```cpp
#include <iostream>
#include <string>
#include <list>

using namespace std;

template <typename T>
void invertir_orden(list<T> &l)
{
    T aux;
    typename list<T>::iterator final=l.end();
    final--;
    typename list<T>::iterator principio=l.begin();
    while (final!=principio)
    {
        aux=*principio;
        *principio=*final;
        *final=aux;
        principio++;
        if(principio!=final)
        {
            final--;
        }
    }

}

template <typename T>
void mostrar_lista(const list<T> &l)
{
    typename list<T>::const_iterator p;
    for (p = l.begin(); p != l.end(); p++)
    {
        cout << *p << " ";
    }
    cout << endl;
}

int main()
{
    list<int> aux;
    aux.push_back(2);
    aux.push_back(10);
    aux.push_back(14);
    aux.push_back(13);
    aux.push_back(1);
    aux.push_back(5);
    aux.push_back(3);
    aux.push_back(80);
    mostrar_lista(aux);
    invertir_orden(aux);
    mostrar_lista(aux);
}
```