**Abstract**之前在[(原創) 如何使用for\_each() algorithm? (C/C++) (STL)](http://www.cnblogs.com/oomusou/archive/200)曾經討論過for\_each()，不過當時功力尚淺，只談到了皮毛而已，這次看了effective STL的item 41、43後，對for\_each()又有了更深入的了解，因此做了本篇心得報告。  
  
**Motivation**看到了eXile的[C++中实现 foreach](http://www.cppblog.com/eXile/archive/2007/01/26/18018.html)使用了巨集對foreach做改善，也看到了很多人對STL style的for\_each()做討論，使我想對STL的for\_each()再做了一次研究。  
  
**Introduction**  
學習過STL的container後，想要存取每一個iterator，你一定寫過以下的程式

http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  for(vector<int>::const\_iterator iter = ivec.begin(); iter != ivec.end(); ++iter) {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << \*iter << endl;  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gif1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif3

當時我覺得STL什麼都好，就是以下這一串又臭又長

http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.giffor(vector<int>::const\_iterator iter = ivec.begin(); iter != ivec.end(); ++iter) {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif

若不常寫，一時還會寫不出來，其實若配合container，C++其實不應該這樣寫迴圈，正確的方式該使用for\_each()，語法會變的相當簡單。  
  
for\_each()事實上是個function template，其實做如下[effective STL item 41]

http://www.cnblogs.com/Images/OutliningIndicators/None.giftemplate<typename InputIterator, typename Function>  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifFunction for\_each(InputIterator beg, InputIterator end, Function f) {  
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  while(beg != end)   
http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    f(\*beg++);  
http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif

由以上source可知，for\_each()只能配合global function和function object。  
  
以下我們將對procedure based、object oriented、generics三種paradigm與for\_each()搭配做探討。  
  
**Procedure Based與for\_each()搭配  
1.不傳入參數**

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_GlobalFunction.cpp  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with global function  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 7http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
15http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifvoid printElem(int& elem) {  
16http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  cout << elem << endl;  
17http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
18http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
19http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
20http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
21http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
22http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  for\_each(ivec.begin(), ivec.end(), printElem);  
24http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gif1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif3

23行

http://www.cnblogs.com/Images/OutliningIndicators/None.giffor\_each(ivec.begin(), ivec.end(), printElem);

只需將vector::begin()，vector::end()和global function name傳給for\_each()即可，再也不用for迴圈那種複雜的語法了。   
  
**2.傳入參數**若要傳參數給global function，就不能再只傳global function name而已，必須透過ptr\_fun()這個function adapter將global function轉成function object，然後再用bind2nd()將參數bind成一個function object。

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_GlobalFunctionWithParameter.cpp  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with global function with Parameter  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 7http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <functional>  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
15http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
16http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifvoid printElem(int elem, const char\* prefix) {  
17http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  cout << prefix << elem << endl;  
18http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
19http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
20http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
21http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
22http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
24http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  for\_each(ivec.begin(), ivec.end(), bind2nd(ptr\_fun(printElem), "Element:"));  
25http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:3

**Object Oriented與for\_each()搭配**  
**1.不傳入參數**使用function object

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_FunctionObject.cpp  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with function object  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 7http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
15http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifstruct printElem {  
16http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void operator() (int elem) {  
17http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << elem << endl;  
18http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }   
19http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
20http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
21http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
22http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
24http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
25http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  for\_each(ivec.begin(), ivec.end(), printElem());  
26http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gif1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif3

**2.傳入參數**若使用function object，也可以將參數傳給printElem()，透過constructor的技巧接收參數。

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_FunctionObjectWithParameter.cpp  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with function object with parameter  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 7http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
15http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifstruct printElem {  
16http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  const char\* \_prefix;  
17http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
18http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  printElem(const char\* prefix) : \_prefix(prefix) {}  
19http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
20http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void operator() (int elem) {  
21http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << \_prefix << elem << endl;  
22http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }   
23http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
24http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
25http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
26http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
27http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
28http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
29http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  for\_each(ivec.begin(), ivec.end(), printElem("Element:"));  
30http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

