

利用假期空闲之时，将这几年 GCJ, ACM, TopCoder 参加的一些重要比赛作个回顾。昨天是 GCJ2006 的回忆，今天时间上更早一些吧，我现在还清晰记得 3 年前，我刚刚参加 ACM 时参加北京赛区 2005 和杭州赛区 2005 的情况。

2005 年 ACM-ICPC——酸甜苦辣

我进入清华大学开始本科学习的时间是 2004 年 8 月，在进入清华大学的第一年里，由于基础课学习比较紧张，再加上计算机系不允许大一学生自带电脑，我没有参加 2004 年的 ACM 比赛。不过在大一一年中没有停止这方面的练习，对 ACM 还是热情高涨。

大概在 2005 年 7 月底，与同班同学 shell（贝小辉）和 superzn（张宁）一起决定组队参加 ACM 比赛。对于队名没有太多的想法，就随便取了一个字典序靠前一点的 bomber。随后进行的几场训练中，我的编程状态一直保持得很好，训练比赛的主要方式都是：我主写程序，shell 和 superzn 负责翻译题目，思考算法和测试。这种组队模式一直沿用到我们后面的所有比赛中。

2005 年底，我们报名参加了 2005 年的北京赛区和杭州赛区的比赛。顺利通过了预赛进入了现场决赛。记得当时北京赛区预赛的时候，我和 superzn 一起在参加百度之星程序设计大赛，shell 依靠一人之力过了 6 题，最后以第二名的资格参加北京赛区现场比赛。

北京赛区：

2005 年的北京赛区地点设在隔壁的北京大学，由于交通非常方便，我们没有和大部分选手住在一起，不过也没有参加 Java-Challenge 和晚上的表演。

练习赛之前，说到比赛位置抽签，本身意义不是很大，可是邬老师神奇的 RP 把两只清华的队伍抽在一起，结果练习赛进行了一半，另一只清华的队伍 THU1

（队员是：吴景岳，栗师和金凯，好像后来队名改成了 DreamCatcher，不是很确定）被要求换到一个比较远的地方，理由是有些学校觉得这样不合理。后来很多赛区也出现过队伍座位在一起的情况，邬老师的 RP 果然不是盖的。

记得练习赛时和复旦的 LemonTree（盛城）一起在场地里闲逛，结果果然不到 10 分钟就被要求回座位了。还有当时比赛场地是一个体育馆，有些队伍把气球放飞之后气球就飘在天花板下了，总裁判李文新老师还威胁我们说，如果明天正式比赛把气球放飞，就不算通过相应的题目，除非有办法把气球取下来。

然后就是比赛的过程了，下面有底纹的文字是我找到的当时留下的比赛总结：

E：快速排序。5 分钟 1Y。

我想 5 分钟的时间可以争取这几年 ACM 国内赛区的最快出题记录了吧。

G：二分答案+最小生成树。25 分钟 1Y。

这题就是经典的最优比例生成树问题，我们一致认为这题比较简单。不过后来被李文新老师批评了，说法是误导其他的队伍。不过说到最优比例生成树问题，TCO2006 的时候 fwj 和 tomek 竟然都没有见过这道题目，这题可是源于 POI 呀。我想我们认为这道题目简单的主要原因是我们都在冬令营上见过这道题目，如果第一次看见，想出算法可能确实需要一些时间。在这里向被我们影响的队伍道歉，最终 G 提交了 200 多次，但是只有 8 个队伍 AC。

C: 二分图最大匹配。42 分钟 1Y

题目要求计算一张图的最小覆盖集，可能唯一的 tricky 是发现图是二分图。

D: 遇到了一定的困难，发现 A 很简单，于是先放一下

D 是一道比较综合的题目，设计一些简单的计算几何和字符串处理的知识。

A: 简单的几何问题，出现了一个低级错误，提交了 3 次均为 WA。

A 是北京赛区最简单的题目，我的程序里犯了一个很低级的错误，可能也是经验不足造成的吧。

D: 重新写，但是没有考虑一种情况，WA 了 1 次。

87 分钟，复旦的 Abuacus 过了 4 题占据了 Rank1。由于队伍模式的原因，我们在还有很多简单题目的情况下卡住了长达 30 分钟。

A: shell 突然发现了 A 程序中的低级错误，105 分钟 AC，重新夺回 Rank1。

这是很重要的一步，现在想来如果没有这个发现，后果可能不堪设想。

B: 二分答案+2SAT。129 分钟 AC。

B 是一道明显的 2SAT 问题，由于题目比较长，我们没有很早发现这道简单题。

D: 发现了 D 的没有考虑的情况，140 分钟 AC。

看了一个 board，那时 Abuacus, Eccentric 都只有 4 题，能够在第一次参加正式比赛就做到 6-4 的领先，当时心情很激动，不过由于缺少经验，也影响了接下来的发挥。其实，现在回想起来，这次比赛其实是一个很好的 AK 的机会。

F: DP。程序比较复杂，WA 了 4 次。

F 是一道比较复杂的动态规划的题目，其实 WA 的原因是一个应该用 int64 的地方，我们使用了 int，这个地方的确很难发现。

H: F 一时无法 AC，只好转攻 H。H 就是普通的模拟题。开始没有考虑坦克和炮弹可能在 1/3 秒相遇，WA 了 1 次。

比赛还有一个小时，封板。

H: shell 发现了坦克和炮弹可能在 1/3 秒相遇的情况，250 分钟左右 AC。

对于我们这种组队模式，当主写程序的选手状态不好的时候，很容易出现连续卡题的情况，这种情况的出现很不利于水平的正常发挥。在北京赛区的比赛中，我们很幸运没有出现连续卡处的情况。

记得，当时北京赛区的 Judge 是半自动的，就是说如果结果是 AC，速度就会非常快，否则由于人的介入，不能 AC 的提交往往需要等一段时间。我们第 2 次提交 H 之后，没有得到很快的回复，以为已经 WA 了，于是我和 superzn 继续测试一些数据。但此时，突然有一个 mm 从左边走过来插气球，这个气球也成为了全场唯一的蓝色气球，这个意外之喜最后成就了第一个分区赛冠军。

F: 下面就是痛苦地提交 F，一直战斗到最后一刻，WA 了 14 次，留下了北京赛区最大的遗憾。

在最后时刻我们似乎发现了那个 int64 的错误，不过当时思路已经比较混乱了，没能改对。F 的问题也导致没有时间写 I，当时如果直接重写后者换 superzn 来写 F，完全可以在比赛结束前 AC。

比赛的大致过程如上所述，那个神奇的气球，我现在仍然记忆犹新。最终有 4 个队伍攻破 7 题，Abacus 的组成应该是盛城，timegreen 和 suzhan 吧，Eccentric 中

我只记得辛韬，ZSU_Panku 中我记得 Savior（陈实）。上述的老朋友之后见面的机会就很少了，分区比赛也成为了我好需要老同学重要的交流机会了。

我 ACRush 的 ID 估计就是那时开始使用的吧，转眼就已经 3 年多了。

比赛前后还记得经常与复旦大学的吴永辉老师聊天，在那之后的每次比赛我都能见到他年轻的身影。

现在回想起北京的分区赛，很有幸能够在第一次参加 ACM 正式比赛就获得分区比赛的冠军。我想是由于现场气氛对许多队伍都有不小的影响吧，当时许多队伍都卡在几道比较繁琐的题目上了，题目的算法性都不是很强。我大概从那时才刚刚接触 TopCoder，如果能够早一些，相信会更适应这样的比赛。

杭州赛区：

2005 年的 ACM 杭州赛区比赛在浙江大学举行，杭州赛区的时间就在北京赛区结束后一周，最初选择杭州赛区的原因很飘逸：我自己家在杭州。实际上也差不多，我随队伍（当时 THU 派了 3 只队伍参加杭州赛区的比赛，除了我们队之外，b142857（侯启明），zhy（周源），ysy（杨思雨）组队，另外一只由汪汀，王俊和黄源河组成）一同抵达杭州车站之后就马上回家休息了，直到比赛前才赶回。在北京到杭州赛区之间的一周中，我的状态就在不断下滑，在家中完全失去了比赛的气氛，回到赛场再也找不到感觉了。一场悲剧即将上演。我们先看看比赛过程吧，下面有底纹的文字是我找到的当时留下的比赛总结：

G：初看很简单，但是调试了 30 分钟没有结果。

G 是一道数学问题，其实《具体数学》书上有明确的公式，不过我们使用的递推方法应该也可以得到正确的结果。程序中犯了一些低级的错误，由于实在不在状态，调试了 30 分钟还没有找到错误。这里还暴露了一个组队模式的问题，在后来的组队模式中，如果像这样没有想清楚算法的题目队友是一定不允许我去写的。

A：模拟。41 分钟 AC，当时肯定没有想到这是唯一一道 1Y 的题目。

A 是一道模拟题，1Y 的时候已经很晚了，排名也很靠后。

C：图论。但是由于堆栈逸出 RTE 了 5 次，浪费了大量的时间。

C 的问题关于树中祖先关心的判定，题目很简单，实现的方法也很容易，就是通过一遍 DFS 来计算。但是我们忽视了一个从来没有遇到过的问题：堆栈溢出。而且，堆栈在本地机器上运行过程中，Eclipse 提供了 8MB 左右的堆栈，所以没有溢出，但是在提交之后的环境下运行就溢出了。而且每次 RTE 之后，我们一直在尝试修改数组的大小，一直没有找到根本原因。调试 C 的同时，我也尝试修改 G，结果 G 也错了 8 次之多，并且始终都是 WA。

