c++ map与unordered_map区别及使用_c++ map和unordered_map_别说话写代码的博客-CSDN博客

成就一亿技术人!

原创

于 2018-12-01 13:06:34 首次发布

转自：map和unordered_map的差别和使用_陈云佳的专栏-CSDN博客_map和unordered

需要引入的头文件不同
map: #include < map >
unordered_map: #include < unordered_map >

内部实现机理不同
map： map内部实现了一个红黑树（红黑树是非严格平衡二叉搜索树，而AVL是严格平衡二叉搜索树），红黑树具有自动排序的功能，因此map内部的所有元素都是有序的，红黑树的每一个节点都代表着map的一个元素。因此，对于map进行的查找，删除，添加等一系列的操作都相当于是对红黑树进行的操作。map中的元素是按照二叉搜索树（又名二叉查找树、二叉排序树，特点就是左子树上所有节点的键值都小于根节点的键值，右子树所有节点的键值都大于根节点的键值）存储的，使用中序遍历可将键值按照从小到大遍历出来。
unordered_map: unordered_map内部实现了一个哈希表（也叫散列表，通过把关键码值映射到Hash表中一个位置来访问记录，查找的时间复杂度可达到O(1)，其在海量数据处理中有着广泛应用）。因此，其元素的排列顺序是无序的。哈希表详细介绍

优缺点以及适用处
map：

优点：

有序性，这是map结构最大的优点，其元素的有序性在很多应用中都会简化很多的操作
红黑树，内部实现一个红黑书使得map的很多操作在lgn的时间复杂度下就可以实现，因此效率非常的高
缺点： 空间占用率高，因为map内部实现了红黑树，虽然提高了运行效率，但是因为每一个节点都需要额外保存父节点、孩子节点和红/黑性质，使得每一个节点都占用大量的空间

适用处：对于那些有顺序要求的问题，用map会更高效一些

unordered_map：

优点： 因为内部实现了哈希表，因此其查找速度非常的快
缺点： 哈希表的建立比较耗费时间
适用处：对于查找问题，unordered_map会更加高效一些，因此遇到查找问题，常会考虑一下用unordered_map
总结：

内存占有率的问题就转化成红黑树 VS hash表，还是unorder_map占用的内存要高。
但是unordered_map执行效率要比map高很多
对于unordered_map或unordered_set容器，其遍历顺序与创建该容器时输入的顺序不一定相同，因为遍历是按照哈希表从前往后依次遍历的
map和unordered_map的使用
unordered_map的用法和map是一样的，提供了 insert，size，count等操作，并且里面的元素也是以pair类型来存贮的。其底层实现是完全不同的，上方已经解释了，但是就外部使用来说却是一致的。

C++ Map常见用法说明

常用操作汇总举例：

```
#include <iostream>
#include <unordered_map>
#include <map>
#include <string>
using namespace std;
int main()
{

    unordered_map<int, string> myMap={{ 5, "张大" },{ 6, "李五" }};
    myMap[2] = "李四";
    myMap.insert(pair<int, string>(3, "陈二"));


auto iter = myMap.begin();
while (iter!= myMap.end())
    {
        cout << iter->first << "," << iter->second << endl;
        ++iter;
    }


auto iterator = myMap.find(2);
if (iterator != myMap.end())
           cout << endl<< iterator->first << "," << iterator->second << endl;
```

```
system("pause");

return 0;

}
```

此时用的是unordered_map，输出的结果为：

```
3,陈二
2,李四
6,李五
5,张大

2,李四
```

若把unordered_map换成map，输出的结果为：

```
2,李四
3,陈二
5,张大
6,李五

2,李四
```

set和unordered_set的使用方法类似于map和unordered_map，详情请见：

【总结】unordered_map,unordered_set,map和set的用法和区别

参考

【1】c++中map与unordered_map的区别

c++中map与unordered_map的区别_wolfrevoda的专栏-CSDN博客_map和unorderedmap的区别

【2】C++Map常见用法说明

http://blog.csdn.net/shuzfan/article/details/53115922#二-插入操作