

Good Integers distinct in C++

```
#include <iostream>
#include <vector>
#include <algorithm>

using namespace std;

int GoodIntegers(vector<int>& arr) {
    sort(arr.begin(), arr.end()); // Sort the array

    int ans = 0;

    for (int i = 0; i < arr.size(); ++i) {
        if (arr[i] == i) { // Check if the value at index i
            matches i
            ++ans;
        }
    }

    return ans; // Return the count of good integers
}

int main() {
    vector<int> arr = {0, 1, 5, 7, 8, 9, 4};

    cout << GoodIntegers(arr) << endl;

    return 0;
}
```

Input:

vector<int> arr = {0, 1, 5, 7, 8, 9, 4};

Step 1: Sort the array

Sorted arr = {0, 1, 4, 5, 7, 8, 9}

↑ ↑ ↑ ↑ ↑ ↑
 Index 0 1 2 3 4 5 6

Step 2: Compare each element with its index

Index i	arr[i]	arr[i] == i	Count (ans)
0	0	✓ Yes	1
1	1	✓ Yes	2
2	4	✗ No	2
3	5	✗ No	2
4	7	✗ No	2
5	8	✗ No	2
6	9	✗ No	2

Final Output:

cout << GoodIntegers(arr); // Output: 2

✓ Because arr[0] = 0 and arr[1] = 1 match their indices.