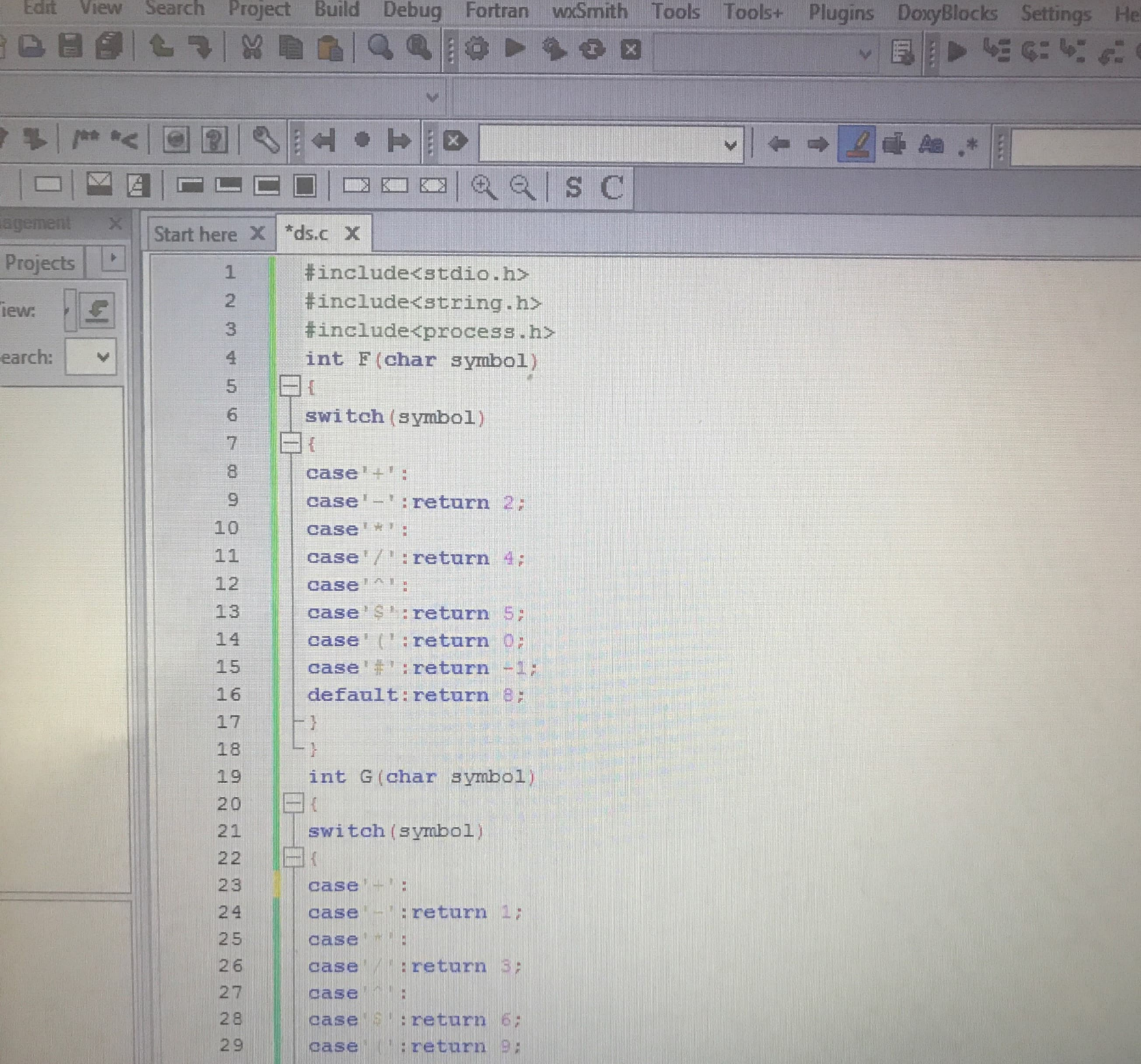

Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings
Projects
Start here
*ds.c
#include<stdio.h>
#include<string.h>
#include<process.h>
int F(char symbol)
{
switch(symbol)
{
case'+':
case'-':return 2;
case'*':
case'/':return 4;
case'^':
case'$':return 5;
case'(':return 0;
case'#':return -1;
default:return 8;
}
}
int G(char symbol)
{
switch(symbol)
{
case'+':
case'-':return 1;
case'*':
case'/':return 3;
case'^':
case'$':return 6;
case'(':return 9;