I：shell 在我郁闷地调试 C 和 G 中 AC 了，之前 WA 了一次。

I 是动态规划问题，WA 一次可能是忽视了一些边界情况。

D：网络流，没有想到先贪心进行优化。TLE 了 5 次最终没有通过。

D 就是计算最小割，我们事先准备了先流推进算法，不过根据这道题目的模型，先流推进算法遇到最坏情况：二分图。由于当时 dinic 还不是很流行，我们 TLE 了 5 次还没有通过。

郁闷地调试 D 和 G。

E,B：都尝试过，但是都出现了不明的问题。

在随后的时间里，不断调试 D 和 G，但是始终不能 AC。之后又尝试 E 和 B，E 通过分段的方法可以处理，B 是数学题目。正常的话 E 和 B 并不是很困难的题目，但是当时已经非常混乱，连样例都没有通过。

最终我们只过了 3 题，排在 21 名，经历了 I 参加 ACM 以来最惨痛的失败。这次失败主要归过与我状态太差，基本上什么题目都不能顺利通过。当然题目的选择也有很大的问题：G 确实不是难题，但是由于未知的原因始终不能通过，后来我把纸上的程序敲在 ZJU 上就 AC 了，至于现场为什么不能 AC 我现在还是不能明白。如果说第一题的选择直接影响了我们的信心，那么 D 的堆栈溢出则完全打乱了我们的节奏。对于我们的组队模式，卡出 2 题已经超出了极限，我们不可能再尝试其他题目。

Abacus 也来到了杭州，他们前期体现了强劲的先期优势，在 2 小时就达到了 6 题；b142857（侯启明），zhy（周源），ysy（杨思雨）的队伍表现得相当神勇，在最后一小时超越了 Abacus，夺得了冠军。

杭州赛区的失败至今仍是心中痛苦的回忆，不过这个教训也是对我今后的学习生活的一种警示。

总结：

2005 年是我第一年参加 ACM-ICPC 的比赛，两场 ACM 分区赛，我们经历了夺冠的兴奋，也经历了环顾四周等待比赛结束的无奈。2004 年清华没有获得任何分区赛的冠军，2005 年清华打了个漂亮的翻身仗，先后在成都，北京和杭州夺得冠军，而且是三支不同的队伍。

两个赛区的 G 都是有传奇色彩的题目。北京赛区中，我们 25 分钟 1Y 了 G，导致许多队伍跟风失败，最终达到了 208 提交 8AC 的低通过率。但是，杭州赛区中，G 从比赛一开始就占用了我们大量的时间，直到最后都没有通过，估计至少浪费了两个小时左右。真所谓成也在 G，败也在 G。

北京赛区后，POJ 的论坛上传闻说我曾经说过“起身去厕所，不许碰键盘。。。”，很敬仰那些同学搞笑和扯淡的功底，我们虽然定下了以我主写程序的组队模式，但是也非常重视配合和每个人在队伍中的重要作用。

当时清华没有组织校内 PK 选拔，选择了成都赛区的冠军队 THU1 参加全球总决赛，当时总决赛队伍是以参考第二赛区的成绩决定的，现在回想起来也是很合理的。由于最终我们未能得到机会参加全球总决赛，接下来几个月我们情绪低落，bomber 从那时也就宣布解散了吧。

2005 年的比赛过程中，我见到了许许多多的老朋友。用吴永辉老师的话，ACM 竞赛可以看作一些老朋友一起进行的一场智力游戏。

附北京赛区前 5 名：

1 Tsinghua University=>bomber First Place 7 788

2 Fudan University=>Abacus Second Place 7 983

2 Shanghai Jiao Tong University=>Eccentric Second Place 7 1084
3 ZhongShan (Sun Yat-sen) University=>ZSU_Panku Third Place 7 1194
4 Peking University=>Monkey King Fourth Place 6 768

找不到杭州赛区的排名了，只发现了这个：

21 THU *bomber3 501 1/41 0/-- 6/197 5/-- 0/-- 0/-- 8/-- 0/-- 2/143 22

谢谢韩家龙同学的热心帮助，找到一个排名的链接是：

<http://acm.zju.edu.cn:8080/icpc2005/ranklist/index.html>

利用假期空闲之时，将这几年 GCJ, ACM, TopCoder 参加的一些重要比赛作个回顾。GCJ2006, ACM2005 和 TCCC2006 之后，2006 年对于我来说是一个大丰收，今天晚上先回顾 Mobile Robot 成立先期的事情吧，明天再总结惊心动魄的 ACM 上海 2006 和 ACM 西安 2006 吧。

2006 年 ACM-ICPC(上)——Mobile Robot 的成立初期

回忆到 2005 年清华没有组织校内 PK 选拔，选择了成都赛区的冠军队 THU1 参加全球总决赛，bomber 从那时也就宣布解散了。

早在 2006 年初，THU1 准备参加 ACM-ICPC 2006 世界总决赛的训练时，我们的队伍就已经成立了。队伍其他两名选手是一起参加 IOI2004 的 geworm(鬲融)和 wd.h(胡伟栋)。

Mobile Robot 的组队比赛：

至于 Mobile Robot 的队名，我们是为了纪念 2004 年 4 名参加 IOI 的选手第一次合作的时候使用的帐号，如果回到 2004 年的 PKU 月赛，也许可以看到 thmr3191 的身影，这个 ID 最初是我们 4 人共同使用的。其中 thmr 就是 Tsinghua Mobile Robot 的缩写。当然我们觉得 Mobile Robot 读起来也比较容易上口。

Mobile Robot 成立之后做的第一件事情就是配合 THU1 准备 World Final2006 的训练，先后模拟比赛了两次 Northeastern Europe(NEERC)的题目。这两次训练中，我们队伍的主要模式都是：

- (1) geworm 全程负责读题，思考算法和出数据；
- (2) wd.h 和我在比赛前 2 个小时一起攻简单的题目；
- (3) 2 小时后 wd.h 就开始死磕难题，我主写程序一直到 3 个半小时左右，结合 wd.h 对难题的把握，大家开始合攻难题。

这种拖后中卫的打法，对于 NEERC 的题目难度非常合适，两场比赛我们都做到了 AK（全过 11 题）。这种组队模式也一直沿用至总决赛。当时 wd.h 的状态很好，对于 NEERC 的题目难度，我觉得世界上很难有队伍能够有信心做到 AK。

队伍成立初期的顺利使我们更有信心，我们利用暑假时间进行了一些必要的训练以迎接 2006 年下半年的 ACM 分区比赛。

北京赛区预赛——网络赛赛网络：

2006 年下半年有 3 个国内赛区，包括北京，上海和西安，其中北京赛区最先举行。2006 年北京赛区的地点设在了清华大学，这也是我唯一一次参与组织 ACM 分区比赛的机会。

10 月 中旬举行了北京赛区网络预赛，网络预赛的参与者是所有报名参加北京赛区的队伍，以决定哪些队伍拥有参加现场比赛的资格。那段时间，我们队伍主要精力放在了 准备比赛上，我们都没有参与网络预赛的命题和测试平台工作。由于

清华距离上次承办分区比赛已经相隔很多时间，直接导致网络比赛过程中出现了严重的网络问题，在这里作为清华 ACM 队的一员向受到影响的队伍道歉。

不过，我也是作为“局外人”来了解这次网络阻塞的，因为我确实没有参加任何与网络赛有关的活动。现在回想起来，我认为平台的稳定性是一个不可推卸的原因，但是主要应该归咎于题目描述和样例的设计，当然还有测试数据的错误。设想这样一种情况，如果一个比赛过程中，从某一时刻起，突然增加 1000 个提交需要 rejudge，然后所有队伍还都在这一时刻起尝试提交，我想现有的大部分 OJ 都很难在 1 小时之内平息这些提交吧。再举一个更夸张的例子，如果 OJ 准备的测试机器的测试速度已经完全跟不上提交的速度，那么卡住是不可避免的。我们通过网络预赛的教训总结出一些网络预赛题目的重要经验：

- (1) 对于容易上手的题目，测试样例一定要足够强。
- (2) 对于简单的题目，必须仔细确保测试数据是正确的。
- (3) 题目描述必须没有任何歧义，避免选手通过提交来不断尝试各种理解。

如果题目能够很有效控制提交数目，对于测试系统的要求其实不是很高。例如复活赛和现场决赛的时候，测试系统会大部分时间处在空闲阶段。反之，如果提交处在上述病态的情况下，只有非常专业的测试系统才能胜任这样的挑战，当然不包括我们的测试系统。

总之，对于网络问题我作为清华 ACM 队的一员深表歉意，如果还有下一次的机会，我们一定努力做得更好。

北京赛区验题赛：

如果说网络预赛过程中，网络出了一些问题，那么，决赛则是结果更出乎我们的意料之外。在北京赛区现场赛之前几天，我们 3 支队伍进行了验题赛，比赛虽然不正式，但是过程仍然很激烈。

先列一下决赛的 9 道题目吧：

- A. Robot
- B. Animal Run
- C. Another Minimum Spanning Tree
- D. Connect It, If You Can!
- E. Guess
- F. XAR
- G. What a Special Graph
- H. Ruler
- I. A Funny Stone Game

现场赛只有 BEHI 这 4 道题目有队伍成功通过，可是在验题赛中我们队伍的进程完全不是这样，下面是我们的做题情况：

22 分钟 A 题，数学方法，1Y

首先，我们 3 支队伍在 30 分钟之内都 1Y 了 A 题。A 题是一道中等难度的数学题，可能 A 题需要明确高次等差数列的求和公式，而且通过枚举来代替一些假设可以大大简化问题。现场比赛时有些队伍做了不正确的假设导致始终 WA。

