目录

[一、 读题](#_Toc26250_WPSOffice_Level1) [1](#_Toc26250_WPSOffice_Level1)

[二、 做题前注意](#_Toc10804_WPSOffice_Level1) [1](#_Toc10804_WPSOffice_Level1)

[三、 比赛前注意](#_Toc18932_WPSOffice_Level1) [1](#_Toc18932_WPSOffice_Level1)

[四、 字符串题](#_Toc15371_WPSOffice_Level1) [2](#_Toc15371_WPSOffice_Level1)

[五、 数据类型](#_Toc18870_WPSOffice_Level1) [3](#_Toc18870_WPSOffice_Level1)

[六、 排序](#_Toc11538_WPSOffice_Level1) [4](#_Toc11538_WPSOffice_Level1)

[七、 C++ STL](#_Toc30035_WPSOffice_Level1) [5](#_Toc30035_WPSOffice_Level1)

[八、 质数](#_Toc16200_WPSOffice_Level1) [6](#_Toc16200_WPSOffice_Level1)

[九、 坐标题](#_Toc19505_WPSOffice_Level1) [6](#_Toc19505_WPSOffice_Level1)

[十、 提高编写速度和质量](#_Toc11342_WPSOffice_Level1) [6](#_Toc11342_WPSOffice_Level1)

[十一、 函数](#_Toc25054_WPSOffice_Level1) [7](#_Toc25054_WPSOffice_Level1)

[十二、 段错误](#_Toc16422_WPSOffice_Level1) [7](#_Toc16422_WPSOffice_Level1)

[十三、 格式错误](#_Toc26201_WPSOffice_Level1) [7](#_Toc26201_WPSOffice_Level1)

[十四、 做题习惯](#_Toc14029_WPSOffice_Level1) [8](#_Toc14029_WPSOffice_Level1)

[十五、 比赛规则和做题方法](#_Toc10682_WPSOffice_Level1) [8](#_Toc10682_WPSOffice_Level1)

[十六、 比赛考点](#_Toc23492_WPSOffice_Level1) [10](#_Toc23492_WPSOffice_Level1)

[十七、 题目难度及变化](#_Toc15906_WPSOffice_Level1) [11](#_Toc15906_WPSOffice_Level1)

1. 读题

特别是对于难理解题意、输出要求复杂的题目，先认真题目，才开始做题。

对样例实现一遍：手算、画图。

遇到特殊题目，或完全不理解题意的题目，用一个特别暴力、简短的代码实现一下。

1. 做题前注意

求的是啥

数据范围

输入输出格式

是否多组数据（注意初始化）

特殊数据

1. 比赛前注意

对编译环境进行测试

要能实现代码运行、调试

比赛前打好基本模板（应该允许的，见muban.txt），保存成txt放在桌面，需要的时候拷贝之。

如果允许打其它模板，把BFS、并查集、最短路、搜索二叉树、字符串排序等的模板打一下，保存成txt放在桌面，注意命名。前提是真的可以，最好问清楚！

我忘记了去年是否有杀毒软件，如果有，我强烈建议关掉它们！

去年好像电脑有多个输入法，我的建议是去掉只剩一个。

Codeblocks和Dev c++ 字体的大小，最好调一下。

去年遇到某些浏览器不能使用的情况，注意老师的最新通知！

去年好像是不禁外网的（神坑），我用360浏览器，然后一不小心就拖动了网页内容，然后进行百度搜索了。。。

当然，利用网页搜索题目，想也知道，这是绝对不行的！

当然这要是不小心碰上了，请镇定自若的关闭之。

去年还遇到一个比较坑的事情，就是我提交代码时，发现网页框里有别人的代码！！！

遇到这样的事情，镇定自若，不要慌张，这样的事情其它同学大概率也会遇到！不要贪小便宜，遇到这样的事情，肯定有比较完善的补救措施，不要脑袋一热，做出错误的事情！

机子，键盘不行的话，请及时联系老师更换

运到编译器的问题，尽快联系老师解决

去年我的电脑就是有点小问题，碰一下屏幕，屏幕就没有内容，还有同学反应有些键盘比较难用，像这样的问题，尽管问老师，看能否换机，换键盘！

1. 字符串题

看清

字符串的 输入 输出 格式

--------------------------------------------------

Scanf 自动跳过’ ’,’\n’,tab,无法读入这些字符。

用fgets()。

以’\n’结束读入。不能读入某些’\n’。

scanf("%d\n",&a);

fgets(str,maxn,stdin);

天梯赛评测系统，不能使用gets(str)

天梯赛评测系统，一定要去除末尾'\n'。

len=strlen(str)-1;

str[len]=0;

针对格式设计不同的输入输出

scanf("%c %c",&x,&y); ///跳过的是多个连续的空格

scanf(“%d:%d”,&a,&b);

printf(“%05d”,a); ///用0补全不够的5位

--------------------------------------------------

熟悉字符串函数

strlen

strcpy

strcmp

strncpy (末尾加'\0')

