

① Infix to postfix expression

USN - 1R19CS004

Date - 9/Nov.

```
#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 'c' : return 0;
```

```
        case '#' : return -1;
```

```
case
```

```
        default : return 8;
```

```
}
```

```
int G(char symbol)
```

```
{    switch(symbol)
```

```
    {    case '+' :
```

```
        case '-' : return 1;
```

```
        case 'x' :
```

```
        case '/' : return 3;
```

```
        case '^' :
```

```
        case '$' : return 6;
```

```
        case 'c' : return 9;
```

```
        case ')' : return 0;
```

```
        default : return 7;
```

```
}
```

```
}
```

```
Void push(char s[20], int *top, char symbol)
```

```
{  
    (*top)++;  
    s[*top] = symbol;  
}
```

```
}
```

```
Void pop(char s[20], int *top)
```

```
{  
    char elem;  
    elem = s[*top];  
    (*top)--;  
    return elem;  
}
```

```
}
```

```
int main()
```

```
{  
    char infix_expr[20], s[20], symbol, postfix[20];
```

```
    int i, top = -1, j = 0;
```

```
    printf("enter the infix expression \n");
```

```
    scanf("%s", &infix_expr);
```

```
    push(s, &top, '#');
```

```
    for(i = 0; i < strlen(infix_expr); i++)
```

```
    {  
        symbol = infix_expr[i];
```

```
        while (F(s[top]) > G(symbol))
```

```
            postfix[j++] = pop(s, &top);
```

```
        if (F(s[top]) != G(symbol))
```

```
            push(s, &top, symbol);
```

```
        else
```

```
            top--;
```

```
    }
```

```
    while (s[top] != '#')
```

```
        postfix[j++] = pop(s, &top);
```

```
    postfix[++j] = '\0';
```

```
    printf("postfix exp is %s", postfix);
```

```
    return 0;
```

```
}
```

② Factorial

```
#include <stdio.h>
```

```
int fact(int n)
{
    if (n == 0 || n == 1)    // Base condition.
        return 1;
    else
        return n * fact(n-1);
}
```

```
int main()
{
    int n;
    printf("enter the value of n\n");
    scanf("%d", &n);
    printf("%d! = %d", n, fact(n));
}
```

③ Tower of Hanoi

```
#include <stdio.h>
```

```
void towerofhanoi(int n, char s, char t, char d)
```

```
{
    if (n == 0)
        return;
    towerofhanoi(n-1, s, d, t);
    printf("Move disc %d from %c to %c\n", n, s, d);
    towerofhanoi(n-1, t, s, d);
}
```

```
}
```

```
int main()
```

```
{
    int n;
    printf("enter the value of n\n");
    scanf("%d", &n);
    printf("n, 'A', 'B', 'C');
    return 0;
}
```

```
}
```


④

④ GCD.

```
#include <stdio.h>
```

```
int gcd(int M, int N)
```

```
{ while(M != N)
```

```
{ if(M > N)
```

```
{ return gcd(M-N, N);
```

```
else
```

```
return gcd(M, N-M);
```

```
}
```

```
}
```

```
int main()
```

```
{ int a, b;
```

```
printf("enter the value of a and b \n");
```

```
scanf("%d %d", &a, &b);
```

```
int result = gcd(a, b);
```

```
printf("gcd of two numbers is %d \n", result);
```

```
return 0;
```

```
}
```

⑤ Fibonacci sequence.

```
#include <stdio.h>
```

```
int fib(int n)
```

```
{ if(n <= 1)
```

```
return n;
```

```
else
```

```
return fib(n-1) + fib(n-2);
```

```
}
```

```
int main()
```

```
{ int n;
```

```
printf("enter the value of n \n");
```

```
scanf("%d", &n);
```

```
printf("Fibonacci number of %d is %d \n", n, fib(n));
```

```
}
```

C:\Users\Lenovo\Documents\infix to postfix.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Debug

(globals)

Project Classes Tower of hanoi.cpp Factorial.cpp gcd.cpp Fibonacci Sequence.cpp infix to postfix.cpp

```
2  #include<stdio.h>
3  #include<string.h>
4  int F( char symbol)
5  {
6      switch(symbol)
7      {
8          case '+':
9          case '-':return 2;
10         case '*':
11         case '/':return 4;
12         case '^':
13         case '$':return 5;
14         case '(':return 0;
15         case '#':return -1;
16         default :return 8;
17     }
18 }
19
20 int G( char symbol)
21 {
22     switch(symbol)
23     {
24         case '+':
25         case '-':return 1;
26         case '*':
27         case '/':return 3;
28         case '^':
29         case '$':return 6;
30         case '(':return 9;
```