记得当时 zhuzeyuan 使用了一个奇怪的贪心方法，后来被 OpenGL 找到一个反例，这个测试用例被添加到正式比赛的测试数据之中，这个反例也成为了现场赛中使得许多提交 WA 的重要数据之一。

30 分钟 E 题，贪心，1Y

E 是 2006 北京赛区最简单的题目，只需要直接的贪心法就可以解决。

52 分钟 H 题，深度优先搜索，1Y

H 是一道搜索题，题目时限不是很紧，不需要太多的优化就可以通过。

75 分钟 I 题，标准的博弈 SG 问题，1Y

I 是标准的博弈问题，通过计算 SG 就可以得到结果。这题其实有一个阴险的地方，就是当某位置石子为大于 0 的偶数时，也需要考虑以保证结果的字典序最小，好在我们及时避开了这个陷阱。现场很多队伍调入这个陷阱中，耽误了一些时间。

129 分钟 B 题，最短路径问题，3Y

B 是一张平面图的最大流问题，由于图形比较有特点，所以可以建图来计算最小割。但是这张图有 10^6 个点， $2 \cdot 10^6$ 条边，最短路径需要用堆来辅助实现，首先由于数组开小了 RTE 了一次，然后由于用 map 实现 TLE 了一次。这题浪费了许多时间。

G 题，实现和调试了 30 分钟，超时

G 是 2006 北京赛区最困难的题目之一，题目描述很简单，判断一张图是否为 co-graph。我们算法的复杂度是 $O(n \cdot m/32)$ 的，不过由于数据个数比较多，程序运行时间远超过了时限。

C 题，贪心法实现 50 分钟，WA

C 题是计算平面图曼哈顿最小生成树，直接计算是 $O(n^2)$ 的，但是题目中 n 接近 100000。我使用了一个贪心算法，其实和标准算法差距不大，不过还是导致 WA。其实提前写 C 不是很合理的选择，当时没有注意到 D。C 和 G 难度相差无几。

230 分钟 F 题，构造，1Y

F 题是很变态的构造问题，这题完全是 wd.h 做的，我至今还不是很清楚算法。

250 分钟 D 题，计算几何，2Y

D 题是一道比较复杂的计算几何，当判断一条直线是否穿过一个多边形的时候忘记考虑了一种情况，WA 了 1 次。现场许多队伍其实都只忘记考虑了这种情况，但是可惜没有队伍该正确。

这场比赛最终我们队伍以 7 题结束，另外两队也都通过了 7 题。我们因此也没有修改题目难度，随后让大家没有想到的是：一场极低通过率的比赛即将开始了。

北京赛区现场赛：

现场比赛中，我负责在某一个房间为参赛选手送打印资料，比赛 60 分钟左右由于技术问题到 Judge 室处理一些问题。经过 5 个小时的比赛，最终中科大 Student 队通过 4 题获得冠军，厦门大学 btALT 通过 3 题获得亚军，北京大学 RPWT，浙江大学 gogogo 和合肥工业大学 Love Wisdom 也都通过 3 题分列 3-5 位。最终 Student，btALT 和 Love Wisdom 进军总决赛。

回顾比赛现场过程，首先让我们出乎意料的是 E，E 是 2006 北京赛区中最简单的题目，贪心法的方法参加比赛的同学都想到了，可是有一个小小的细节，对于实数比较大小时，需要加入一个微小量 `eps` 来控制精度。E 题没有加 `eps` 的提交占到总提交的 50% 以上，我们称之为“经典提交”。这个小 `tricky` 不慎导致很多队伍迟迟不能通过第一题，对许多队伍的状态有不小的影响。

其次是 A，A 题收到了很多队伍的提交，但是最终都没有队伍通过 A。原因是大家做了一些不保证正确的假设，当时我们都通过枚举的方法避免了这些假设。

另外，有一些队伍提早接触了 F 和 G，并深深地陷入其中。中科大早在 240 分钟就通过了第 4 题，可是之后他们在 G 上花费了不少精力，我们甚至想跑到他们那里告诉他们 G 是最难的。

记得 180 分钟到 240 分钟，我们只接收到了不超过 10 次提交，每次大家听到提交的声音，所有 Judge 一起点鼠标抢测试权。后来，在 TCCC2006 上和 Ying 说起此事，据他说和 2004 年的广州赛区有许多相似之处。

总结：

祝贺所有获得好成绩的队伍，恭喜 Student，btALT 和 Love Wisdom 进军总决赛。并再次对网络赛给大家带来的不便道歉。后来清华举行了名为复活赛的比赛，我想复活赛应该就是从那时开始出现的吧？

当年清华一共有 6 支队伍，但是只参加两个赛区的比赛，造成每个赛区之前都要进行小规模 PK，最终只有 4 支队伍有机会参加 ACM 分区赛。Mobile Robot 建立之初比较顺利，获得了参加两个赛区比赛的机会，迎接 Mobile Robot 的将是上海和西安赛区的挑战。

我们 3 人能够走在一起，要感谢吴文虎老师的支持，组队初期虽然没有经历大战，但是那些快乐的时光至今都很难忘怀。

附北京赛区排名

- 1 University of Science and Technology of China=>Student First Place 4 628
- 2 Xiamen University=>btALT Second Place 3 417
- 3 Peking University=>PKU RPWT Third Place 3 460
- 4 Zhejiang University=>gogogo Fourth Place 3 471
- 5 Hefei University of Technology=>Love Wisdom Fifth Place 3 477
- 6 Fudan University=>Yin-Yang Sixth Place 2 204
- 7 Tianjin University=>TJU_Rhythm Seventh Place 2 212
- 8 Beihang University=>L3 Eighth Place 2 254
- 9 Peking University=>PKU4 Ninth Place 2 275
- 9 Harbin Institute of Technology=>IAC Ninth Place 2 282

10 ZhongShan (Sun Yat-sen) University=>ZSU_Pyrenean Tenth Place 2 288
11 Zhejiang University=>Phoebus Eleventh Place 2 311
11 University of Electronic Science and Technology of China=>Gryffindor Eleventh Place
2 319
12 ZhongShan (Sun Yat-sen) University=>ZSU_Himalayas Twelfth Place 2 324
12 Fuzhou University=>OrOrz Twelfth Place 2 412
13 Beijing University of Posts and Telecommunications =>Vitamin Thirteenth Place 2 432
14 Ningbo Institute of Technology,Zhejiang University=>impact Fourteenth Place 2 435
15 Huazhong University of Science and Technology=>rodimus Fifteenth Place 2 442
16 Nanjing University=>PHOENIX Sixteenth Place 2 459
17 Fuzhou University=>Laplacian Seventeenth Place 2 467
17 Fudan University=>Shuangwaiwai Seventeenth Place 2 500
17 National University of Defense Technology=>Robust Seventeenth Place 2 517
18 Fudan University=>Free Wings, Yeah! Eighteenth Place 2 518
18 Hong Kong University of Science and Technology=>HKUST1 Eighteenth Place 2 692

利用假期空闲之时，将这几年 GCJ, ACM, TopCoder 参加的一些重要比赛作个回顾。GCJ2006, ACM2005 和 TCCC2006 之后，2006 年对于我来说是一个大丰收，昨天晚上回顾了 Mobile Robot 成立先期的事情吧，今天先发惊心动魄的 ACM 上海 2006 吧。

2006 年 ACM-ICPC(中)——Mobile Robot 上海对决

回忆到，当年清华一共有 6 支队伍，但是只参加两个赛区的比赛，造成每个赛区之前都要进行小规模 PK，最终只有 4 支队伍有机会参加 ACM 分区赛。Mobile Robot 建立之初比较顺利，获得了参加两个赛区比赛的机会，迎接 Mobile Robot 的将是上海和西安赛区的挑战。

比赛前：空前的豪华阵容

记得清华大学出发上海赛区的时间是 10 月 20 日晚上，至于为什么能记得如此精确，是因为在那之前我经历了真正意义上的“赶火车”。10 月 18 日 TCCC2006 在圣地亚哥落下帷幕，19 日从旧金山机场起飞会北京，飞机着陆时间是 20 日下午 2 点半，进海关之后已经快 4 点了，我立即乘坐机场大巴直奔火车站与大部队会合，之间都没有时间回寝室。

来到中亚饭店报道拿到参赛队伍名单的时候，就赫然发现上海 2006 的参赛队伍实力达到了几年来一个不可逾越的巅峰。上海交大的 1234 队都出现在了名单中，还有浙大和北大的 Final 队都来了，这些还不够，芜湖一中的 Loner(周冬)，上海微软 ATC 的 lympanda 也参加比赛。上海交大一直是这几年来清华国内最强劲的对手，如今交大又占据主场优势，实力深不可测。上海微软 ATC 虽然是旅游队，但是 lympanda 凭借在 TopCoder 上的表现，没有人敢轻视这位无冕之王的实力。

对于上海赛区，清华也派出了华丽的阵容，参赛的有 3 支队伍，除了 Mobile Robot（我，geworm(鬲融)和 wd.h(胡伟栋)）之外，还有鼎盛时期的 Shangri-La（b142857(候启明)，lxd(林希德)和 zhuzeyuan(朱泽园)）以及后来的西安赛区亚军 GotoFly（zcgzcgzcg(朱晨光)，wangjun(王俊)，tedcn(龙凡)）。记得练习赛之前，大家一起围圆桌吃饭，朱晨光突然和旁边的王俊说，好像就我们两个没有参加过 IOI，然后另一边林希德补了一句，就我们三个不是金牌。