--------------------------------------------------

熟悉转换，如atoi，atof

--------------------------------------------------

读入：字符串长度为n，结束符长度为1，需要的存储大小至少为n+1

char a[10000+1];

char a[10000+10];

--------------------------------------------------

char a[maxn],b[maxn];

swap(a,b);

是两个指针的变换。

1. 数据类型

注意( [ ) ]

int [-2^31,2^31)

long long [-2^63,2^63)

short

unsigned int

unsigned short

两数相乘也许大于int范围，

首先结果的数据类型为long long，

其次，两数的数据类型至少有一个为long long，

否则要这样写1ll\*a\*b

--------------------------------------------------

判断两个实数是否相等

相差eps的都可以视为相等

eps可以取1e-8,1e-6,...

e.g.判断

x>y if (x>y+eps)

x<y if (x<y+-eps)

x==y if (y-eps<x && x<y+eps)

--------------------------------------------------

除法

double / double

double / int

int / double

不要

int / int

--------------------------------------------------

读入：100(保存成实数类型)

scanf("%lf",&a);

--------------------------------------------------

保留k位小数

scanf(“%.2f”,a); //2位

--------------------------------------------------

可能出现-0.00，会输出-0.00

If (fabs(a)<eps)

Printf(“0.00”);

虽然出现可能性较低

1. 排序

bool cmp(int a,int b)

{

return a<b;

}

仅一个关键字，倒序输出

sort(a,a+n);

for (i=n-1;i>=0;i--)

a[1]-a[n]

Sort(a+1,a+n+1)

两个关键字排序

E.g

struct node

{

int a,b;

bool operator<(node y)

{

if (a==y.a)

return b<y.b;

return a<y.a;

}

}

但是如果是多个关键字（三个关键字）的排序，这样写可能条例更清晰，不容易错

//第一关键字的排序

if (a.first>b.first)

return ;

else if (a.first<b.first)

return ;

else

{

//第二关键字的排序

...

}

字符串排序

c qsort 字符串排序.txt

注意c和c++使用方法的不同，比赛前不看模板，自行打一遍

1. C++ STL

有必要学一下

#include <map>

是否出现；赋值

unordered\_map某些方面比map快，但天梯赛一般不会出现用map超时，用unordered\_map不超时的情况

map<x,y> ma;

x,y均为数据类型

ma.clear(); //清空

ma[x]=y; //赋值

if (ma.find(x)==ma.end()) //x不在ma中

xxx

printf("%d",ma[x]); //前提是x在ma中

--------------------------------------------------

#include <set>

Set<int> se;

去重；求个数

se.clear(); //清空

se.size(); //统计个数

se.insert() //添加

--------------------------------------------------

#include <stack>

Stack<int>st;

St.push(x); //加入

St.pop(); //删除

X=St.top(); //顶端

St.empty() //是否空

1. 质数

求1-n质数的方法

判断是否质数的方法

数字1不是质数

M=sqrt(n+eps); //自动int(xxx)

For (i=2;i<=m;i++)

1. 坐标题

做题前，先明确x,y,z坐标轴的定义，n,m的定义，读入方式。

g++:y1是保留变量，不能用！

改成clang++提交

1. 提高编写速度和质量

true->1 false->0

min(a,b) max(a,b) \_\_gcd(a,b)

main函数外的变量，初始值为0。

特别是数组，请尽量放在main函数外。

多个语句写在一行，有逗号隔开，这样可以省去加括号和换行的时间。但是break,continue,return等操作，不能写在一行。

Codeblocks/Dev c++ 快捷键操作

Ctrl+L 删掉一行

Ctrl+T 光标所在行与上一行发生交换

Ctrl+R 替换

利用maxn等变量修改数组范围，e.g. a[maxn]，这个因人而异。

多个类似的操作，函数替代，以此节省思考量 / 如果写错导致的调试时间，如坐标题。

判断对错，很多个标准

bool judge()

{

......

}

if (judge())

printf(“”);

else

printf(“”);

1. 函数

如果有返回值的最好在最后加上return xxx，即使这句话没有作用

int xxx()

{

if ()

return xxx;

......

return yyy;

}

1. 段错误

数据范围写小了

特殊样例

代码写错了(下标)

1. 格式错误

空格改成'\_'进行测试

造几个有针对性的数据

不要理所当然，回去再看题意

一道题做错，创建新的工程做下一题

初始代码的maxn写大一点没关系，一般都不会内存超限

1. 做题习惯

模板保存成txt放在桌面并打开，当做对一道题，拷贝之。（因人而异）

一道题做错，创建新的工程，做下一题。

初始代码的maxn写大一点没关系（如1e5），一般都不会内存超限。

1. 比赛规则和做题方法

仅供参考：

1.再强调一下规则

竞赛的预定时长为3小时

参赛队员仅可以携带无计算功能的铅笔或水笔入场

不能携带任何可用计算机处理的软件或数据

程序须经过若干测试用例的测试，每个测试用例分配一定分数

可以反复提交，取最高分，提交错误不扣分

2.