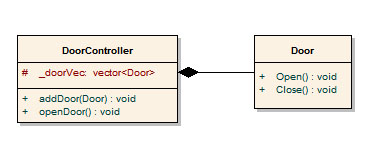
http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:3

function object有很多種寫法，但只要是function object都可以跟for\_each()合作。  
  
**3.member\_function與for\_each()搭配**  
**3.1 不傳入參數**本文的重點來了，在物件導向世界裡，最常用的就是for\_each()配合member function，這該怎麼寫呢?直覺會這樣子寫

http://www.cnblogs.com/Images/OutliningIndicators/None.giffor\_each(\_doorVec.begin(), \_doorVec.end(),&Door::open);

由於global function name本身就是一個pointer，所以想藉由&Door::open傳進一個address，但這樣compile並不會過，正確解法是

http://www.cnblogs.com/Images/OutliningIndicators/None.giffor\_each(\_doorVec.begin(), \_doorVec.end(), mem\_fun\_ref(&Door::open));

透過mem\_fun\_ref()這個function adapter將member function轉成function object。  


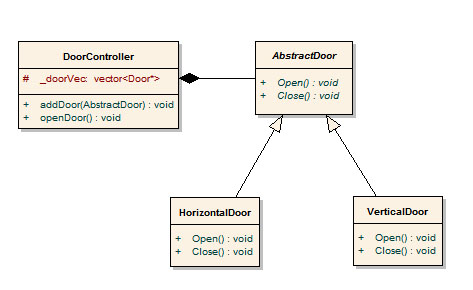
 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_MemberFunctionObject.cpp  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with member function with object  
 7http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 8http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <functional>  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
15http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
16http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass Door {  
17http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
18http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void open() const {  
19http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "open door horizontally" << endl;  
20http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
21http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
22http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void close() const {  
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "close door horizontally" << endl;  
24http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
25http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
26http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
27http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass DoorController {  
28http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifprotected:  
29http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<Door> \_doorVec;  
30http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
31http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
32http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void addDoor(Door aDoor) {  
33http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    \_doorVec.push\_back(aDoor);  
34http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
35http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
36http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void openDoor() const {  
37http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    for\_each(\_doorVec.begin(), \_doorVec.end(), mem\_fun\_ref(&Door::open));  
38http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
39http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
40http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
41http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
42http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  DoorController dc;  
43http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.addDoor(Door());  
44http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.addDoor(Door());  
45http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.openDoor();  
46http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gifopen door horizontally  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifopen door horizontally

37行

http://www.cnblogs.com/Images/OutliningIndicators/None.giffor\_each(\_doorVec.begin(), \_doorVec.end(), mem\_fun\_ref(&Door::open));

值得注意的是，mem\_fun\_ref()用在object的member function。若要搭配多型，vector必須放pointer，也就是得使用object pointer的member function，此時得使用mem\_fun()將member function轉成function object。  


 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_MemberFunctionObjectPointer.cpp  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with member function with object pointer  
 7http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 8http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <functional>  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
15http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
16http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass AbstractDoor {  
17http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
18http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  virtual void open() const {  
19http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "open door horizontally" << endl;  
20http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
21http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
22http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  virtual void close() const {  
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "close door horizontally" << endl;  
24http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
25http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
26http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
27http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass HorizontalDoor : public AbstractDoor {  
28http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
29http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
30http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass VerticalDoor : public AbstractDoor {  
31http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
32http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void open() const {  
33http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "open door vertically" << endl;  
34http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
35http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
36http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void close() const {  
37http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "close door vertically" << endl;  
38http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
39http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
40http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
41http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass DoorController {  
42http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifprotected:  
43http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<AbstractDoor\*> \_doorVec;  
44http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
45http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
46http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void addDoor(AbstractDoor& aDoor) {  
47http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    \_doorVec.push\_back(&aDoor);  
48http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
49http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
50http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void openDoor() const {  
51http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    for\_each(\_doorVec.begin(), \_doorVec.end(), mem\_fun(&AbstractDoor::open));  
52http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
53http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
54http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
55http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
56http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  DoorController dc;  
57http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.addDoor(HorizontalDoor());  
58http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.addDoor(VerticalDoor());  
59http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.openDoor();  
60http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gifopen door horizontally  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifopen door vertically

