```
/* The following code example is taken from the book
 * "The C++ Standard Library - A Tutorial and Reference"
 * by Nicolai M. Josuttis, Addison-Wesley, 1999
 *
 * (C) Copyright Nicolai M. Josuttis 1999.
 * Permission to copy, use, modify, sell and distribute this software
 * is granted provided this copyright notice appears in all copies.
 * This software is provided "as is" without express or implied
 * warranty, and with no claim as to its suitability for any purpose.
 */
#include <iostream>
#include <memory>
using namespace std;

/* define output operator for auto_ptr
 * - print object value or NULL
 */
template <class T>
ostream& operator<< (ostream& strm, const auto_ptr<T>& p)
{
    // does p own an object ?
    if (p.get() == NULL) {
        strm << "NULL";         // NO: print NULL
    }
    else {
        strm << *p;             // YES: print the object
    }
    return strm;
}

int main()
{
    const auto_ptr<int> p(new int(42));
    const auto_ptr<int> q(new int(0));
    const auto_ptr<int> r;

    cout << "after initialization:" << endl;
    cout << " p: " << p << endl;
    cout << " q: " << q << endl;
    cout << " r: " << r << endl;

    *q = *p;
//  *r = *p;    // ERROR: undefined behavior
    *p = -77;
    cout << "after assigning values:" << endl;
    cout << " p: " << p << endl;
    cout << " q: " << q << endl;
    cout << " r: " << r << endl;

//  q = p;      // ERROR at compile time
//  r = p;      // ERROR at compile time
}
```