## Best way to compare elements in a vector and return a object

Asked 15 years, 10 months ago Modified 11 years, 8 months ago Viewed 2k times



I have a use case wherein numbers are monotonically increasing in an vector of integers

1



П



1

vec[0] = 2
vec[1] = 5
vec[2] = 8
vec[3] = 10

If I am passed number 6, I want to return vec[1], since it lies between vec[1] and vec[2], similarly if passes 9 would have to return vec[2]. My experience with STL is limited, so wanted to check can we solve this with STL or you have to iterate over each by storing the previous and when hit a number greater than the passed number you return

c++ stl

Share Improve this question Follow

asked Jan 30, 2009 at 16:23

kal

29.3k • 49 • 132 • 149

## 3 Answers

Sorted by:

Highest score (default)





The STL has four reusable binary search algorithms in the <algorithm> header: <algorithm> lower bound, upper bound, equal range, and binary search.

**10** 



lower\_bound doesn't do exactly what you want: when the desired element is not present in the sequence, it returns an iterator that refers to the element one past the element that you want. However, you should be able to wrap it with code that implements your behavior without much trouble.



Share Improve this answer Follow







You can use binary search.

2

Share Improve this answer Follow

answered Jan 30, 2009 at 16:26



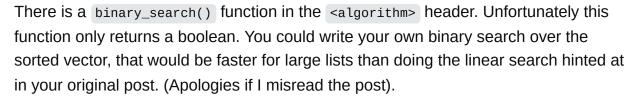
**4,773** • 7 • 35 • 43



1



0





Share Improve this answer Follow







This is a great example of why good naming is important. Many people who are looking for a binary search will not find lower\_bound, upper\_bound, or equal\_range unless they already know that those are binary searches too. – bk1e Jan 30, 2009 at 16:45