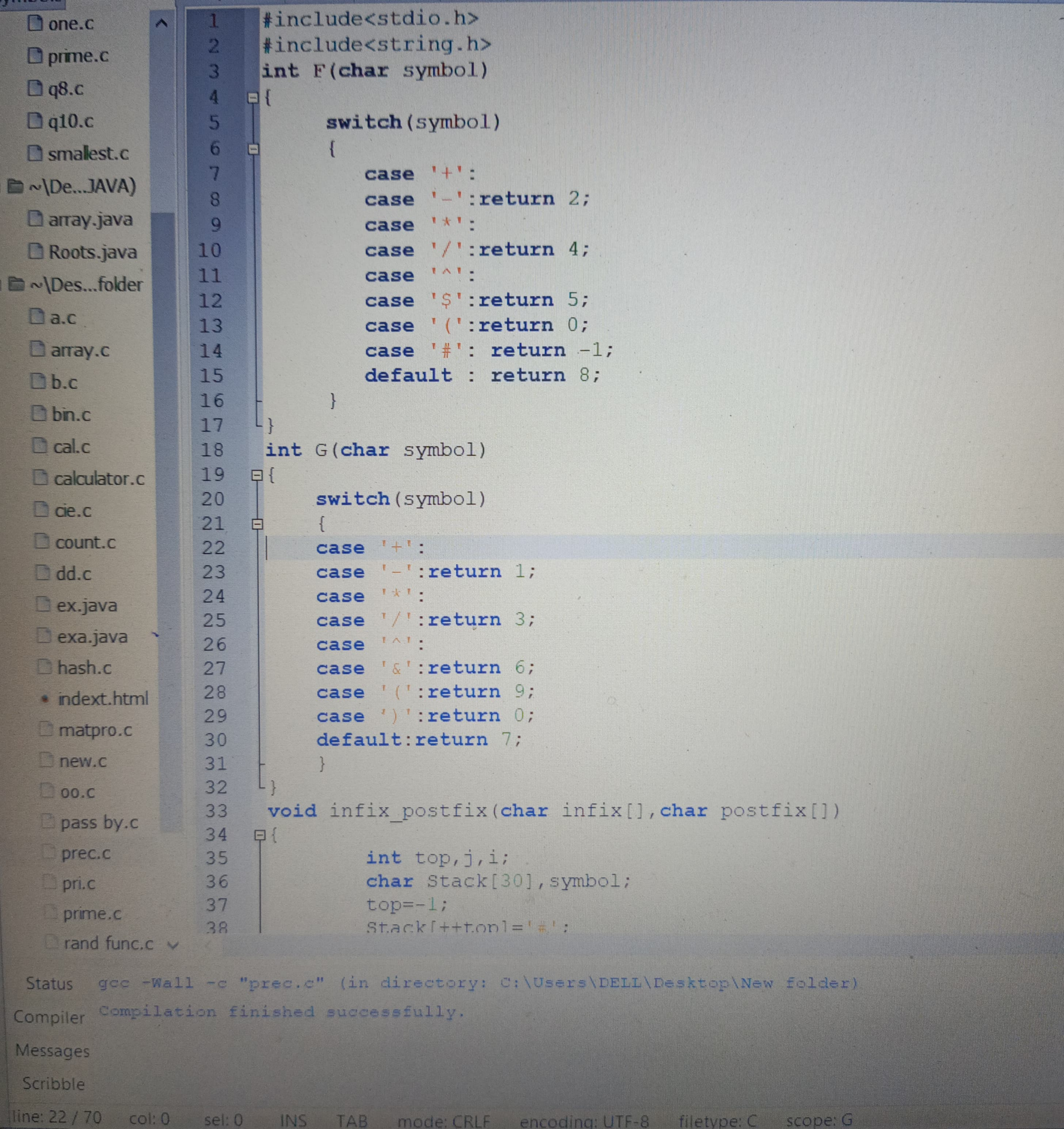
C-Program to convert Infix expression to Postfix expression:-

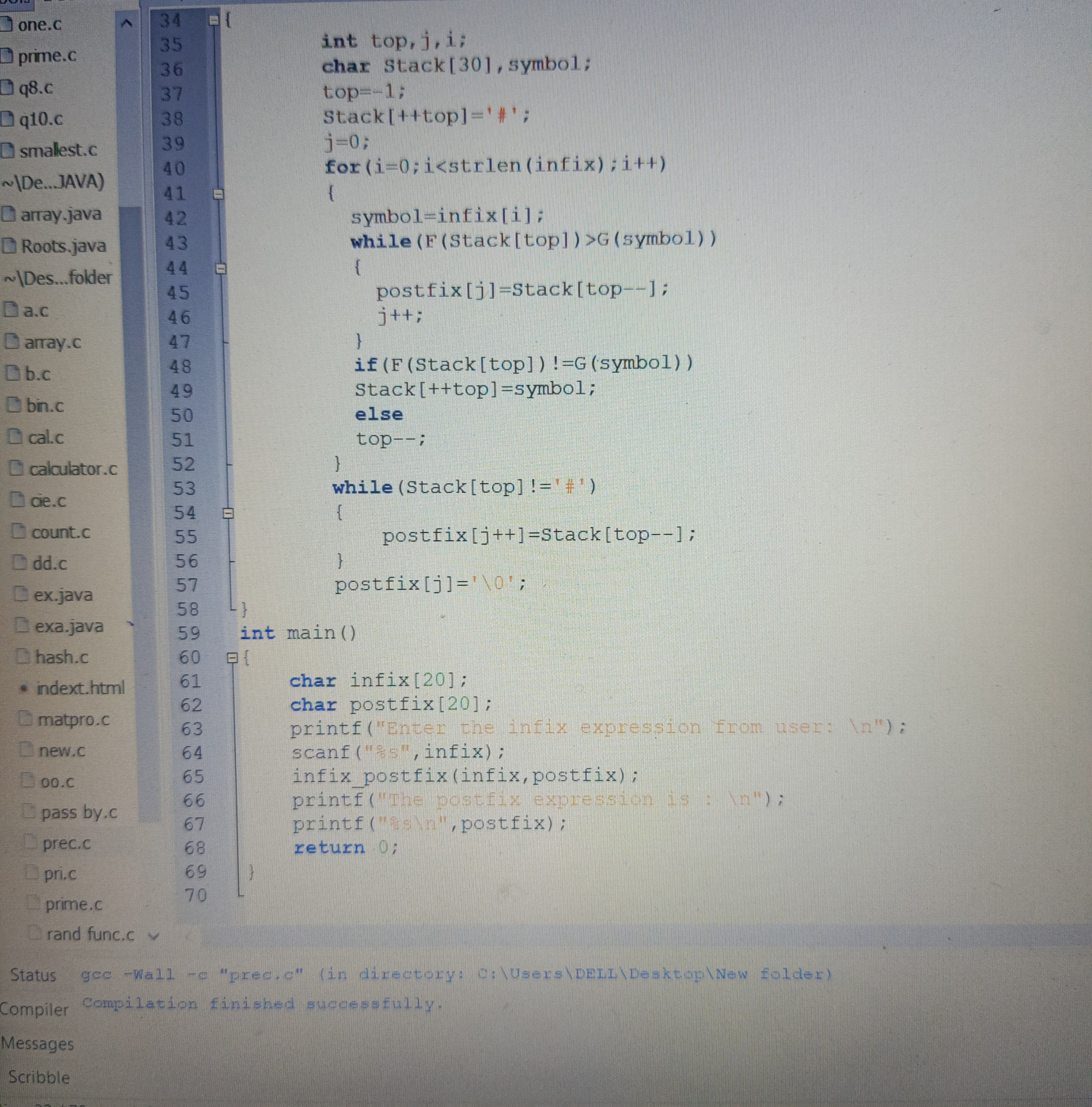
```
# include<stdio.h>
# include<string.h>
 int F(char symbol)
{

   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,i,j;
   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;
