Binary Search in C++ Standard Template Library (STL)

Binary search is a widely used searching algorithm that requires the array to be sorted before search is applied. The main idea behind this algorithm is to keep dividing the array in half (divide and conquer) until the element is found, or all the elements are exhausted.

It works by comparing the middle item of the array with our target, if it matches, it returns true otherwise if the middle term is greater than the target, the search is performed in the left sub-array.
If the middle term is less than target, the search is performed in the right sub-array.

The prototype for binary search is :

```
binary_search(startaddress, endaddress, valuetofind)

startaddress: the address of the first element of the array.
endaddress: the address of the last element of the array.
valuetofind: the target value which we have to search for.
```

```cpp
// CPP program to implement
// Binary Search in
// Standard Template Library (STL)
#include <algorithm>
#include <iostream>

using namespace std;

void show(int a[], int arraysize)
{
    for (int i = 0; i < arraysize; ++i)
        cout << a[i] << " ";
}

int main()
{
    int a[] = { 1, 5, 8, 9, 6, 7, 3, 4, 2, 0 };
    int asize = sizeof(a) / sizeof(a[0]);
    cout << "\n The array is : ";
    show(a, asize);

    cout << "\n\nLet's say we want to search for 2 in the array";
    cout << "\n So, we first sort the array";
    sort(a, a + asize);
    cout << "\n\n The array after sorting is : ";
    show(a, asize);
    cout << "\n\nNow, we do the binary search";
    if (binary_search(a, a + 10, 2))
        cout << "\nElement found in the array";
    else
        cout << "\nElement not found in the array";

    cout << "\n\nNow, say we want to search for 10";
    if (binary_search(a, a + 10, 10))
        cout << "\nElement found in the array";
    else
        cout << "\nElement not found in the array";

    return 0;
}
```

The output of the above program is :

```
The array is : 1 5 8 9 0 6 7 3 4 2 0

Let's say we want to search for 2 in the array
 So, we first sort the array
```

```
The array after sorting is : 0 1 2 3 4 5 6 7 8 9

Now, we do the binary search
Element found in the array

Now, say we want to search for 10
Element not found in the array
```

Related Article: std::bsearch in C++

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

Recommended Posts:

Set in C++ Standard Template Library (STL)

The C++ Standard Template Library (STL)

Map in C++ Standard Template Library (STL)

Multimap in C++ Standard Template Library (STL)

Deque in C++ Standard Template Library (STL)

Queue in Standard Template Library (STL)

Multiset in C++ Standard Template Library (STL)

Sort in C++ Standard Template Library (STL)

List in C++ Standard Template Library (STL)

Pair in C++ Standard Template Library (STL)

Unordered Sets in C++ Standard Template Library

Priority Queue in C++ Standard Template Library (STL)

Meta Binary Search | One-Sided Binary Search

Linear Search vs Binary Search

Interpolation search vs Binary search

Improved By : ChandanaAkriti

Article Tags : C++ Binary Search cpp-algorithm-library STL

Practice Tags : STL Binary Search CPP

14

1.4

To-do Done

Based on **61** vote(s)

Feedback/ Suggest Improvement Add Notes Improve Article

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments