

C:\Users\Nikita\Desktop\programs\prefixevaluate.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help



Project Classes [*] prefixevaluate.cpp infixprefix.cpp gcdrecursion.cpp factorialrecursion.cpp

```
1  #include<stdio.h>
2  #include<math.h>
3  #include<string.h>
4  #include<ctype.h>
5  double compute(char symbol,double op1,double op2)
6  {
7      switch(symbol)
8      {
9          case '+':return op1+op2;
10         case '-':return op1-op2;
11         case '*':return op1*op2;
12         case '/':return op1/op2;
13         case '$':
14         case '^':return pow(op1,op2);
15     }
16 }
17
18 int main()
19 {
20     double s[20];
21     double res;
22     double op1,op2;
23     int top,i;
24     char prefix[20],symbol;
25     printf("Enter the prefix expression:\n");
26     scanf("%s",prefix);
27     top=-1;
28     strrev(prefix);
29     for(i=0;i<strlen(prefix);i++)
30     {
31         symbol=prefix[i];
32         if(isdigit(symbol))
33             s[++top]=symbol-'0';
34         else
35         {
36             op2=s[top--];
37             op1=s[top--];
38             res=compute(symbol,op1,op2);
39             s[++top]=res;
40         }
41     }
```

C:\Users\Nikita\Desktop\programs\prefixevaluate.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

