

CCF考试资料

Dev环境配置：

考试环境：

C/C++： Dev-CPP 5.4.0 (Min GW 4.7.2)

Dev选择C++11标准：

可以点击菜单栏的“工具”->“编译选项”进入如下界面，勾选“编译时加入以下指令”，填入"-std=c++11"

头文件：

- 建议打印常用头文件的列表

```
#include<bits/stdc++.h>
using namespace std; //万能头文件

#include <iostream>
#include <cstdio>
#include <fstream>
#include <algorithm>
#include <cmath>
#include <deque>
#include <vector>
#include <queue>
#include <string>
#include <cstring>
#include <map>
#include <stack>
#include <set>
```

优先队列的自定义排序（C++）：

```

struct cmp
{
    bool operator()(const pair<int, int>& a, const pair<int, int>& b)
    {
        //优先队列的排序与普通排序相反
        //此为从小到大排序
        return a.first > b.first;
    }
};

/*将capital和profits绑定，按照capital从小到大排序，用堆排序*/
priority_queue<pair<int, int>, vector<pair<int, int> >, cmp>
projects;

```

Sort函数：

```

/*
头文件：#include <algorithm>
第三个参数compare，是个自定义的比较函数的指针，名字可以随便命名；原型如下：
    bool cmp (const Type1 &a, const Type2 & b );
简单说来就是：
1.比较a和b，如果是想升序，那么就定义当a<b的时候返回true；
2.如果是想降序，那么就定义当a>b的时候返回true；
*/
#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
using namespace std;

bool cmp(const int& a, const int& b)
{
    return a > b; //从大到小排序
}

int main()
{
    int a[10] = {2, 3, 30, 305, 32, 334, 40, 47, 5, 1};
    vector<int> nums(a, a + 10);

    sort(nums.begin(), nums.end(), cmp);

    for(auto x : nums)
        cout << x;
    cout << endl;

    system("pause");
    return 0;
}

```