我第一次有机会敬仰候启明的时候，是自己第一次参加 NOI——NOI2002 天津，候启明以满分的成绩获得冠军，当时的亚军就是林希德。之后的冬令营，我以非正式营员参加测试神奇得获得第三名，但更重要的是冬令营测试的冠亚军就是候启明和林希德，之后我再没有和两位前辈在 OI 上交过手。时光飞逝，我代表 bomber 队参加 2005 年的杭州赛区，被候启明领军的 Legendary Team 打得一败涂地。随后 2006 年初的 Google Code Jam China(GCJC)2006 上，我获得第三名，记得当天与获得亚军的候启明同住在总统套房。其实在上海赛区之前我没有在正式比赛中战胜过他们。Shangri-La 的另一人 zhuzeyuan 现在是我的队友，记得他当时在 TopCoder 仍然是 Target（你可是早点在今年 Final 之前把 Target 拿回来呀，几次涨停就可以了:P），实力也不在前二人之下。

面对知根知底的 Shangri-La，大家都知道一场大战在即，从实力上分析，我觉得 Mobile Robot 略弱，不过赛前我们都没有信心一定能够在比赛中占据优势。而

且我们心里深知，上海赛区的结果将很有可能直接决定清华大学当年的总决赛队伍。面对这种残酷的现实，我们都无可奈何，有时一些有实力进军总决赛的队伍在清华都没有机会参加分区比赛。

个人的经验看来，我认为在势均力敌的时候，最重要一点是明白自己的优势和劣势所在，要用自己最强的方面来对抗对手，避免暴露出自己的劣势。我想在这一点上 Shangri-La 可能没有我们做得好。Shangri-La 的优势是三人的总体实力很强，他们完全可以采用三大高手的组队模式；我们组队时间长，配合默契，而且当时自己刚刚从 TCCC2006 以 100% 正确率回国，保持了良好的状态。从单人比赛上讲，当时我的状态即使有 Petr 和 Tomek 在场，我都并不认为自己一定会有明显的劣势。

比赛场地是上海大学的一个大体育馆，现场气氛很热烈，想到我们用 4 个机房办的北京赛区的现场比赛，不由地觉得有些寒酸。

在场外碰到了 lympanda，他向我了解刚结束的 TCCC2006 的情况，我于是给他描述了一下几道现场比赛过程中 1000 分的题目，结果全部都被 panda 秒杀了，无限敬仰呀！同时也第一次见到了周冬。

提一件飘逸事情，记得当时练习赛有 3 题，封版时没有队伍通过 B 题，但是其实在封版后我们通过了这题，我们应该是当时唯一在练习赛中 AK 的队伍。记得之后好像还和中山大学的郭老师交流过这题。不过至于 B 为什么能够 AC：B 题是一道需要 SPJ 的题目，可是练习赛的时候没有 SPJ，而我又坚信自己的程序是正确的，于是我不断提交。可能是由于 Judge 不耐烦了，才用 Yes 的方法让我们停止。

比赛之前的晚上我们都休息得很好，第二天早上以充足的精力迎接史诗般的上海赛区决战。

比赛过程：400 米赛跑

我中学是很喜欢参加 400 米比赛，400 米比赛从起跑姿势角度应该认为是短跑，但是 400 米已经远远超过了冲刺极限。所以 400 米跑中，要求我们从发令枪响起的时候就加速启动，直到拼尽全力为止。

我们回顾一下比赛的过程吧，上海赛区比赛之后，我们写下了详细的比赛过程：

按照一贯的方法，鬲融从 A 开始读，胡伟栋从 J 开始读，我准备编程的环境，然后从中间选择题目读。鬲融读完 A 后，发现 A 是一道简单题。于是选择先写 A 题。

A：给定 ACM 网上预选赛的晋级规则，求每所学校的晋级队伍数。

简单模拟题，时间复杂度 $O(N)$ 。本题题目有一个疑问：一个学校出线的队伍数目可能比该学校参加预选赛的队伍数目还多，但是题目描述和样例表明不需要考虑这种情况。

ACM 比赛的第一题的选择对比赛的进程影响很大，当没有优先选择比赛中最简单的题目时，更需要保持冷静。

在我写 A 的过程中，鬲融看了 B 和 C，发现 C 也非常简单，于是马上做 C。

C：将一棵节点带权的树划分成两半，使两半的权值和的差最小。

枚举或者树的遍历，时间复杂度 $O(N)$ 。去年有一个深刻的教训：当遍历树的时候，很容易造成堆栈溢出(杭州赛区留下的疙瘩)。对于本题的范围，保险起见没

有使用 DFS，而是使用 BFS。使用 BFS 略微增加了编程量，但是可以在一定程度上避免不必要的麻烦。

鬲融发现 D 是一道经典的统计题，在不到 3 分钟的讨论后，我们得出了可行的算法，于是下面写了 D。在写 D 的时候，鬲融和胡伟栋把题看完了，经过讨论，决定胡伟栋开始想一道数学题 I，而鬲融继续看题义不是很清楚的 G 和 J 两题。

D：给出平面里的一系列的点，要找一个矩形，使矩形边上的点最多。

离散化后，先判断一条直线的情况。枚举两条横边，然后枚举竖边，一旦一条右边的边比左边的边好，左边的边就不会再有作用。因此，枚举竖边的过程很容易做到 $O(N)$ ，总的复杂度是 $O(N^3)$

这题其实存在 $O(N^2 \log N)$ 的算法，我想出题人也应该没有在第一时间想到吧。

做出 D 后写了 B，原因是 B 的算法相对简单。但是意想不到的是：B 的样例不合法，而且 Clarification 的速度非常慢，于是只好先把 B 放在一边。这个过程浪费了不少宝贵时间。

随后看到 team109(Shangri-La)过了一道红色气球，感觉是 G，于是鬲融给我讲了 G，但是发现 G 实在不像算法简单或程序简单的题目。

此时，胡伟栋推出了 I 的一个很简明的公式。而且我们发现 I 的气球也是红色的。刚才看 G 很可能是被误导的。于是写了 I。

写 I 的过程中，由于鬲融看的题基本已经被做完了，胡伟栋给鬲融讲了 H 的题意，鬲融想出了一个可行的做法，不过由于还有更简单的题并没有马上做。

I：求杨辉三角形第 N+1 行不能被质数整除的数的个数。

可以找出规律，然后写出公式，时间复杂度 $O(\log N)$ 。我在并不知题目意思的情况下写过了 I，依靠胡伟栋的公式，我们度过了此次比赛第一段艰难的时期。

从过 D 题到过 I 题，大概有 40 分钟的时间。这段时间我们主要的失误有：

(1) B 的样例不合法，这其实不是我们的错误。

(2) 受气球颜色的误导，过早思考一道很难的题目。

(3) 后来听朱泽园的建议：由于一个人的问题导致卡住，应该一个人解决，不应该让全队都陷入混乱中。

我认为我们直接选择写另外一道题目主要有两个原因：一是 B 卡住的原因比较特殊，不是我们花时间就能克服的；二是 I 的算法比较清楚，非常稳定。

等 I 过了之后，B 的 Clarification 出来了，Judge 换了一组样例。于是开始调 B。

B：给出 16 个数，将他们排成十字架形，使力矩平衡。求本质不同的方案数。

十字架一共有 4 个“臂”。首先找出一个“臂”的所有情况。然后将等值的无重复的合并成两个相对的“臂”。然后从 16 个数中选出 8 个，将他们作为一组，另外 8 个作为另一组，这种方案的总数为前 8 个成对的方案数乘后 8 个成对的方案数。最后把方案数求和除 4。

本题属于搜索题，而且时限特别紧，由于搜索问题的优化空间往往很大，而且又是多组数据，所以很容易造成 TLE。在调对样例后 TLE 了一次，改进了算法后由于开小数组 RE 了一次，终于在第三次提交通过了 B 题。

记得高中时一次在网上做题(只记得是 xreborner 出的题目)的时候，比赛一开始就写一道时限很紧的题目，估计开始程序的正确性是没有问题的，就是效率比较低；

但是，在不断优化的过程中改错了程序，导致 1 个小时以后当程序不 TLE 了以后，程序变成了 WA。这样使得信心完全崩溃。

听了很多同学讨论之后，发现不少队伍就完全卡在了 B 题上。以后对这种题目只能倍加小心，现在我们还没有什么特别好的方法。而且，B 题的样例只具有测试性，不具有调试性，编程时必须特别仔细才行。

这时大概才 1 个多小时，我们看似顺利地通过了 5 题；但是现场的情况并没有任何优势可言，Shangri-La 在几分钟后通过第 5 题，两个队伍的罚时只相差 1 分钟。

随后我们到达了第二段艰难的时期，主要原因是鬲融把 F 的题目看漏了一个条件，与胡伟栋讨论后误以为这个题比较容易，很快这题出现 Run-time Error 好在重读题后发现错误并及时放弃，没有再浪费时间去把 Run-time Error 调成 WA。如果当时死做此题则后果不堪设想。

在鬲融和胡伟栋分别读 F 的程序以及题目的时候，由于现在剩下的题目都比较复杂，我们选择了一道相对清楚的题目 H 继续做。

H： 对一个集合进行两种操作：插入一个数和询问 MOD Y 最小的数中最后一个数。

由于数字的范围是 $[1..500000]$ ，先选择一个合适的 M。当 $Y \leq M$ 时，通过插入时直接保存来处理，当 $Y > M$ 时直接枚举，用并查集求一个元素的后续。这样每一次操作的时间复杂度是 $O(\text{Sqrt}(500000))$ 。