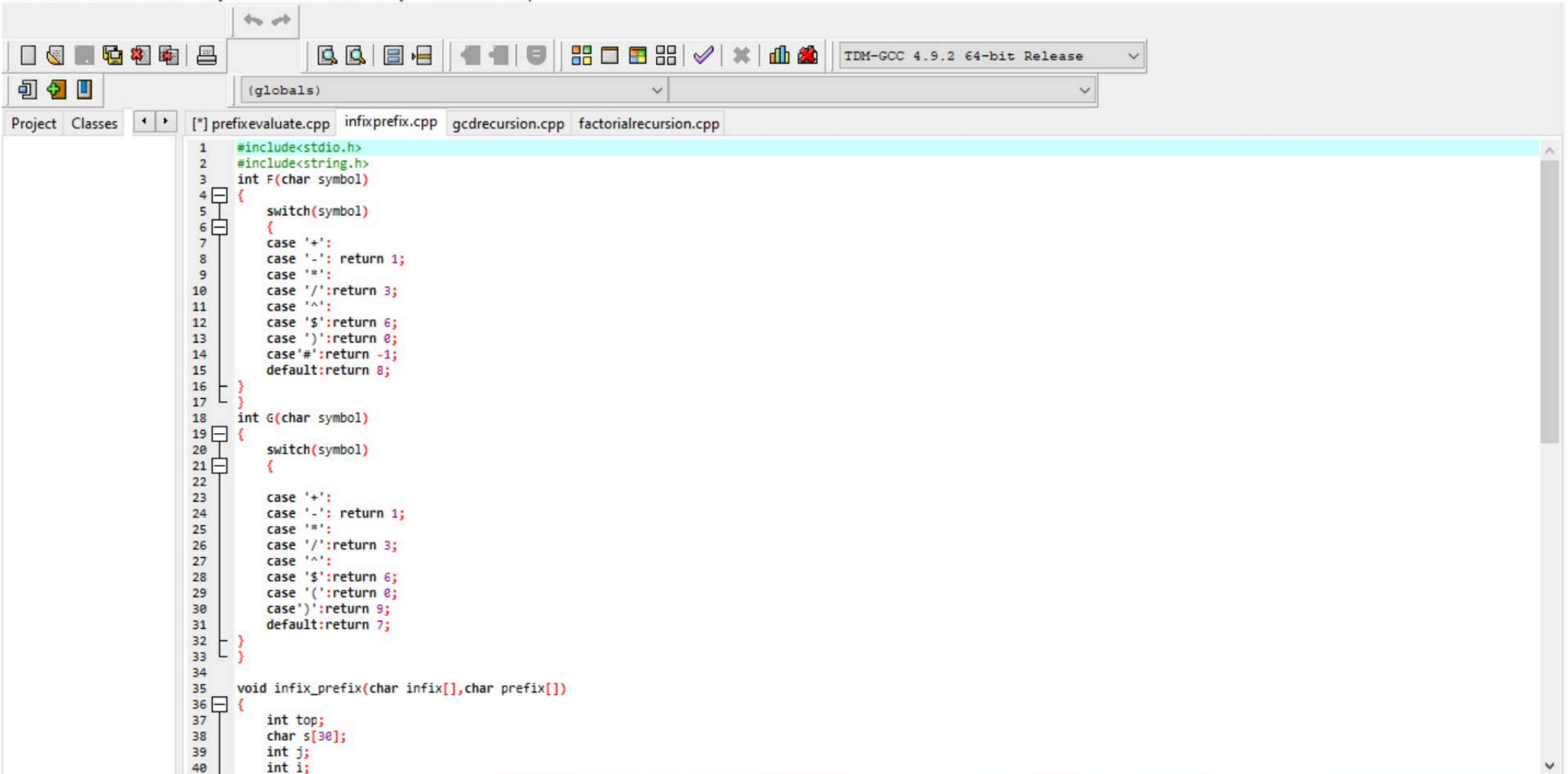


Project Classes [*] prefixevaluate.cpp infixprefix.cpp gcdrecursion.cpp factorialrecursion.cpp

```
6 {
7     switch(symbol)
8     {
9         case '+':return op1+op2;
10        case '-':return op1-op2;
11        case '*':return op1*op2;
12        case '/':return op1/op2;
13        case '$':
14        case '^':return pow(op1,op2);
15    }
16 }
17
18 int main()
19 {
20     double s[20];
21     double res;
22     double op1,op2;
23     int top,i;
24     char prefix[20],symbol;
25     printf("Enter the prefix expression:\n");
26     scanf("%s",prefix);
27     top=-1;
28     strrev(prefix);
29     for(i=0;i<strlen(prefix);i++)
30     {
31         symbol=prefix[i];
32         if(isdigit(symbol))
33             s[++top]=symbol-'0';
34         else
35         {
36             op2=s[top--];
37             op1=s[top--];
38             res=compute(symbol,op1,op2);
39             s[++top]=res;
40         }
41     }
42     res=s[top--];
43     printf("result=%f\n",res);
44 }
45 }
```

Type here to search



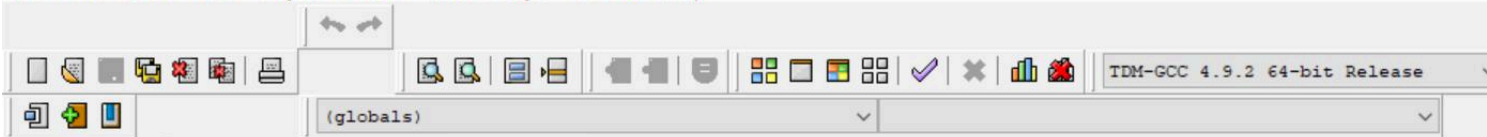


```
1 #include<stdio.h>
2 #include<string.h>
3 int F(char symbol)
4 {
5     switch(symbol)
6     {
7         case '+':
8         case '-': return 1;
9         case '*':
10        case '/':return 3;
11        case '^':
12        case '$':return 6;
13        case ')':return 0;
14        case '#':return -1;
15        default:return 8;
16    }
17 }
18 int G(char symbol)
19 {
20     switch(symbol)
21     {
22
23         case '+':
24         case '-': return 1;
25         case '*':
26         case '/':return 3;
27         case '^':
28         case '$':return 6;
29         case '(':return 0;
30         case ')':return 9;
31         default:return 7;
32     }
33 }
34
35 void infix_prefix(char infix[],char prefix[])
36 {
37     int top;
38     char s[30];
39     int j;
40     int i;
```



C:\Users\Nikita\Desktop\programs\infixprefix.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help



