



程序设计与算法

OpenJudge 在线做题指南

郭 炜

微信公众号

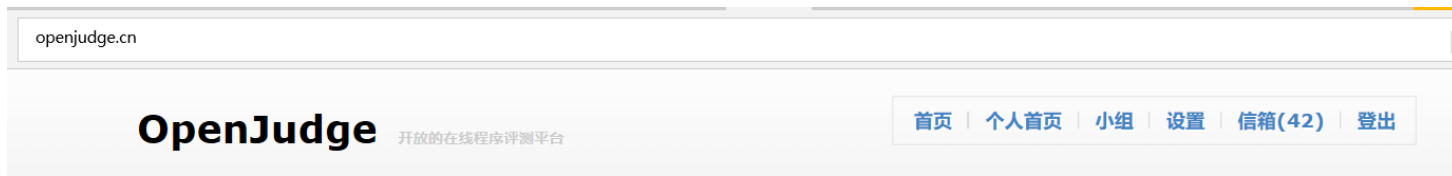


微博: <http://weibo.com/guoweiofpku>

学会程序和算法，走遍天下都不怕!

http://openjudge.cn

第一步：注册账号



你还没有头像，[设置头像](#)后，别人会更容易记住你。

正在进行的比赛(110)... (查看全部)

	程序设计基础每日一练20171105	1 题目	31 参与者	38 提交数
	开始时间: 2017-11-05 06:00:00 结束时间: 2017-11-06 10:00:00			
	测控2016级练习03	3 题目	111 参与者	651 提交数
	开始时间: 2017-11-02 11:00:00 结束时间: 2017-11-05 23:00:00			
	2017算法课第二次作业			

活跃的小组... (查看全部)



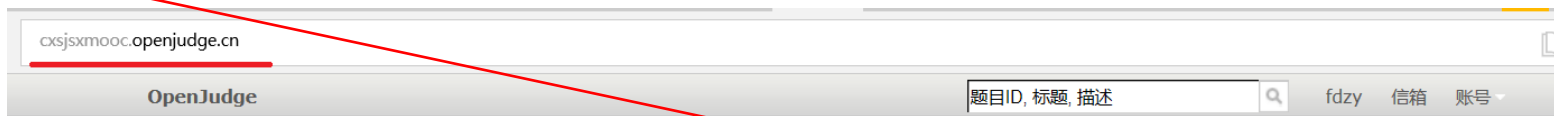
OpenJudge是开放的在线程序评测系统
您可以创建自己的OJ小组 [申请创建](#)

公告

- 10-17 交流QQ群号为609300410
- 10-11 请16级同学将昵称修改为“16-1-学号-姓名”格式，并选择自己的班级
- 09-16 [第二次作业](#)
- 09-12 新学期清理成员，凡不是实名制的都将被清出团队
- 09-11 请登录潍坊五中信息技术
- 08-19 请各位同学加入小组时务必实名，我们将定期清查非本小组成员，并将不符合要求的同学移出本组，谢谢配合！
- 08-04 初中红/绿校服OI暑假作业
- 08-04 蓝校服衔接班OI集训作业2

做题网址: <http://cxjsxmooc.openjudge.cn>

第二步: 加入程序设计实习MOOC组



管理员

林舒 someone frank Jia(刘家瑛)
Janelle

公告

成员(12932)

[查看全部](#)