51行

http://www.cnblogs.com/Images/OutliningIndicators/None.giffor\_each(\_doorVec.begin(), \_doorVec.end(), mem\_fun(&AbstractDoor::open));

使用了mem\_fun()。  
  
**3.2傳入參數**  
問題又來了，若要使member function也傳入參數呢?這時得使用bind2nd將function object和參數bind在一起，變成另外一個新的function object。

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_MemberFunctionObjectPointerWithParameter.cpp  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with member function with object pointer  
 7http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 8http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <functional>  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
15http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
16http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass AbstractDoor {  
17http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
18http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  virtual void open() const {  
19http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "open door horizontally" << endl;  
20http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
21http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
22http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  virtual void close() const {  
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "close door horizontally" << endl;  
24http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
25http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
26http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  virtual void openDoorBy(const char\* name) const {  
27http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << name << " ";  
28http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    open();  
29http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
30http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
31http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
32http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass HorizontalDoor : public AbstractDoor {  
33http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
34http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
35http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass VerticalDoor : public AbstractDoor {  
36http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
37http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void open() const {  
38http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "open door vertically" << endl;  
39http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
40http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
41http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void close() const {  
42http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << "close door vertically" << endl;  
43http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
44http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
45http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
46http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass DoorController {  
47http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifprotected:  
48http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<AbstractDoor\*> \_doorVec;  
49http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
50http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
51http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void addDoor(AbstractDoor& aDoor) {  
52http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    \_doorVec.push\_back(&aDoor);  
53http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
54http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
55http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void openDoor() const {  
56http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    for\_each(\_doorVec.begin(), \_doorVec.end(), bind2nd(mem\_fun(&AbstractDoor::openDoorBy), "John"));  
57http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
58http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
59http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
60http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
61http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  DoorController dc;  
62http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.addDoor(HorizontalDoor());  
63http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.addDoor(VerticalDoor());  
64http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  dc.openDoor();  
65http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

1http://www.cnblogs.com/Images/OutliningIndicators/None.gifJohn open door horizontally  
2http://www.cnblogs.com/Images/OutliningIndicators/None.gifJohn open door vertically

56行

http://www.cnblogs.com/Images/OutliningIndicators/None.giffor\_each(\_doorVec.begin(), \_doorVec.end(), bind2nd(mem\_fun(&AbstractDoor::openDoorBy), "John"));

透過了bind2nd將參數結合後，成為一個新的function object。  
  
**Generics與for\_each()搭配**  
**1.Function Template**  
**1.1不傳入參數**在泛型世界裡，那for\_each()該怎麼配合function template呢?

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_FunctionTemplate.cpp  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with function template  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 7http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
15http://www.cnblogs.com/Images/OutliningIndicators/None.giftemplate<typename T>  
16http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifvoid printElem(T elem) {  
17http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  cout << elem << endl;  
18http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
19http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
20http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
21http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
22http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
24http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  for\_each(ivec.begin(), ivec.end(), printElem<int>);  
25http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  //for\_each(ivec.begin(), ivec.end(), (void(\*)(int))printElem);  
26http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gif1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif3

若使用function template，有兩種寫法  
一種是

http://www.cnblogs.com/Images/OutliningIndicators/None.giffor\_each(ivec.begin(), ivec.end(), printElem<int>);

由於template function需要在compile時確定型別，所以要加上<int>確定為int型別。  
另外一種寫法

http://www.cnblogs.com/Images/OutliningIndicators/None.giffor\_each(ivec.begin(), ivec.end(), (void(\*)(int))printElem);

template function並沒有確定型別，但轉成function pointer時，並須明確轉成int型別的function pointer。  
  
**1.2 傳入參數**若要如function object那樣能傳參數呢?funtion template是可以，不過有些限制，若使用nontype parameter，只能使用以下三種型別  
1.int或enum  
2.pointer：pointer to object，pointer to function，pointer to member。  
3.reference：reference to object，reference to function。