Compiler Resources Compile Log Debug Find Results Close

About Compilation

- Output Size: 150.046875 KiB
- Compilation Time: 0.20s

Line: 5 Col: 2 Sel: 0 Lines: 94 Length: 1432 Insert Done parsing in 0.031 seconds

Type here to search

C:\Users\Lenovo\Documents\infix to postfix.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Debug

(globals)

Project Classes Tower of hanoi.cpp Factorial.cpp gcd.cpp Fibonacci Sequence.cpp infix to postfix.cpp

```
23 {
24     case '+':
25     case '-':return 1;
26     case '*':
27     case '/':return 3;
28     case '^':
29     case '$':return 6;
30     case '(':return 9;
31     case ')':return 0;
32     default :return 7;
33 }
34 }
35
36 void push(char S[20],int *top,char symbol)
37 {
38     (*top)++;
39     S[*top]=symbol;
40 }
41
42 char pop(char S[20],int *top)
43 {
44     char elem;
45     elem=S[*top];
46     (*top)--;
47     return elem;
48 }
49
50
51
```

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation

- Output Size: 150.046875 KiB
- Compilation Time: 0.20s

Line: 5 Col: 2 Sel: 0 Lines: 94 Length: 1432 Insert Done parsing in 0.031 seconds

Type here to search



C:\Users\Lenovo\Documents\infix to postfix.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Debug

(globals)

Project Classes Tower of hanoi.cpp Factorial.cpp gcd.cpp Fibonacci Sequence.cpp infix to postfix.cpp

```
53 int main()
54 {
55     char infix_expr[20]; //="a+b";
56     char S[20], symbol, postfix[20];
57     int i, top=-1, j=0;
58     printf("enter infix expression\n");
59     scanf("%s", &infix_expr);
60     //initia
61     push(S, &top, '#');
62     //procedure
63     //read every symbol
64
65     for(i=0; i<strlen(infix_expr); i++)
66     {
67         symbol=infix_expr[i];
68         while(F(S[top]) > G(symbol))
69         {
70             postfix[j++] = pop(S, &top);
71         }
72         if(F(S[top]) != G(symbol))
73         {
74             push(S, &top, symbol);
75         }
76         else
77         {
78             top--;
79         }
80     }
81 }
```

Compiler Resources Compile Log Debug Find Results Close

About Compilation:

- Output Size: 150.046875 KiB
- Compilation Time: 0.20s

Line: 5 Col: 2 Sel: 0 Lines: 94 Length: 1432 Insert Done parsing in 0.031 seconds

Type here to search

C:\Users\Lenovo\Documents\infix to postfix.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Debug

(globals)

Project Classes Tower of hanoi.cpp Factorial.cpp gcd.cpp Fibonacci Sequence.cpp infix to postfix.cpp

```
66 {
67     symbol=infix_expr[i];
68     while(F(S[top]) > G(symbol))
69     {
70         postfix[j++] = pop(S,&top);
71     }
72     if(F(S[top]) != G(symbol))
73     {
74         push(S,&top,symbol);
75     }
76     else
77     {
78         top--;
79     }
80
81
82 }
83 //remaing elements should be put into postfix
84 while(S[top] != '#' )
85 {
86     postfix[j++]=pop(S,&top);           // postfix[j]=pop(S,&top);      j++;
87 }
88
89     postfix[++j]='\0';
90     printf("postfix exp is %s",postfix);
91     return 0;
92 }
93
94
```

Compiler Resources Compile Log Debug Find Results Close

About Compilation

- Output Size: 150.046875 KiB
- Compilation Time: 0.20s

Line: 5 Col: 2 Sel: 0 Lines: 94 Length: 1432 Insert Done parsing in 0.031 seconds

Type here to search

C:\Users\Lenovo\Documents\infix to postfix.exe

enter infix expression

a+(b/c+k)

postfix exp is abc/k++

Process exited after 22.49 seconds with return value 0

Press any key to continue . . .