但是，开始选择 M 为 1000，并没有充分估计复杂度的平衡性。开始我们认为并查集的常数较大，但是后来感觉到并查集的常数相对小一些，于是把 M 选为 500 就过了。后来在同出题人交流的时候，被告之 M 取在 400-800 的范围内都可以。

第二段艰难的时期的原因可能不仅仅是题目看漏，也由于题目难度已经增加。好在当时特别是当 H 的程序 TLE 之后，我们都比较冷静，相信自己 H 题的算法是正确的算法，只是参数的选择不够合理。这种考验在平时一般是很少遇到的，经受这次考验之后，再遇到类似的问题，我们应该能够更冷静一些。

过了 H 之后，鬲融已经再次确认了 J 的题意，随着气球的指引，我们决定攻克本次比赛最大的“纸老虎”：J，而做 J 的过程中胡伟栋一直在想 G，已经大致得到了算法的框架。

J： 给出一些数的大小限制的关系，求更精确的关系后者判断为矛盾。

转换成图的模型，不断迭带调整，直到不能调整为止。当然这题由于大小限制关系中既有 ' \leq ' 还有 ' $<$ '，所以需要判断 $XI < XI$ 是不合法的。时间复杂度 $O(N^3)$ 。

J 题的数据输入输出比较复杂，但是题目本身很简单。这题对编程能力提出了很高的要求。

通过 J 题之后，看了一下 board，当时 Shangri-La 只有 5 题，不过在我们还没有反应过来的时候 Shangri-La 就也同样 7 题了，罚时上我们领先 4 次提交。

在比赛的前 200 分钟，我延续了 TCCC2006 的良好状态。我们配合默契，在面對 BHJ 这样琐碎的题目时，队友会提前把需要注意的细节总结在纸上，整个过程都保持得很平滑。另外，鬲融和胡伟栋在我做每一题时都准备了很合适的测试数据，大大减小了我测试的时间并很有效地提高了提交正确率。由于剩下的题目难度明显高出一个档次，在通过 7 题时的罚时领先是最后获得比赛胜利的重要砝码。

现在剩下的只有 E, F, G 三道没有队通过的题目了, 其实最终也没有队伍通过。我们曾经读错过 F, 因此这次分别尝试了 E 和 G, 但是都失败了。这里我们曾经讨论过先做 E 还是先做 G, 当时面临的选择是: E 的复杂度估计较高, 不知优化后能否通过, 而 G 的算法性更强, 胡伟栋仍然没能完全清楚如何解决, 实现的复杂度高于 E。我们这次带有赌博性的选择了 E, 并没有仔细考虑如果 E 的时间要求过于严格会出现什么问题(可能与 B 的相对轻松通过有一定关系), 在此后的比赛中我们应该注意周全考虑, 尽量选择题意清楚并且复杂度容易估计的题目。

E: 带限制的有向图的第 K 短路。

本题其实有标准的 A* 算法, 我们也使用了这个算法。但由于复杂度过高, 我们的程序一直 TLE。据 Judge 说这题对时间的要求非常严格。

G: 在一个森林的若干点布置仪器, 仪器有作用范围 D, 仪器之间的距离需要大于等于 D, 求最大的覆盖长度和最小的权。

本题可以使用二次方状态的动态规划, 类似 CTSC2004 的一题。我和胡伟栋讨论后成功想出了正确的方法, 并且在最后时刻写出了程序, 而鬲融和胡伟栋则出了一些测试数据, 将数据调过后时间已经很紧张了。首先由于忘记优化 floyd 超时了几次。在优化了 floyd 算法之后, 还是没有考虑到图不连通的情况, 未能在比赛结束前调出。

F: 给出一些公理, 假设和定理之间的关系, 依次尝试, 推出尽可能多的定理。在同等情况下要求使用的假设最少。

本题可以用最小费用最大流解决, 但是比较容易实现的网络流算法都很难在时限内出解。

现在看来, EFG 三题中的 E 没有比赛中想像得那么难, 当时比赛中受到巨大压力的影响没能攻破此题, 不过 FG 题即使出现在 CTSC 难度的比赛上也非常合适。

比赛结束: 罚时险胜

比赛结束时我们与 Shangri-La 同为 7 题, 上海交大一队最终也通过了 7 题, 我们依靠罚时的微弱优势险胜。经过惊天地泣鬼神的 300 分钟, 我们终于获得了梦寐以求的 2006 上海赛区冠军。Mobile Robot 凭借着上海赛区的夺冠, 已经可以认为获得了进军 2007 东京世界总决赛的入场券。

比赛结束后见到了很多复旦的老朋友和吴永辉老师, 吴老师还是和以前一样, 玩笑开个不停。那些复旦的老朋友赛前由于是参与出题, 我们一直没有看到, 颁奖仪式之前, 我们终于有机会在后场一起聊天。不过聊天这些时间中, 我和 b142857 错过了郭老师在颁奖仪式前进行的“点名”(当时郭老师要求我们上去讲解题目), 实在不好意思。

总结:

首先向 Shangri-La 致敬, 即使在最后一刻, 相信大家都还仍然有机会, 棋逢对手也是我 ACM 生涯中的一大幸事。当两支队伍都通过 7 题的时候, 排在第 3 名的队伍才刚刚 5 题。也就是说, 我们两支队伍其实只用了 2/3 的比赛时间就锁定了上海赛区的冠亚军。

上海交大一队最终也通过了 7 题，比赛开始时我们就注意到了交大开场时非常不顺利，不过顽强的交大一队稳扎稳打，在最后一小时也成功通过了第 7 题。在这里向交大致敬，开场的种种不利不亚于我在杭州 2005 时的起跑，你们能够沉着应战，破釜沉舟的精神一直值得我们学习。当时交大一队中有一位来自辽宁的选手辛韬，记得 NOI2003 之前我们有数次交手都以我失败告终，后来经常在许多网上比赛中切磋，2005 年 ACM 北京赛区也有你熟悉的身影，在清华早有耳闻你在交大的优异成绩，祝福你今后越来越好。

这次比赛的命题工作是由复旦大学担任的，复旦大学的命题特点与东欧的命题风格很接近，题目的算法性偏强，题目对程序运行速度要求普遍高。

按照 panda 的说法，上海微软 ATC 队由于一些配合的失误，最终只通过 5 题，不过也是前 10 的队伍之一。另外，芜湖一中队也顺利通过 5 题排进前十。GotoFly 通过第 3 题的时候还排在第 3，不过后来卡在 J 上了，有些可惜。

后来，分配总决赛名额的时候，上海赛区得到了 10 个总决赛名额，这个数字相信也是这些年来之最吧。

这是一场值得纪念的比赛，参加 2006 上海赛区的强队实力远高于几年内的各大赛区，能够站立在上海赛区的最高领奖台是 Mobile Robot 获得的最高荣誉之一。经历了上海赛区大战的洗礼，接下来的西安赛区出场的则是更成熟的 Mobile Robot。

附上海赛区排名：

- 1 Tsinghua University => Mobile Robot 7 617
- 2 Tsinghua University => Shangri-La 7 709
- 2 Shanghai Jiao Tong University => Lacotix 7 1034
- 2 Fudan University => Symphonic Rain 6 924
- 2 Shanghai Jiao Tong University => Prodigies 6 957
- 2 Fudan University => ShuangYY 6 1098
- 2 Xiamen University => btALT 5 534
- 3 National Taiwan University => puyo 5 563
- 4 The First Middle School of Wuhu => WHYZ 5 652
- 4 Microsoft ATC Shanghai => Core-Stop-Dump 5 729
- 4 Zhejiang University => Golden Keyboard 5 817
- 5 Peking University => PKU_T2 5 819
- 6 Zhejiang University => Dasher 5 956
- 6 East China University of Science and Technology => Redfield 4 325
- 7 Tsinghua University => GotoFLY 4 367
- 7 Zhejiang University => sago 4 392
- 7 National University of Defense Technology => Robust 4 438
- 8 Zhongshan University => ZSU-Tanglha 4 512
- 9 Tongji University => Revenge 4 559
- 10 Shanghai Jiao Tong University => H-E-A-T 4 600

10 Xidian University => ACMore1 4 602
11 Wuhan University => Moonmist 4 723
12 Zhongshan University => ZSU_Olympus 4 765
12 Nanjing University of Science and Technology => Narcissus 4 774
13 Northwestern Polytechnical University => 1010 1100 4 845
14 Harbin Institute of Technology => Gaminerie 3 200
15 Fudan University => White * [Bear + Dew + Cloud] 3 337
15 Zhongshan University => ZSU-Everest 3 360
15 Hefei University of Technology => Happy For Unpain Teeth 3 406
16 Beijing University of Posts and Telecommunications => Neon 3 442
17 Shenzhen University => Aspire 3 453
18 Harbin Institute of Technology => Corsair 3 457
18 Nanjing University of Science and Technology => backbones 3 461
18 South China Agricultural University => scau_update 3 468
19 Huazhong University of Science and Technology => rodimus 3 480
20 Ningbo University => ACHC 3 505
21 Peking University => www_pku 3 513
21 Fuzhou University => OrOrz 3 520
22 National University of Defense Technology => Puzzle 3 520
22 Donghua University => Gespenst 3 535
22 Hunan University => Footmen 3 535
23 Tianjin University => TJU_Rocket 3 563
24 National Taiwan University => ABCDE 3 568
24 Nanjing University of Aeronautics and Astronautics => ZipFish 3 575
25 University of Electronic Science and Technology of China =>
transistor 3 599
26 Shanghai Jiao Tong University => Lucifer 3 608
26 Zhengzhou University => ZZU_cheapwine 3 650
27 Wuhan University => Silence 3 685
27 Hangzhou Dianzi University => MP3 3 687
28 Jinan University => JNU-FLY 3 830
29 Huazhong Normal University => Neptune 3 857