程序设计实习MOOC

» 程序设计与算法 (一) 第一周测验(2017冬季)(6题) 结束时间: 2017-11-19 23:00:00

» 程序设计与算法 (三) 测验题汇总(2017秋季)(22题) 结束时间: 2017-12-22 23:00:00

» 程序设计与算法 (三) 第五周测验(2017秋季)(3题) 结束时间: 2017-11-05 23:00:00

» 程序设计与算法 (三) 第六周测验(2017秋季)(4题) 结束时间: 2017-11-12 23:00:00

» 程序设计与算法 (三) 第七周测验(2017秋季)(7题) 结束时间: 2017-11-12 23:00:00

» 程序设计与算法 (三) 第八周测验(2017秋季)(7题) 结束时间: 2017-11-12 23:00:00

» 程序设计与算法 (三) 第九周测验(2017秋季)(4题) 结束时间: 2017-11-19 23:00:00

» 程序设计与算法 (三) 魔兽世界大作业(2017秋季)(1题) 结束时间: 2017-11-27 23:00:00

» 计划中的比赛

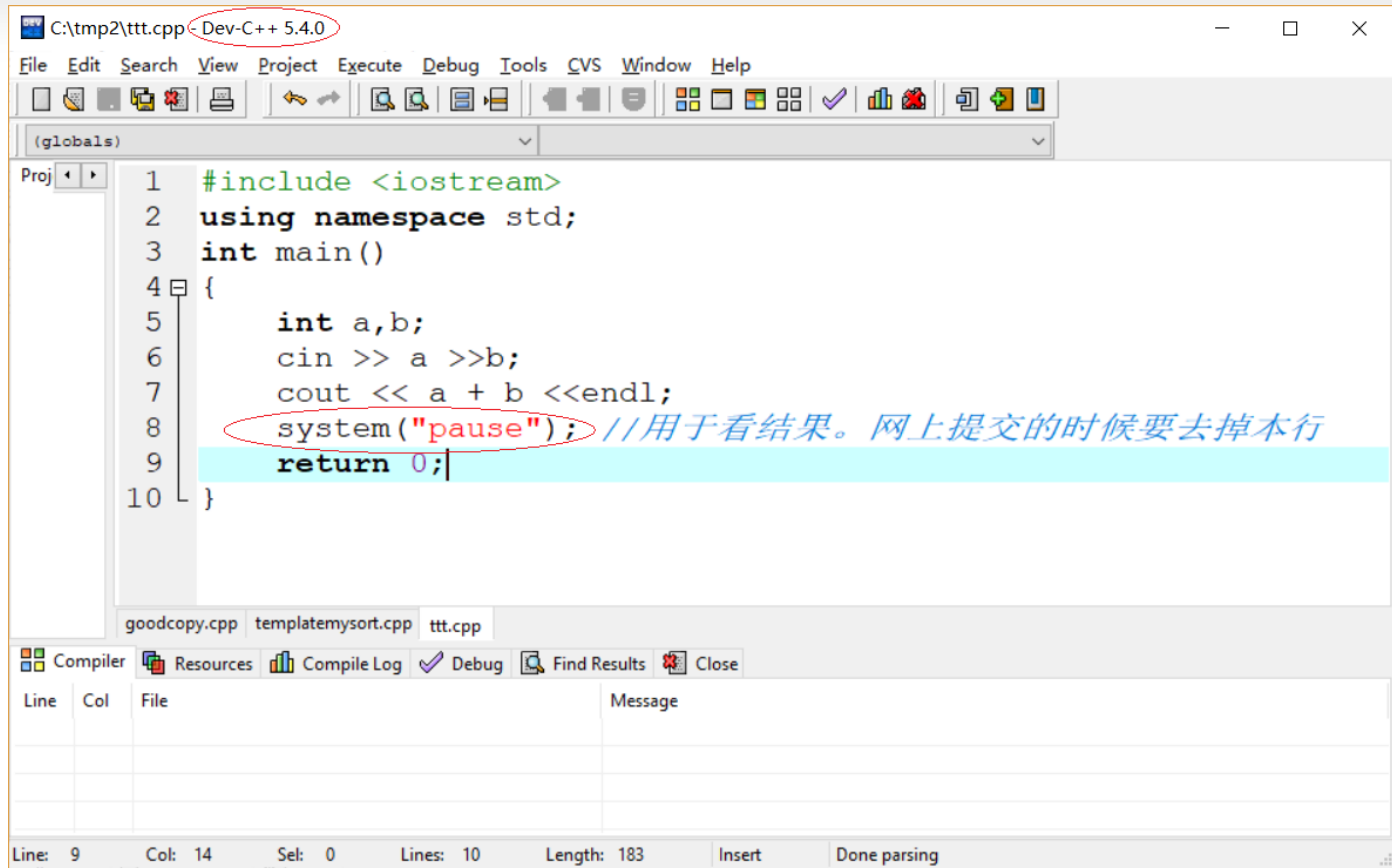
编程环境

编程环境建议用：
Dev C++

程序编译好运行的时候，运行结果会在一个黑窗口里面一闪而过，为了看到结果，需要在
return 0;
前面写：

`system("pause");`

提交的时候记得去掉
这一行



提交程序

样例输入

```
7
add 1
add 1
ask 1
ask 2
del 2
del 1
ask 1
```

样例输出

```
1
2
1 2
0 0
0
2
1 0
```

提示

Please use STL's set and multiset to finish the task

查看

提交

统计

提问

提交程序

Submit Your Solution for Problem 1: Set

编程语言

☒ G++(5.3)

☐ GCC(5.3)

源代码:

