

```
1 #include <iostream>
2 #include <string>
3 #include <list>
4
5 using namespace std;
6
7 template <typename T>
8 list<T> invertir_orden(const list<T> &l)
9 {
10     list<T> ret;
11     typename list<T>::const_reverse_iterator p;
12     for (p = l.rbegin(); p != l.rend(); p++)
13     {
14         ret.push_back(*p);
15     }
16     return ret;
17 }
18
19 template <typename T>
20 void mostrar_lista(const list<T> &l)
21 {
22     typename list<T>::const_iterator p;
23     for (p = l.begin(); p != l.end(); p++)
24     {
25         cout << *p << " ";
26     }
27     cout << endl;
28 }
29
30 int main()
31 {
32     list<int> aux;
33     list<int> aux2;
34     aux.push_back(2);
35     aux.push_back(1);
36     aux.push_back(1);
37     aux.push_back(1);
38     aux.push_back(1);
39     aux.push_back(5);
40     aux.push_back(3);
41     mostrar_lista(aux);
42     aux2=invertir_orden(aux);
43     mostrar_lista(aux2);
44 }
```