利用假期空闲之时，将这几年 GCJ, ACM, TopCoder 参加的一些重要比赛作个回顾。GCJ2006, ACM2005 和 TCCC2006 之后，2006 年对于我来说是一个大丰收，晚上发 ACM 西安 2006 吧。

2006 年 ACM-ICPC(下)——古城西安

艰苦的 ACM2006 上海赛区结束之后，我们原本以为清华会选择另外 3 支队伍参加西安赛区的比赛。况且，大三的课程实验任务很重，我们也就停止了计划的定期训练。大概在 25 天之后，12 月 20 日左右突然收到邬老师的通知，准备出发参加 ACM 西安赛区比赛。我们的 2006 西安之旅也就是在这比较仓促的准备中开始的。

西安赛区之前，我们没有定下明确的目标，比赛过程中处处都采用了追求稳健的方法，当时也是为了避免一年前的杭州悲剧重演。

西安的比赛没有太多兴奋的 AC，没有惊心动魄的场面，所有过程都在类似旅游的气氛中结束了。作为 Mobile Robot 的最后一战，这里也作一个简要的回顾吧。由于这次题目又是 Srbga（刘汝佳）命题的，我最后也顺便列举一下我总结的他命题的特色吧。

比赛过程：

2006 年清华同样派出了 3 支队伍参加西安赛区，除了 Mobile Robot（我，geworm(鬲融)和 wd.h(胡伟栋)）之外，还有 Panacea（OpenGL(唐文斌)，zhy(周源)和 liuhe(刘贺)，如果你第一眼看见这个词就知道是什么意思，我相信您一定准备过 GRE 吧？）以及一起参加过 2006 上海赛区比赛的 GotoFLY（zcgzcgzcg(朱晨光)，wangjun(王俊)，tedcn(龙凡)）。

2006 年冬天是我第一次来到西安，刚下火车就深深感受到了西安的古城气息。到达宾馆之后，我们做得第一件事情就是玩，印象很深的是钟楼，鼓楼还有大雁塔。我们做的最主要的事情可以用 GotoFLY 的 4 个大写字母 GFLY 来表示，就是“公费旅游”。

比赛之前的晚上，我们认真讨论了比赛中采用的策略，参加西安赛区没有太多传统强队，我们一致觉得应该优先采用比较稳健的策略。赛后有比较简短的总结：

按照一贯的方法，鬲融从 A 开始读，胡伟栋从 J 开始读，我准备编程的环境，然后从中间选择题目读。与上海赛区相比，比较顺利的是我一下就看到了一道比较简单的题目 B，于是我马上写了 B 题。

B：使用火柴棒拼接数字，给定 n, m ，求使用不超过 n 根火柴棒拼成的 m 的倍数中最大的数字。

本题主要的思想是动态规划，方程很容易得出，时间复杂度 $O(nm \cdot 10)$ 。但是本题有两个难点：

(1) 如何得到最大的数字？标准的方法是使用 BFS，搜索的过程中必须非常注意搜索的顺序，但是这样实现很容易出错。我注意到了 $n \leq 100$ 的条件，因此结果不会超过 50 位，于是我使用 3 个 int64 来保存结果，这样既实现简单又不容易错。虽然程序常数比较大，但是不至于导致超时。

(2)不能不使用火柴棒，而数字 0 也需要火柴棒来拼成。例如 $n=5$ ， $m=97$ 的结果是-1，而 $n=6$ ， $m=97$ 的结果是 0。

ACM 比赛的第一题的选择对比赛的进程影响很大，此次比赛很成功地使用了很稳妥的算法。

E：标准的课堂睡觉问题，求最早所有人不睡觉的时间。

简单的模拟题，而且时限很宽松。

E 其实是最简单题目，但是由于我的低级错误，不仅得到了一次罚时，而且浪费了宝贵的时间。好在当时比较冷静，很快改正了错误。

D：对于一个不超过 $100 \times 100 = 10000$ 的表达式，可以在其中加入 * 来表示任何字符，如果一个表达式只能和唯一的等式对应，则称为 A 类表达式。给定一个表达式 B，要求通过改变最少的字符使它变成 A 类表达式。

本题使用了一次搜索再检索的方法，可以有效控制程序的速度。估计时限问题后选择先提交节省了不少时间。

J：给定一个 4×4 的网格的边框图形，问是否可以通过在 4×4 的网格上放 6 个 2×2 的正方形框得到。

H：给定一组旋转后的乐谱及初始音符及结束音符，求原始的音符序列。

开始胡伟栋没有看到旋转的角度是整数的条件，于是推出了可以解决实数角度的数学公式。但是由于公式需要考虑的情况比较复杂，幸运的是我们使用了枚举角度再判断的方法。

写了读入部分之后，看到 Panacea 过了 H 题。昝融重新读了题目，发现了角度是整数的条件，于是稍微修改就得到了正确的程序。

本题需要判断的条件很多，需要考虑得很仔细：

- (1) 先把坐标按照 $X[i]$ 排序。
- (2) 通过初始音符及结束音符得到 sd ，判断 sd 是否在 1 到 5 之间。
- (3) 判断相邻两个字符的距离是否在 sd 到 $5sd$ 之间。
- (4) 判断每个点的坐标是否可以对应一个音符。

F：求 3 维空间的 Voronoi 图，输出每个 Voronoi 块体积占的比例。

胡伟栋提出了本题的正确算法：分割立方体。如果一个立方体的 8 个顶点都到一个点最近，那么这个矩形内的所有点都到这个点最近，否则就分割这个立方体，直到立方体的体积少到一定程度为止。

开始程序的精度不够，后来胡伟栋提出了一个启发式的分割立方体的方法，思想就是使得两个立方体的分割面以尽量大的概率穿过 Voronoi 平面。程序的效果很好，而且那时时限也已经放宽，于是顺利通过了 F 题。其实本题蒙特卡罗法也可以过，我们也想到了这个方法，但是由于随机过程写的效果有问题，导致精度不够。

通过 F 题的时间是 270 分钟，封版时除了 Panacea 通过 4 题，其他队伍最多只有 3 题。当时 Panacea 正好坐在我们背后，很可惜他们卡在了 D 和 G 上，最终也只有 4 题。最后半小时我们也没有尝试 H 或者 A，因为其他队伍很难在最后一小时通过 4 题。我们就简单尝试了几次 C 就默默地等待着西安赛区比赛的结束。

最终我们通过 6 题排名第一，由于已经获得了上海赛区的冠军，不参加 ICPC 的排名，不过仍然获得了 Solaris 杯。

总结：

西安的 ICPC 冠军是北京大学的 T2 (rainer, dzx 和 cici)，后来他们依靠西安的夺冠成绩进入了 2007 年的全球总决赛。T2 在最后一小时表现极其神勇，顺利通过 2 题一举超过了 GotoFly 和 Panacea，而且最后几分钟还很有希望通过 H，赛后和 rainer 讨论之后发现算法相差无几，可能就是一些小细节缺陷吧。2006-2007 年度的 ACM 比赛，从上海到西安，还有东京的总决赛，Mobile Robot 每次比赛都能看到 T2 熟悉的身影。我们在封版之后常常由于压力较大很难攻破题目，他们在封版之后的冷静心态非常值得我们的学习。

GotoFly 与 T2 一样通过 5 题，由于 5 分钟的罚时劣势位居第三，Panacea 通过 4 题依靠罚时优势排在第四。两队都欠缺一些运气，很可惜。另外印象很深的是，坐在对面的朝鲜队通过了 4 题在 ICPC 中排名第 2，如果西安能够多一个名额的话，他们就能够出现在总决赛赛场了。

西安赛区是 Srbga (刘汝佳) 出的题目，从高中时期就开始做了 Srbga 出的题目，西安赛区的题目又一次体现出了刘汝佳命题的许多重要特色：

- (1) 如果 Srbga 大哥出是一套题目，那么你会明显觉得题目拿在手上明显重一些。Srbga 大哥出题很重视题目描述或者故事的完整性，他很少使用一些僵硬的题目模型和描述。有时读着他出的题目更像是在读小说，欣赏故事。当然，由于 Srbga 大哥出的题目描述完整，刚拿到题目的时候会感觉很难上手。我们很容易发现没有一道题目容易上手。用数据表达，比赛开始时的空机时间偏长。
- (2) Srbga 大哥出的题目和世界总决赛的题目风格近似，题目多半对编程能力提出很高的要求，相比之下对算法的要求不是非常高，考察的都是比较基本的算法。如果用一个字形容就是：野。
- (3) Srbga 大哥出的题目对算法的考察范围非常广，虽然对于某特殊的算法要求不高。有时还需要很强的组合算法能力。
- (4) Srbga 大哥出的题目中很注意数据的设计，例如 C 题中特别生成了极端的情况，J 则使用了接近 20000 组数据。一般情况下，不经过精心设计的随机算法会吃尽苦头。