Project Classes [*] prefixevaluate.cpp infixprefix.cpp gcdrecursion.cpp factorialrecursion.cpp

```
39  int j;
40  int i;
41  char symbol;
42  top=-1;
43  s[++top]='#';
44  j=0;
45  strrev(infix);
46  for(i=0;i<strlen(infix);i++)
47  {
48      symbol=infix[i];
49
50      while(F(s[top])>G(symbol))
51      {
52          prefix[j]=s[top--];
53          j++;
54      }
55      if(F(s[top])!=G(symbol))
56          s[++top]=symbol;
57      else
58          top--;
59  }
60
61  while(s[top]!='#')
62  {
63      prefix[j++]=s[top--];
64  }
65
66  prefix[j]='\0';
67  strrev(prefix);
68  }
69
70  int main()
71  {
72      char infix[20];
73      char prefix[20];
74      printf("Enter the valid infix expression\n");
75      scanf("%s",infix);
76      infix_prefix(infix,prefix);
77      printf("The prefix expression is\n");
78      printf("%s\n",prefix);
```

```
45     strrev(infix);
46     for(i=0;i<strlen(infix);i++)
47     {
48         symbol=infix[i];
49
50         while(F(s[top])>G(symbol))
51         {
52             prefix[j]=s[top--];
53             j++;
54         }
55         if(F(s[top])!=G(symbol))
56             s[++top]=symbol;
57         else
58             top--;
59     }
60
61     while(s[top]!='#')
62     {
63         prefix[j++]=s[top--];
64     }
65     prefix[j]='\0';
66     strrev(prefix);
67 }
68
69
70 int main()
71 {
72     char infix[20];
73     char prefix[20];
74     printf("Enter the valid infix expression\n");
75     scanf("%s",infix);
76     infix_prefix(infix,prefix);
77     printf("The prefix expression is\n");
78     printf("%s\n",prefix);
79 }
80
81
82
83
84
```

C:\Users\Nikita\Desktop\programs\gcdrecursion.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help



Project Classes [*] prefixevaluate.cpp infixprefix.cpp gcdrecursion.cpp factorialrecursion.cpp

```
1  #include<stdio.h>
2  int gcd(int n1,int n2);
3  int main()
4  {
5      int n1,n2;
6      printf("Enter two numbers:\n");
7      scanf("%d%d",&n1,&n2);
8      printf("Gcd of two numbers is %d",gcd(n1,n2));
9      return 0;
10 }
11 int gcd(int n1,int n2)
12 {
13     if(n2!=0)
14         return gcd(n2,n1%n2);
15     else
16         return n1;
17 }
18
19
20
21
22
23
```

Type here to search



C:\Users\Nikita\Desktop\programs\factorialrecursion.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help



Project Classes [*] prefixevaluate.cpp infixprefix.cpp gcdrecursion.cpp [*] factorialrecursion.cpp

```
1  #include<stdio.h>
2  int fact(int n)
3  {
4      if(n==0)
5          return 1;
6      else
7          return n*fact(n-1);
8  }
9  int main()
10 {
11     int n;
12     printf("Enter the value of n\n");
13     scanf("%d",&n);
14     printf("The factorial of %d is %d\n",n,fact(n));
15 }
16
17
18
19
20
21
22
23
```

Type here to search

Taskbar showing various application icons (Chrome, File Explorer, etc.) and system tray information including time (18:30) and date (07-10-2020).

C:\Users\Nikita\Desktop\programs\prefixevaluate.exe

Enter the prefix expression:

--+7*45+20

result=-25.000000

Process exited after 88.2 seconds with return value 0

Press any key to continue . . .



Type here to search



ENG

18:38

07-10-2020



C:\Users\Nikita\Desktop\programs\infixprefix.exe

Enter the valid infix expression

(x*y)+z

The prefix expression is

+*xyz

Process exited after 15.16 seconds with return value 0

Press any key to continue . . .



Type here to search



ENG

18:36

07-10-2020



C:\Users\Nikita\Desktop\programs\gcdrecursion.exe

Enter two numbers:

60 120

Gcd of two numbers is 60

Process exited after 7.189 seconds with return value 0

Press any key to continue . . .



Type here to search



ENG

18:36

07-10-2020



C:\Users\Nikita\Desktop\programs\factorialrecursion.exe

Enter the value of n

5

The factorial of 5 is 120

Process exited after 6.011 seconds with return value 0

Press any key to continue . . .



Type here to search



18:34
07-10-2020