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_FunctionTemplateWithNontypeParameter.cpp  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with function template with nontype parameter  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 7http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
15http://www.cnblogs.com/Images/OutliningIndicators/None.giftemplate<typename T, int i>  
16http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifvoid printElem(T elem) {  
17http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  cout << i << ":"  << elem << endl;  
18http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
19http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
20http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
21http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
22http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
24http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  for\_each(ivec.begin(), ivec.end(), printElem<int, 5>);  
25http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gif5:1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif5:2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif5:3

所以無法如function object那樣可以傳入字串或任意型別，最少在目前ISO C++標準是做不到的。  
  
既然討論了function template，那最具威力的class template是否也能搭配for\_each()?  
  
**2.Class Template**  
**2.1 不傳入參數**

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_ClassTemplate.cpp  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with class template  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 7http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <functional>  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
15http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
16http://www.cnblogs.com/Images/OutliningIndicators/None.giftemplate<typename T>  
17http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass printElem : public unary\_function<T, void> {  
18http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
19http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void operator() (T elem) {  
20http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << elem << endl;  
21http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
22http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
23http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
24http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
25http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
26http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
27http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
28http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  for\_each(ivec.begin(), ivec.end(), printElem<int>());  
29http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gif1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif3

17行

http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass printElem : public unary\_function<T, void> {

因為printElem只接受for\_each()所傳的參數，算是單參數而已，所以public繼承了unary\_function<T,void>，因為for\_each的定義

http://www.cnblogs.com/Images/OutliningIndicators/None.giftemplate <class InputIterator, class UnaryFunction>  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifUnaryFunction for\_each(InputIterator first, InputIterator last, UnaryFunction f);

傳進去的是UnaryFunction型別，第一個type parameter T表示傳入的型別，第二個type parameter void，表示回傳的型別，最後重新定義operator()。  
  
**2.2 傳入參數**  
若要使class template也能傳入參數，一樣利用function object的技巧，借用constructor。

 1http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gif/\*   
 2http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif(C) OOMusou 2007 http://oomusou.cnblogs.com  
 3http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifFilename    : GenericAlgo\_for\_each\_ClassTemplateWithParameter.cpp  
 4http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifCompiler    : Visual C++ 8.0 / BCB 6.0 / gcc 3.4.2 / ISO C++  
 5http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifDescription : Demo how to use for\_each with class template & parameter  
 6http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifRelease     : 05/11/2007 1.0  
 7http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif\*/  
 8http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
 9http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <vector>  
10http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <iostream>  
11http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <algorithm>  
12http://www.cnblogs.com/Images/OutliningIndicators/None.gif#include <functional>  
13http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
14http://www.cnblogs.com/Images/OutliningIndicators/None.gifusing namespace std;  
15http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
16http://www.cnblogs.com/Images/OutliningIndicators/None.giftemplate<typename T, typename U>  
17http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifclass printElem : public unary\_function<T, void> {  
18http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifprivate:  
19http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  U \_prefix;  
20http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
21http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gifpublic:  
22http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  printElem(U prefix) : \_prefix(prefix) {}  
23http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
24http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  void operator() (T elem) {  
25http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    cout << \_prefix << elem << endl;  
26http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif  }  
27http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};  
28http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
29http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockStart.gifint main() {  
30http://www.cnblogs.com/Images/OutliningIndicators/ExpandedSubBlockStart.gif  int ia[] = {1, 2, 3};  
31http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  vector<int> ivec(ia, ia + sizeof(ia) / sizeof(int));  
32http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif    
33http://www.cnblogs.com/Images/OutliningIndicators/InBlock.gif  for\_each(ivec.begin(), ivec.end(), printElem<int, const char\*>("Element:"));  
34http://www.cnblogs.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}

**執行結果**

http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifElement:3

**Conclusion**  
STL的for\_each()事實上很好用，不過由於限制很多，所以常令很多新手卻步，本文試著將所有會遇到問題的地方都提出來討論，包括procedure based、object oriented、generics三種paradigm與for\_each()的搭配都涵蓋了，希望對各位有幫助。

分类: [[1] C/C++语言基础](http://www.cnblogs.com/Purple_Xiapei/category/378741.html)