

简单的程序诠释C++ STL算法系列之三：find_if

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。

find_if算法 是find的一个谓词判断版本，它利用返回布尔值的谓词判断pred，检查迭代器区间[first, last)上的每一个元素，如果迭代器iter满足pred(*iter) == true，表示找到元素并返回迭代器值iter；未找到元素，则返回last。

函数原型：

```
template<class InputIterator, class Predicate>
InputIterator find_if(
    InputIterator _First,
    InputIterator _Last,
    Predicate _Pred
);
```

示例代码：

```
/******
 * Copyright (C) Jerry Jiang
 * File Name   : find_if.cpp
 * Author      : Jerry Jiang
 * Create Time : 2011-9-29 22:21:29
 * Mail        : jbiaojerry@gmail.com
 * Blog        : http://blog.csdn.net/jerryjbiao
 * Description : 简单的程序诠释C++ STL算法系列之三
 *              非变异算法 : 条件查找容器元素find_if
 *****/

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

using namespace std;

//谓词判断函数 divbyfive : 判断x是否能5整除
bool divbyfive(int x)
{
    return x % 5 ? 0 : 1;
}

int main()
{
    //初始vector
    vector<int> iVect(20);
    for(size_t i = 0; i < iVect.size(); ++i)
    {
        iVect[i] = (i+1) * (i+3);
    }

    vector<int>::iterator ilocation;
    ilocation = find_if(iVect.begin(), iVect.end(), divbyfive);

    if (ilocation != iVect.end())
    {
        cout << "第一个能被5整除的元素为: "
              << *ilocation << endl //打印元素: 15
              << "元素的索引位置为: "
              << ilocation - iVect.begin() << endl; //打印索引位置: 2
    }

    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