2006 年的 Mobile Robot 的 ACM 分区赛比赛任务已经全部结束了，2006 对于 Mobile Robot 来说是丰收的一年，我们圆梦了上海和西安的双冠军。在随后的冬天，我们加紧训练准备东京的总决赛。

利用假期空闲之时，将这几年 GCJ, ACM, TopCoder 参加的一些重要比赛作个回顾。回顾了 GCJ2006 和 2005 年的 ACM 之后，转向 TopCoder 的比赛吧。我参加的最早的 TopCoder 赛事是 TCCC2006。

TCCC2006——死亡之组

TopCoder Collegiate Challenge(简称 TCCC)是 TopCoder 一般在秋季举行的面向全世界在校学生的程序设计大赛，2006 年的 TCCC 在圣地亚哥举行。从北京到旧金山的飞行只需要 11 个小时左右，所以不至于那么疲劳。路上一切都很顺利，很感谢 OpenGL 的提醒，对于超过 8 个小时飞行自带拖鞋和枕头对我来说还是很重要的。

TCCC2006 使用的标准的 TopCoder 现场比赛形式，比赛有 48 名选手参加，首先 48 名选手被分为 16 个人一组，每组分别进行半决赛，前 2 名直接晋级决赛，3-6 名晋级 wildcard 比赛，wildcard 比赛 12 人中的前两名填补决赛的最后 2 个名额，决赛由 8 个选手参加。TopCoder 现场比赛中很重要的一个创新是：每名比赛选手在观众席前都有一个同步的显示器，这样观众可以看到选手任何时刻做的事情，极大增强了互动性。

TCCC2006 的 Room 1 和后面的 Final Round 都可谓是死亡之组。现在就回忆一下这两场激烈的比赛吧。

Room 1:

至于 3 个房间的分配，TopCoder 按照注册截止时选手的 Rating 分布蛇形分配。但是 TCCC2006 的房间实力分布极不平衡，我与上届冠军 tomek，著名选手 reid，Egor，halyavin 还有 Rating 不高但是实力极强的 Ying 和 ardiankp 同被分到了 Room 1，赛前 Room 1 成为公认的死亡之组。

在圣地亚哥，我和师兄 Macsy（张一飞）同一个房间，很感谢师兄的关心，我这几天休息的都很好。要知道如果同房的人有 10 小时左右的时差的话，一人必须很小心才能保证不影响另一人的休息。

Room 1 在我抵达美国的第二天早上进行，选手允许提前 30 分钟准备一些必要代码。不过现在大家都比较喜欢学习 Petr 那样一行代码都不打。下面就是比赛的过程：

250 分题目是：给定 $n(n \leq 50)$ 个整数 A_i 和一个阈值 d ，计算 n 个整数所有排列 P_i 中满足 $|A_{P_i} - A_{P_{i+1}}| \leq d$ 的排列中，所有不同可能 A_{P_1} 的个数。这题最标准的方法是动态规划，基本思想是把 n 个整数排序之后，计算两条相邻元素不超过 d 的序列。我使用了一种更精巧的算法，把 n 个整数排序之后，对于 A_i ，如果 A_i 可能作为排列的第一个元素，那么 A_i 必定在某一个方向（大小）连续而在另一个方向每间隔两个元素相连。这个算法比较容易实现，但是正确性证明比较难，甚至让人第一感觉是错的。我写完程序测试了所有样例都正确就提交了，243 分。提交之后我又测试了许多数据，并在纸上尝试证明正确性。

赛后，我看了网络上的讨论记录。在我提交 250 分题之后，立刻遭到了 misof 的怀疑，他认为我的算法有问题。据 Macsy 学长的回忆，OpenGL 在我屏幕前看我写完程序，也认为我的算法是错的，不过后来他们讨论之后发现没有反例。

关掉 250 分题目之后，我刚刚意识到 Room 1 的 3 题分数不是 250-500-1000，而是 250-600-900。现在看来，对于 250 比较顺利的情况，应该先做 500，若 250 不顺利或者想出奇制胜的话，可以先开 1000 分。当时没有什么经验，我认为 900 比 600 应该简单一些，于是就打开了 900。

900 分题目是：给定一张 $n(n \leq 10)$ 个点的带权有向完全图（也就是 n^2 个实数）和一个衰减系数 p ，求一条经过 $d(d \leq 10)$ 条边路径（不需要保证简单路径），要求这条路径的指数衰减长度（指数衰减是指第 i 段的长度乘以 p^{i-1} 然后求和）最接近 1000。这题如果使用穷举法，就需要 10^{10} 左右的计算量，在 TopCoder 的测试机上也不能通过，由于路径长度很容易超过 1000，所以很难找到多项式时间的动态规划。我马上有了个想法——双向搜索。对于长度为 d 的路径，其实可以看作从某一个点 p 出发的一条反向的长度 $\lceil d/2 \rceil$ 的路径和一条正向的 $\lfloor d/2 \rfloor$ 的路径，对于固定的节点 v 来说，这种两个方向的路径都不超过 $n^{\lceil d/2 \rceil}$ ，这样只要枚举一个方向的路径然后二分查找另一个方向即可。复杂度是 $O(dn^{2+\lceil d/2 \rceil})$ 。

现场比赛调试环境不是很好，我花了不少时间调试以发现程序中的错误。提交之后 690 多分，还不到 700。不过对于 900 分的题目在那种压力下还可以接受。提交之后我花了 15 分钟左右测试，没有发现错误。于是就准备做 600 了。

600 分题目是：一道经典的数学题，给定一些盘子叠放的规则，计算顶层盘子的最大可能大小。其实算法不是很难，只要二分顶层盘子的大小，然后依次贪心计算来判断底层是否能够满足即可。只是贪心的时候要考虑两种情况，一时想不清楚。我当时已经感觉很疲劳，思路不是很清楚，最后 40 分钟时间也没能调试通过。这题过于琐碎，Room 1 中最终没有选手通过 600 分题，并且成就了一个刺激的 Challenge 阶段。

Coding 阶段我和 tomek 采用了截然不同的策略，我跳过 600 直攻 900，而 tomek 在 600 中挣扎了很长时间才放弃。Coding 阶段结束时，有 4 名选手提交了 3 题。我依靠速度优势领先同样提交 250 和 900 的 tomek 35 分左右。

Challenge 阶段开始时，我盲 cha (blind challenge) 了一个最后时刻提交的 900 分程序，但是由于我选择的数据实在太弱，失去了 25 分。这样我和后面的 tomek 只相差 10 分左右了，所以我决定只要 tomek 不动，我也不动了。其实，当时 tomek 已经知道自己的 900 是错的，Challenge 阶段他估计已经放弃了。我的 Challenge 阶段最终就以 -25 分结束。

之后的 Challenge 就是 Ying（王颖）展现勇气 and 智慧的舞台了。他 Challenge 掉了所有提交的 600，凭借 225 分的加分超过了我，排在榜首。这样比赛的形式也一目了然了，7 位选手提交了 900，我依靠速度优势领先第四名 reid 超过 100 分。只要我两道题目能够 Passed System Test 就足以进入 Final Round 了。

System Test 之前，我和 Ying 讨论他“超神”的 Challenge 阶段。这是我第一次参加 TopCoder 的现场比赛，发现 System Test 结果显示是按照 System Test 之前的排名倒序进行的。测试到我时，除了 tomek 的 4 名选手的 900 都过了。显示我的结果时，两个绿框闪烁了很久终于显示出了两个大大的钩，我终于可以欢呼庆祝胜利了。我前面的 Ying 也两题全过了。这样我们两位中国选手得以在死亡之组携手出现，这场比赛真可谓是中国选手的胜利。Reid 只能在 Wildcard 赛再作努力，tomek 则被直接淘汰出局了。

Final Round:

接下来的两天里，我观看了 Room2, Room3 和 Wildcard 的比赛。第 2 天晚上我们参加了 TopCoder 赞助的 Laser Tag 游戏，我们所有中国人组成了一队，我的发挥很差，原因是这个游戏与 CS 不同，选手头上没有感光器，而我喜欢遇到人就攻击头部，所以狭路相逢多半是我失败。活动中，我有幸结识了许多 Dev 的神人，当时由于 vividmxx 没有参加，magicpig 和 PE 的竞争很激烈，最终 PE 获得了“浙江大学建校 100 年来第一个 TCCC 冠军”。记得赛后我 uncle 来到现场，我 uncle 是浙江大学本科毕业的，magicpig 见我 uncle 第一句话就是“浙江大学建校有 100 年历史了吧？”汗死了。另外 zjq 也获得了 Design 的亚军。

第三天中午 Championship Round 开始了。决赛时，场地里安装了很多摄像头，可以说我们的任何举动都在严密监视下了。这回我提前确认了题目分数是标准的 250-500-1000 的分布。参加决赛的选手除我之外有：andrewzta, ardiankp, bmerly, Eryx, mathijs, Petr 和 Ying。面对决赛选手的实力，我已经没有意义定一个类似于“保几争几”的目标了，努力发挥自己的水平是最应该做的。下面就是比赛的过程了：

250 分题目是：给定 n 个正三角形，每个顶点都有数字，选出 6 个三角形拼成一个正六边形，要求相邻的数字必须相同。三角形允许旋转，计算能够得到多少个本质不同的正六边形。题目很长，我仔细读了两遍才开始写，算法很清楚，就是枚举六边形中心和四周的 7 个数字，然后判断是否有足够的三角形。在判断本质不同的时候犯了一个错误，调试了几分钟，提交之后只有 215 分了，看了一下排名，Petr 有 232 分之高，其他选手都还没有提交。测试了几分钟发现程序的运行时间不是很稳健，很容易到达 0.8 秒左右，测试了 15 分钟之多才逐渐放心下来，因为基本上所有数据都 0.8 秒左右。赛后 Macsy 告诉我，我的程序速度瓶颈是在 set 的判断，所以时间比较稳定，不会超时。我当时的犹豫和没有经验浪费了至少 20 分钟的时间。