  }
```



```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]='#';
```

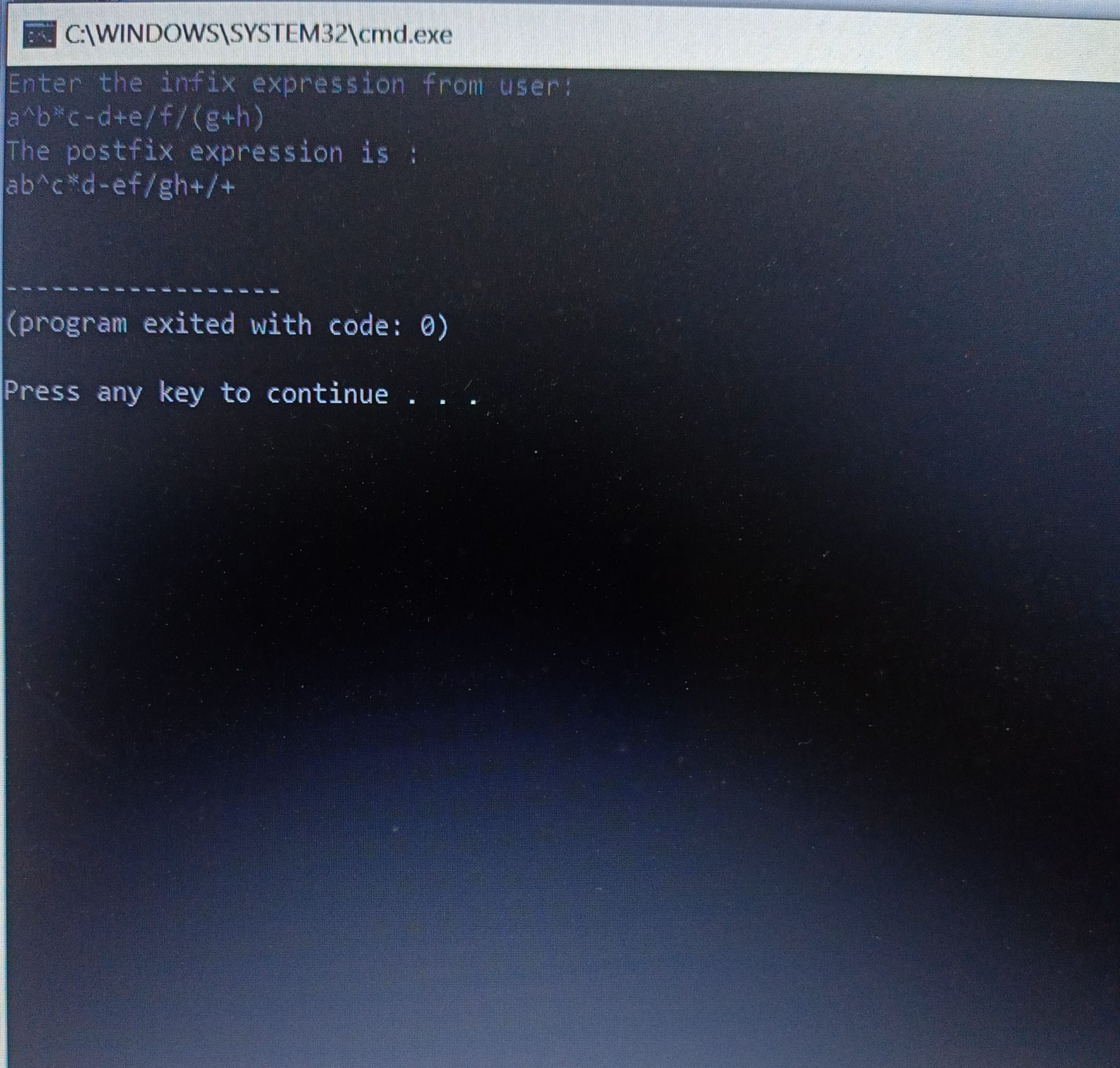
Status   gcc -Wall -c "prec.c" (in directory: C:\Users\DELL\Desktop\New folder)
Compiler   Compilation finished successfully.
Messages
Scribble

line: 22 / 70   col: 0   sel: 0   INS   TAB   mode: CRLF   encoding: UTF-8   filetype: C   scope: G



```
{
        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;
}
```

```
gcc -Wall -c "prec.c" (in directory: C:\Users\DELL\Desktop\New folder)
Compilation finished successfully.
```



```
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 . . .
```