提交的时候，编程语言选 “G++”

```
#include <queue>
#include <stack>
#include <iostream>
using namespace std;
queue<int > q;
int main()
{
    priority_queue<int> pq;
    pq.push(3);pq.push(4);
    pq.push(5);
    int & r = (int & )pq.top();
    r = 1;
    cout <<pq.top() << endl;

    return 0;
}
```

提交

重置

提交以后可能的结果

Accepted

正确，通过

Wrong Answer:

程序不正确，导致输出的答案错误。

Time Limit Exceed:

超时。程序运行时间超过了允许的时间。
原因：有死循环，或算法不好，导致运行太慢

Runtime Error:

运行时错误。通常由于数组越界，或者不合理使用指针

Presentation Error:

几乎对了。只是多了或少了空格，多了或少了换行

Output Limit Exceed:

死循环导致不停输出

提交正确

提交正确以后下载通过码文件,用记事本或任何文字编辑软件打开后拷贝通过码,到中国大学MOOC进行填空



题目

排名

状态

统计

提问

#11287348提交状态

状态: Accepted

下载通过码文件

源代码

```
//By Guo Wei
#include <iostream>
#include <cstdio>
#include <set>
#include <string>
#include <cstring>

using namespace std;
typedef multiset<int>::iterator PTR;

int main()
{
    int n;
    multiset<int> mst;
    set<int> st; //加入过的都放到这里
    PTR p;
```

基本

提

提交出错

#11287347提交状态

状态: Wrong Answer

下载错误测试点输入

下载错误测试点正确输出

➤ 提交出错以后，下载错误测试点输入和输出在本机进行调试

➤ 请注意：下载的数据是Linux格式的，在Windows的记事本中打开会看不到换行。可以拷贝到Dev c++或Word，写字板等其他编辑软件中查看

源代码

```
#include <queue>
#include <stack>
#include <iostream>
using namespace std;
queue<int > q;
int main()
{
    priority_queue<int> pq;
    pq.push(3);pq.push(4);
    pq.push(5);
    int & r = (int & )pq.top();
    r = 1;
    cout <<pq.top() << endl;

    return 0;
}
```

输入和输出

- 严格按照题目描述的要求进行输出，不输出任何题目没有要求输出的内容，比如“请输入两个数：”之类，也不要输出多余的标点符号、空格和换行。也不要少输出任何内容。
- 对题目要求输出的一些固定文字，建议从题目中拷贝，以免敲错单词，敲错大小写

输入和输出

- 对题目中描述了范围的变量，不用在程序中检查其范围。例如，题目说：输入的第一行是整数 n ($0 \leq n \leq 1000$)，则程序中不用检查输入的 n 是否在 $[0,1000]$ 之内。题目说 n 在此范围，那么数据中的 n 就是在此范围。

多组数据的输入输出例题：A+B Problem

输入数据：

第一行是整数 n ，表示有 n 组数据。接下来有 n 行，每行两个整数 a 和 b

输出数据：

对每组数据，输出 a 和 b 的和

输入样例：

3

2 4

1 2

输出样例：

6

3

对于多组数据的题目，不需要全部读入所有数据存起来后再一一处理，可以读入一组数据就处理一组数据并输出。只需确保输出部分符合题目对输出的要求即可，不需担心在本机测试时输入和输出会混合在一起，看起来不符合题目要求。

多组数据的输入输出例题：A+B Problem

解题程序：

```
#include <iostream>
using namespace std;
int main() {
    int n;
    cin >>n;
    for(int i = 0;i < n; ++i) {
        int a,b;
        cin >> a >> b;           //读入一组就处理、输出一组
        cout << a + b <<endl;
    }
    return 0;
}
```

最常见的错误: 局部变量没有初始化

程序在本机对所有下载的数据运行都正确，交上去还是不对，最可能的原因就是：

局部变量没有初始化!

局部变量应当初始化，比如在有的题目里面：

```
....  
int main()  
{  
    int sum; int a[5];  
    ....  
    return 0;  
}
```

这个sum是用来在后面求和用的，那就应该有个初始值，比如：

```
int sum = 0;
```

否则，它的初始值就是随机的，那么后面的求和结果自然不对。a数组没初始化，元素值也随机

最常见的错误: 局部变量没有初始化

但是经常有同学说: 在本机运行是对的, 交上去就不对了。那是你人品好, 那些没初始化的局部变量, 在你本机运行时, 碰巧它初始值就是0, 所以你的程序就对了。交上去就没这么走运。

变量一定要初始化!

变量一定要初始化!

变量一定要初始化!

重要的事情说三遍。