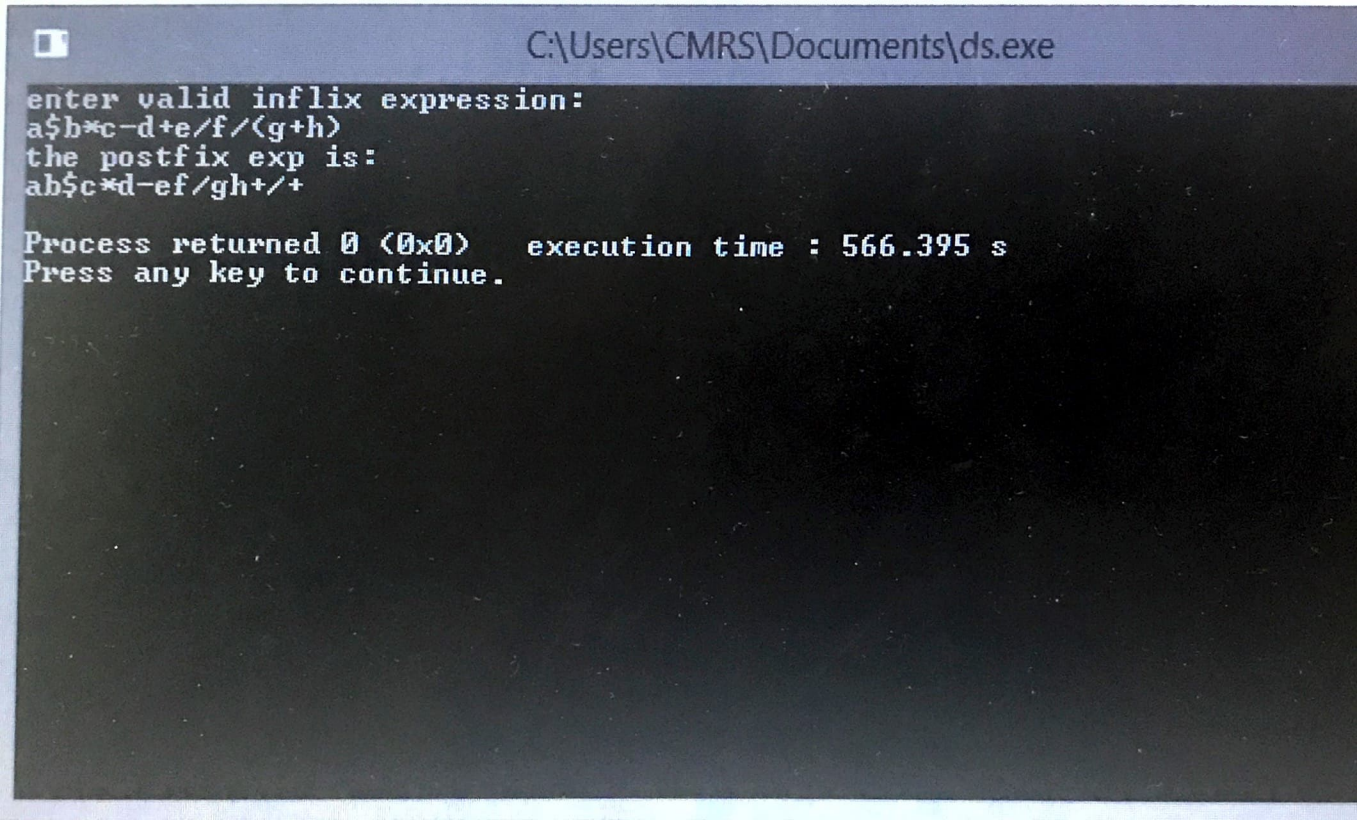
Start here X *ds.c X

```
21      switch(symbol)
22      {
23      case'+':
24      case'-':return 1;
25      case'*':
26      case'/':return 3;
27      case'^':
28      case'$':return 6;
29      case'(':return 9;
30      case')':return 0;
31      default:return 7;
32      }
33      }
34      void infix_postfix(char infix[],char postfix[])
35      {
36      int top,i,j;
37      char s[30],symbol;
38      top=-1;
39      s[++top]='#';
40      j=0;
41      for(i=0;i<strlen(infix);i++)
42      {
43      symbol=infix[i];
44      while(F(s[top])>G(symbol))
45      {
46      postfix[j]=s[top--];
47      j++;
48      }
49      if(F(s[top])!=G(symbol))
```

Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

Start here X *ds.c X

```
41      for(i=0;i<strlen(infix);i++)
42      {
43      symbol=infix[i];
44      while(F(s[top])>G(symbol))
45      {
46      postfix[j]=s[top--];
47      j++;
48      }
49      if(F(s[top])!=G(symbol))
50      s[++top]=symbol;
51      else
52      top--;
53      }
54      while(s[top]!='#')
55      {
56      postfix[j++]=s[top--];
57      }
58      postfix[j]='\0';
59      }
60      void main()
61      {
62      char infix[20];
63      char postfix[20];
64      printf("enter valid inflix expression:\n");
65      scanf("%s",infix);
66      infix_postfix(infix,postfix);
67      printf("the postfix exp is:\n");
68      printf("%s\n",postfix);
69      }
```


C:\Users\CMRS\Documents\ds.exe
enter valid inflix expression:
a$b*c-d+e/f/(g+h)
the postfix exp is:
ab$c*d-ef/gh+/+
Process returned 0 (0x0)   execution time : 566.395 s
Press any key to continue.

PAGE NO.
DATE / /

Lab-2 pgm

Conversion from Infix to post fix expression:

```
#include<stdio.h>
#include<string.h>
#include<process.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 s[30], symbol;
    top = -1;
 s[++top] = '#';
   j = 0;
for (i=0 ; i<strlen(infix); i++)
{
 symbol = infix[i];
   while (F(s[top]) > G(symbol))
   {
    postfix[j] = s[top--];
    j++;
   }
    if (F(s[top]) != G(symbol))
      s[++top] = symbol;
      else
   top--; }
   while (s[top] != '#')
   { postfix[j++] = s[top--];
   }
     postfix[j] = '\0'; }
   void main()
  { char infix[20], postfix[20];
 printf ("Enter valid Infix exp: \n");
  scanf ("%s", infix);
Infix-postfix(infix, postfix);
printf ("The postfix exp is: \n");
printf ("%s \n", postfix); }
```