

```

//This program shows how the mystery function from test 1 in computer architecture worked ...
//David Underhill - 18FEB05

#include <iostream>
using namespace std;

//The mystery function from test 1!
//Pre: *a and *b are arrays; some element in array *a is 0
//Post: each element of `b` is assigned the value of the same element in `a` (up through the first 0 in array `a`)
void mystery(int *a, int *b)
{
    while( a[0] != 0 ) //loop while first element in a isn't 0
    {
        b[0] = a[0]; //assign current first `a` element to `b` ...
        a++; //increment operator on a pointer takes us to the
        b++; // start memory location of the next element
    } //Note: a++ and b++ are equivalent to: a = &a[1]; b = &b[1];
}

int main()
{
    int a[5] = {7, 4, 3, 0, 2}; //allocate some arrays of ints
    int b[5] = {-1, -2, -3, -4, -5};

    cout << "pre-mystery array values:\n"; //print out the arrays contents
    for(int i=0; i<5; i++)
        cout << "a[" << i << "]=" << a[i]
            << " --- "
            << "b[" << i << "]=" << b[i] << endl;

    mystery(a, b); //pass the arrays to the mystery function

    cout << "post-mystery array values:\n"; //print out the arrays new contents
    for(int i=0; i<5; i++)
        cout << "a[" << i << "]=" << a[i]
            << " --- "
            << "b[" << i << "]=" << b[i] << endl;

    return 0;
}

```