

[x Dismiss](#)

Join the Stack Overflow Community

Stack Overflow is a community of 7.0 million programmers, just like you, helping each other.
Join them; it only takes a minute:

[Sign up](#)

c++ vector bubble sort



Love remote work?
Find it on a new kind of career site



Get started

I use `g++ -std=c++11 Sort.cpp` to compile my file.

My problem is the bubble sort don't sort.

Maybe I'm passing the vector by value but I don't know is closely my first time trying work with c++ and I chose use vector library.

My code is:

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

```

using namespace std;

void bubbleSort(vector<int> a);

void printVector(vector<int> a);

int main(int argc, char const *argv[])
{
    vector<int> a{3,2,6,1};
    printVector(a);

    bubbleSort(a);
    printVector(a);
}

void bubbleSort(vector<int> a)
{
    bool swapp = true;
    while(swapp)
    {
        swapp = false;
        for (int i = 0; i < a.size()-1; i++)
        {
            if (a[i]>a[i+1] )
            {
                a[i] += a[i+1];
                a[i+1] = a[i] - a[i+1];
                a[i] -=a[i+1];
                swapp = true;
            }
        }
    }
}

void printVector(vector<int> a)
{
    for (int i=0; i <a.size(); i++)
    {
        cout<<a[i]<<" ";
    }
    cout<<endl;
}

```

In the main I declare a vector type of int's and make the list {3,2,6,1}

After that e call the function `printVector` wich pretends print all numbers of vector on console and call `bubbleSort` function and finally print again.

[c++](#) [c++11](#) [vector](#) [bubble-sort](#)

edited Jun 21 '15 at 10:33

[p.i.g.](#)

1,749 1 9 26

asked Jun 21 '15 at 9:49

[warwcat](#)

115 1 11

3 Pass the vector by reference: `void bubbleSort(vector<int>& a)` – [πάντα ρεῖ](#) Jun 21 '15 at 9:50

1 Aside: clever swapping tricks are more likely to hurt than help these days. – [Hurkyl](#) Jun 21 '15 at 10:38

3 Answers

You need to pass by reference; by making a copy, you sort a temporary copy, and then that's it; it disappears and the original is still unsorted, because, once again, **you sorted only a temporary copy of the entire vector**.

So change `vector<int> a` to `vector<int>& a`.

Here's the code, fixed:

<http://coliru.stacked-crooked.com/a/2f118555f585ccd5>

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

using namespace std;

void bubbleSort(vector<int>& a);

void printVector(vector<int> a);

int main(int argc, char const *argv[])
{
    vector<int> a {3,2,6,1};

    printVector(a);

    bubbleSort(a);

    printVector(a);
}
```

```
}

void bubbleSort(vector<int>& a)
{
    bool swapp = true;
    while(swapp){
        swapp = false;
        for (size_t i = 0; i < a.size()-1; i++) {
            if (a[i]>a[i+1] ){
                a[i] += a[i+1];
                a[i+1] = a[i] - a[i+1];
                a[i] -=a[i+1];
                swapp = true;
            }
        }
    }
}

void printVector(vector<int> a){
    for (size_t i=0; i <a.size(); i++) {
        cout<<a[i]<<" ";
    }
    cout<<endl;
}
```

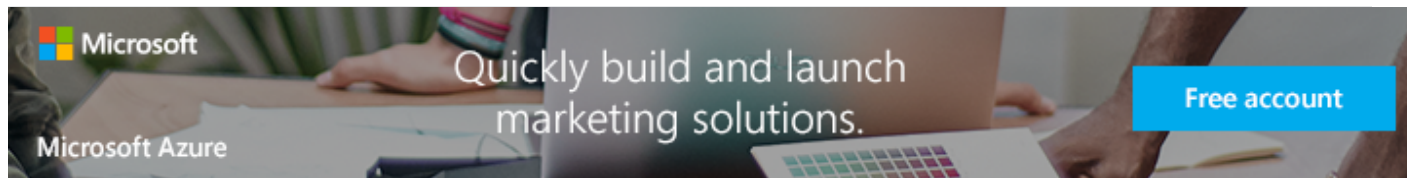
answered Jun 21 '15 at 9:53



VermillionAzure

2,928 7 37

@warwcat You should accept an answer with a check mark. – VermillionAzure Jun 21 '15 at 10:05



You are passing the vectors as values to your functions, meaning that you are sorting a copy,

not the original vector, and then printing a copy of the original vector.

Change the parameter to `vector<int> &a` in the `bubbleSort` function, and to `vector<int> const &a` in the `printVector` function (as you don't need to change the vector content from here).

By the way, your code may be affected by undefined behavior caused by signed integer overflow. You should use another method to swap your elements, `std::swap` for example:

```
std::swap(a[i], a[i + 1]);
```

answered Jun 21 '15 at 9:52



rems4e

2,441 1 9 22

thanks I don't have known this swap function thanks very much – [warwcat](#) Jun 21 '15 at 10:06

I was stuck on it as well for sometime. This question has already been answered fine by VermillionAzure, but still pasting my solution. I ain't using the `std::swap` just because I like to do more by hand.

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

//function to swap values
//need to pass by reference to sort the original values and not just these copies
void Swap (int *a, int *b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}

void BubbleSort (std::vector<int> &array)
{
    std::cout<<"Elements in the array: "<<array.size()<<std::endl;

    //comparisons will be done n times
    for (int i = 0; i < array.size(); i++)
    {
```

```
//compare elemet to the next element, and swap if condition is true
for(int j = 0; j < array.size() - 1; j++)
{
    if (array[j] > array[j+1])
        Swap(&array[j], &array[j+1]);
}
}

//function to print the array
void PrintArray (std::vector<int> array)
{
    for (int i = 0; i < array.size(); i++)
        std::cout<<array[i]<<" ";
    std::cout<<std::endl;
}

int main()
{
    std::cout<<"Enter array to be sorted (-1 to end)\n";

    std::vector<int> array;
    int num = 0;
    while (num != -1)
    {
        std::cin>>num;
        if (num != -1)
            //add elements to the vector container
            array.push_back(num);
    }

    //sort the array
    BubbleSort(array);

    std::cout<<"Sorted array is as\n";
    PrintArray(array);

    return 0;
}
```

answered Jan 14 '16 at 0:35



Testing123

47 8