C:\Users\Lenovo\Documents\Factorial.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Debug

(globals)

Project Classes Tower of hanoi.cpp [*] Factorial.cpp gcd.cpp Fibonacci Sequence.cpp infix to postfix.cpp

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

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation

- Output Size: 151.65625 KiB
- Compilation Time: 0.22s

Line: 18 Col: 1 Sel: 0 Lines: 18 Length: 229 Insert Done parsing in 0.031 seconds

Type here to search

```
C:\Users\Lenovo\Documents\Factorial.exe
Enter the value of n
10
10! = 3628800
-----
Process exited after 51.54 seconds with return value 0
Press any key to continue . . .
```

C:\Users\Lenovo\Documents\Tower of hanoi.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Debug

(globals)

Project Classes Tower of hanoi.cpp Factorial.cpp gcd.cpp Fibonacci Sequence.cpp infix to postfix.cpp

```
1  #include<stdio.h>
2
3  void towerofhanoi(int n,char s,char t,char d)
4  {
5      if(n == 0 )
6          return;
7      //move n-1 disc from source to temp
8
9      towerofhanoi(n-1,s,d,t);
10     printf("Move disc %d from %c to %c\n ",n,s,d);
11     //move n disc from source to destination
12     towerofhanoi(n-1,t,s,d);
13     //move n-1 disc from temp to destination
14 }
15
16 int main()
17 {
18     int n;
19
20     printf("enter the number of discs\n");
21     scanf("%d",&n);
22     towerofhanoi(n,'A','B','C');
23     return 0;
24 }
```

Compiler Resources Compile Log Debug Find Results Close

Abort Completion

- Output Size: 150.0244140625 KiB
- Compilation Time: 0.22s

Line: 24 Col: 2 Sel: 0 Lines: 24 Length: 464 Insert Done parsing in 0 seconds

Type here to search

C:\Users\Lenovo\Documents\Tower of hanoi.exe

enter the number of discs

4

Move disc 1 from A to B

Move disc 2 from A to C

Move disc 1 from B to C

Move disc 3 from A to B

Move disc 1 from C to A

Move disc 2 from C to B

Move disc 1 from A to B

Move disc 4 from A to C

Move disc 1 from B to C

Move disc 2 from B to A

Move disc 1 from C to A

Move disc 3 from B to C

Move disc 1 from A to B

Move disc 2 from A to C

Move disc 1 from B to C

Process exited after 6.217 seconds with return value 0

Press any key to continue . . . █

C:\Users\Lenovo\Documents\gcd.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Debug

(globals)

Project Classes Tower of hanoi.cpp Factorial.cpp gcd.cpp Fibonacci Sequence.cpp infix to postfix.cpp

```
1  #include<stdio.h>
2
3  int gcd(int M,int N)
4  {
5      while(M!=N)
6      {
7          if(M>N)
8          {
9              return gcd(M-N , N);
10             }
11             else
12                 return gcd(M,N-M);
13         }
14     }
15
16     int main() //accept n and m
17     {
18         int a,b;
19         printf("Enter the value of a and b\n");
20         scanf("%d",&a,&b);
21         int result=gcd(a,b);
22         printf("gcd of two numbers is %d\n",result);
23         return 0;
24     }
```

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation

- Output Size: 150.044921875 KiB
- Compilation Time: 0.19s

Line: 1 Col: 1 Sel: 0 Lines: 24 Length: 343 Insert Done parsing in 0 seconds

Type here to search

C:\Users\Lenovo\Documents\gcd.exe

Enter the value of a and b

16

20

gcd of two numbers is 4

Process exited after 12.32 seconds with return value 0

Press any key to continue . . .

C:\Users\Lenovo\Documents\Fibonacci Sequence.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Debug

(globals)

Project Classes Tower of hanoi.cpp Factorial.cpp gcd.cpp Fibonacci Sequence.cpp infix to postfix.cpp

```
1  #include<stdio.h>
2
3  int fib(int n)
4  {
5      if( n <= 1)
6          return n;          // F0=0 F1=1
7      else
8          return fib(n-1)+fib(n-2);
9  }
10 int main()
11 {
12     int n;
13     printf("enter the value of n\n");
14     scanf("%d",&n);
15     printf("fibonacci number of %d is %d",n,fib(n));
16     return 0;
17 }
```

Compiler Resources Compile Log Debug Find Results Close

About Compilation

- Output Size: 150.0244140625 KiB
- Compilation Time: 0.20s

Line: 17 Col: 2 Sel: 0 Lines: 17 Length: 272 Insert Done parsing in 0 seconds

Type here to search




```
C:\Users\Lenovo\Documents\Fibonacci Sequence.exe
enter the value of n
10
fibonacci number of 10 is 55
-----
Process exited after 7.652 seconds with return value 0
Press any key to continue . . .
```