

对象作为参数的由来

```
1  /*
2  对象作为参数
3  Object as a parameter
4  */
5
6  #include "stdafx.h"
7  #include <iostream>
8
9  //创建一个测试类用于创建对象
10 class CTest {
11 public:
12     //构造函数
13     CTest(int n = 1)
14         :m_nTest(n)
15     {
16         std::cout << "CTest()" << std::endl;
17     }
18
19     //析构函数
20     ~CTest()
21     {
22         std::cout << "~CTest()" << std::endl;
23     }
24
25 private:
26     int m_nTest;
27 };
28
29 //这种方式不能直接调用类对象的构造函数，标记为 can't call class
   constructor
30 void foo(int nArg)
31 {
32     nArg = 2;
33 }
34
35 int main()
36 {
```

```
37      /*
38      这种方式不能直接调用类对象的构造函数，标记为 can't call class
    constructor
39      所以，就有了把对象作为函数参数的出现，从而可以调用类构造函数
40      */
41      int n = 1;
42      foo(n);
43
44      return 0;
45  }
```