

简单的程序诠释C++ STL算法系列之十：search

C++STL的非变异算法（Non-mutating algorithms）是一组不破坏操作数据的模板函数，用来对序列数据进行逐个处理、元素查找、子序列搜索、统计和匹配。

search算法函数在一个序列中搜索与另一序列匹配的子序列。它有如下两个原型，在迭代器区间[first1, last1)上找迭代器区间[first2, last2)完全匹配（或者满足二元谓词binary_pred）子序列，返回子序列的首个元素在[first1, last1)区间的迭代器值，或返回last1表示没有匹配的子序列。

函数原型：

```
template<class ForwardIterator1, class ForwardIterator2>
ForwardIterator1 search(
    ForwardIterator1 _First1,
    ForwardIterator1 _Last1,
    ForwardIterator2 _First2,
    ForwardIterator2 _Last2
);
template<class ForwardIterator1, class ForwardIterator2, class Pr>
ForwardIterator1 search(
    ForwardIterator1 _First1,
    ForwardIterator1 _Last1,
    ForwardIterator2 _First2,
    ForwardIterator2 _Last2,
    BinaryPredicate _Comp
);
```

示例程序：

在vector向量容器v1={5, 8, 1, 4}中搜索是否包含子序列容器向量v2={8, 1}，打印搜索结果“v2的元素包含在v1中，起始元素为：v1[1]”

```
/*
 * Copyright (C) Jerry Jiang
 * File Name : search.cpp
 * Author : Jerry Jiang
 * Create Time : 2011-10-10 23:22:34
 * Mail : jbiaojerry@gmail.com
 * Blog : http://blog.csdn.net/jerryjbiao
 * Description : 简单的程序诠释C++ STL算法系列之十
 * 非变异算法 : 子序列搜索search
 */

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

using namespace std;

int main()
{
    vector<int> v1;
    v1.push_back(5);
    v1.push_back(8);
    v1.push_back(1);
    v1.push_back(4);

    vector<int> v2;
    v2.push_back(8);
    v2.push_back(1);

    vector<int>::iterator iterLocation;
    iterLocation = search(v1.begin(), v1.end(), v2.begin(), v2.end());

    if (iterLocation != v1.end())
    {
        cout << "v2的元素包含在v1容器中，起始元素为"
              << "v1[" << iterLocation - v1.begin() << "]" << endl;
    }
    else
    {
        cout << "v2的元素不包含在v1容器" << endl;
    }
    return 0;
}
```

C++经典书目索引及资源下载：<http://blog.csdn.net/jerryjbiao/article/details/7358796>

阅读更多 登录后自动展开