参赛队员可以在比赛中的任何时刻尝试解决任何梯级的题目。但只有当一支队伍的基础题总分超过800分时，该队进阶部分的题目分数才被判为有效；只有当其进阶题总分超过400分时，该队登顶部分的题目分数才被判为有效。

基础欠缺的同学努力练习基础题。

最好规定个时间，做8道第一部分的题或8道第一部分的题和4道第二部分的题。

以最高分作为目的。要对时间有把握。

可以针对自己容易错误的内容进行总结，并进行针对训练。

比赛剩下1个小时/0.5个小时时看一下团队榜，如果真的遇到第一部分不足800分，第二部分不足400分的时候，部分同学可能需要回头拿下第一部分和第二部分的分数。

3.

我的做题方法，仅供参考：

选择第一部分八道题 5分两道，10分两道，15分两道，20分两道的顺序做题，分四个阶段。

第二部分的题目，如果遇到短时间无法解决的题目，跳过先，把有把握的题目的先做完，再把剩下的题目做完。当然如果过了不少时间仍然没有思路，跳过先。

看一下时间。如果有模板题，先把模板题做了。如果遇到很难或者短时间拿满分的题目，果断骗分，然后再也不看这道题目。

如果时间只剩下10分钟以内(或之类的)，短时间无法解决第三部分的某道题，即使你会做这道题，也要果断放弃这道题。选择 这道题骗分 / 回到第一部分和第二部分把剩下的分数拿回来(主要是检查错误，发现特殊情况，再仔细看一下题目和自己的代码)。

当然，如果还有大片的题目没有完成，然后时间所剩不多，如1个小时，还有第二部分两道题和第三部分的题目没有完成，看队伍情况。如果队伍第二部分离400分还有很远，那么以做第二部分为主，否则以尽量拿到最多分为主，第二部分做把握的题目+第三部分骗分。估计这时心情也是紧张的，所以就不要做那些你认为短时间无法完成编写和调试的题目了。

4.

在进阶和登顶阶段最先完整获得其中任一题分数的前5位队员，分别依次获得50、40、30、20、10分“先锋奖励”。该奖励分数计入个人总分。当团队进阶或登顶成功后，也计入团队的有效分。

这个看自己的能力了。一来高手如云，二来我们赛区的网络不是很好，开始时间也可能会稍微慢点，估计拿到先锋奖励较难。以我们的现实情况，我不建议这样做。

当然，如果你平时的做题策略就是先第二部分和第三部分，那么这没有问题。但如果平时第二部分和第三部分的题的完成率就不是太高，那我就强烈不建议这样做了。

如果你要这样做，建议平时多训练几次这样的做题技巧，比赛时，果断选择裸题做。

5.

比赛如果拿高分：

1.平时刻苦的训练

2.良好的心态

3.骗分大法

......

4.要有大局观。剩下的时间，如果选择题目和对应的做法拿到更高的分数。

如果时间不多，对于某些题目要果断舍弃。

如舍弃第一部分和第二部分无论如何都发现不了错误的2,3分，舍弃第三部分的大模拟题，拿下第三部分的其中一道用暴力的算法可以获得很多分数的题目。

桌面保留一份代码

新的一题，直接拷贝这份代码

(因人而异)

时间把控

如果有些错误难以发现，先跳过。

有时候，天梯赛的特殊样例太坑了

Part1:

25min-30min

30min-40min

40min-60min

60min-90min

90min-120min

Part2:

30min-40min

40min-55min

55min-80min

80min-120min

120min-150min

大致分割，请对号入座

赛前把易错题和薄弱点再看一遍

感觉分数和难度是匹配的，

如果觉得题目容易做，分值高，

那很可能就是有一些特殊样例特别坑！

1. 比赛考点

几大考点

排序

DFS/BFS

并查集

最短路

二叉树，遍历，层次搜索，二叉查找树

几何:

以清华大学邓俊辉教授为首，

出的几何题都比较难，

建议暴力剪枝，骗分为主

包括

斜率DP L3-009 长城

几何+剪枝 / 凸包 L3-012 水果忍者

目前还没有正解 L3-021 神坛

--------------------------------------------------

出现过的其它算法：

大模拟 L3-019 代码排版

dp L3-020 至多删三个字符

基础几何+大模拟 L3-005 垃圾箱分布

基础几何+变形dijkstra 森森美图

树状数组+二分 / 神仙二分堆 L3-002 特殊堆栈

贪心+线段树 L3-017 森森快递

--------------------------------------------------

另外可能还会出现的算法：

最小生成树

更多样的dp

几何创新题（熟悉极角排序，叉积）

更灵活的最短路，二叉树

更隐蔽的并查集，DFS，BFS

不排除给出几道很裸的模板题

1. 题目难度及变化

仅供参考

第一部分和第二部分的超纲题较少。仍然保持出基础题，模板题的传统。

第三部分的题目越来越灵活，越来越难。但骗分是毫无问题的。

仍然保持着出神仙数据的习惯。一些特殊样例很难想到，需要非常细心才行，请自行取舍一道题拿下剩余的分数是否在时间上划算。