按照赛前的计划，我这时应该打开 1000 的题目，但是由于自己对 250 没有信心，而且求稳思想比较重，我先打开了 500 分的题目。现在看来，开 500 分的题目并不算错误，其实在打开 500 分题目的时候，与 Petr 的差距不是很大。

500 分题目是：给定一个机器人的移动命令序列，要求计算结束时机器人的位置。由于移动序列中允许()这样的重复操作，直接模拟是超时的。这类题目的标准算法是利用矩阵乘法，由于之前对于此类题目没有经验，没有准备好就开始写了，导致矩阵处理失败。我果断放弃了调试，换用一种记录中间结果的搜索方法，写完的时候已经只有 280 分了。更重要的是我已经没有时间进攻 1000 分了。提交之后排在第 3，前面是 Petr 和 Eryx。

1000 分题目是：给出一个排队取菜的模型，计算一个等待时间的排队序列。而且对于多种答案的情况，要求计算字典序最小的序列。题目其实不是很复杂，集合动态规划就可以解决，不过模拟取菜过程时需要非常注意细节。Petr 提交了一个 660 分左右的程序，Ying 则在最后一分钟提交了 400+ 分，排在第 2。

Challenge 阶段显得很枯燥无味，前两天大发神威的 Ying (+225) 和 Petr (+300) 都没有尝试 Challenge，整个 Challenge 阶段没有任何一个 Successful Challenge。

System Test 结果出来了，在 bmerly, ardiankp 和 andrewzta 都只通过一题的结果出来之后，排在我后面的 mathijs 两题都 Pass，随后我的 250 和 500 也都 Pass 了。但是，排名在我之前的 Eryx 和 Ying 的 500 分和 1000 分都 Failed System Test 了，我瞬间提升到了第二名的位置。不过虽然 Petr 的 1000 分挂了，但是他依旧凭借 250 和 500 的速度获得了冠军。

在这里说一下 1000 分的真实情况吧，因为这些时间来对于 TCCC2006 Final Round 的 1000 分题目有很多不同的说法。比赛结果中显示没有选手通过 1000 分题，如果仔细分析测试结果，Petr 的程序由于超时出错，而 Ying 的程序由于一个地方没有清 0 而导致错误，确实很可惜。因为如果 Ying 的 1000 能够 Pass 的话，他将是 TCCC 的冠军。不过 Ying 的算法犯了与造成 Petr 超时一样的错误，他们的动态规划程序比标准方法多出一个 n 倍的时间，我曾经成功生成了一个用例，可以让 Ying 和 Petr 的程序都超时，这个例子已经得到了 Ying 的认可。需要指出的是 TCCC2006 是 TopCoder 的测试机的速度还是很慢的，两个程序如果在现在的机子上运行可能只需要 1 秒左右了。

比赛之后和 uncle 到 downtown 游玩了一下，参加完颁奖晚会，第二天就回国了。

总结：

TCCC2006 是我第一次参加 TopCoder 的现场比赛，很有幸能够在这么多的第一次中就进入决赛并且获得第 2 名的成绩。很感谢同参加比赛的同学 Macsy, OpenGL, Ying 还有 PMH 的关心和帮助，你们在我比赛时全程在场边，让我感觉很温暖。

另外，我还有幸认识了 visualage，现在他已经是 arena 的负责人了吧。记得他和 OpenGL 在 Room 1 的 Challenge 阶段通过大声叫中文（在国外，这是最好的密码）告诉我 torek 的 900 是错的，可惜我没有听见。

TCCC2006 对于中国来说是不小的收获，中国选手占领了 Dev 比赛，PE 获得“浙江大学建校 100 年来第一个 TCCC 冠军”，magicpig 和 zjq 分获 Dev 和 Design 的亚军，也就是说中国包揽了所有亚军。在比赛之余，我很高兴认识了众多 TopCoder 的朋友。

Petr 在决赛中表现了非常良好的状态，TCCC 的夺冠标志着 Petr 收获了 2006 年的大满贯。Ying 也采用了很合理的策略，只可惜他的赌博由于运气差一些惜败。我采用了比较保守的策略，在所有决赛选手中排名第 2，这也是我在 TopCoder 的现场赛事中的最高名次了。

TCCC2006 我很感谢家人的关心，父母凌晨很早起床查看我的比赛结果，而 uncle 还特地赶来现场为我加油。这几年的 TopCoder 现场比赛的赞助商列表里都能找到 American Online (AOL) 的身影，TCCC2006 是 AOL 唯一一次进行了 3 个小时左右的全程直播，父母和 uncle 都在网络上观看了现场的影像直播。

TCCC2006 我神奇地保持了 100%的正确率，我个人认为 TopCoder 现场比赛对正确率提出了更高的要求，我们不必太在意 Coding 阶段的那些高分，只要自己的程序是正确的，就是成功的。

利用假期空闲之时，将这几年 GCJ, ACM, TopCoder 参加的一些重要比赛作个回顾。首先是 GCJ2006 的回忆。

Google Code Jam 2006

一波三折：

Google Code Jam 2006 是我第一次到美国参加现场的程序设计比赛。Google Code Jam 2006 的比赛地点设在了纽约，这次纽约之行之前的签证出了不小的问题，这里非常感谢大家对我们的关心，特别感谢吴总（wyy）和鲁小石的帮助，使我最终踏上纽约之旅。

从北京到纽约的飞行时间是 13 个半小时，由于是第一次做超过 8 小时的飞机，没有什么必要的经验和准备，路途非常疲劳。一到宾馆就睡了，结果由于手机铃声的时间使用的是东方时间，差了 12 个小时，一觉把所有事情连晚饭一起都睡过了，随便吃点东西就继续睡了。之后的所有现场比赛我都养成了提前睡觉的习惯，以保证充足的体力。

比赛过程：

比赛时精神状态还算可以，但是分配了比赛房间之后发现自己和 tomek 分在一个房间，真是很不爽；在和旁边的 zhuzeyuan 抱怨的时候，发现他和 Petr 一个房间，彼此彼此吧。

下面就是比赛过程了，总体来说比赛过程比想象的艰苦，不过其实在 System Test 之前的结果还是很满意的，先简单描述一下 3 道题目吧。

250 分的题目是一道平面极值问题，给定 n 个点，求一条直线，使得 n 个点到这条直线的 y 方向截距总和最小。我回忆起金凯在 2003 年集训队论文中报告中讲到的很类似的一道题目，记得一个重要结论是这条直线一定经过两个点，虽然题目有些不同，但是很快得到了相同的重要性质：这条直线一定经过两个点。这样很容易得到一个 $O(n^3)$ 的算法。

500 分的题目是一道反 Hash 函数问题，给定一个 Hash 函数和 x ，求一个最小的非负数 y 使得 $H(y)=x$ 。估计了一下，单向搜索需要 26^8 ，于是我改用双向搜索，这样就变成了 26^4 。但是实现过程比想象的复杂很多，提交了后只有 280 左右了。其实，这题有更简单的数学方法，tomek 的程序有 450+。

1000 分的题目是涉及卷积函数和计算反函数的问题，通过转化变成线性方程求解问题。当时受到现场气氛的影响有些紧张，浪费了不少时间，提交之后 550 分左右。其实，当时一些原理问题都没有想清楚，不过后来和 Ying（王颖）经过讨论验证都是正确的。

Coding 结束之前 Petr, tomek, Ying 和 andrewzta 都提交了 3 题，其中 Petr 领先得比较多，我和其余 3 人差距 50 分以内。

Challenge 阶段开始之后，我由于 500 分题自己使用的是双向搜索的算法，没有注意到有些单向的搜索加模线性方程的方法其实是正确，在 10 分钟以内 cha 错了 2 次。落后于上述的 4 个人，排在第五。

但是下面的 5 分钟发生了戏剧性的一幕，首先是 Petr 的 250 被 cha 了，接着 Ying 的 250 也被 cha 了，这样我面临这样一个情况：tomek 领先我 100+分，andrewzta 领先我 30+分，由于我和 tomek 处在一个房间，所以我做出了一个大胆的决定，就是 challenge tomek 的 1000 分题，我随机生成了一个随机大数据，在最后时刻提交了这个 challenge，系统返回了一个令人窒息的结果：successfully challenge。凭借这 50 分我一举超过了 tomek 和 andrewzta，在 System Test 之前占据了榜首的位置。

戏剧性的结果：

我很有幸能够在第一次参加现场比赛时，就能够和冠军这么接近，如果 System Test 能够全部 Pass 的话，这可以认为是一场完美的比赛。

可是，整个故事就好像是被刻意设计的一样，System Test 之后的结果使我目瞪口呆：首先是 250 分的题目，我由于有一个地方没有及时使用 double，而造成整数越界；然后，1000 分的题目简直是悲剧的最高境界，我在高斯消元的时候没有及时把一个重要变量暂存，导致影响了结果，没有想到竟然躲过了那么多大数据，但是不能通过 System Test。最后排在 50 名左右。这两个错误至今刻骨铭心。

最终 Petr 获得冠军，Ying 亚军，andrewzta 由于 500 挂了排在第 3。

11 月的纽约有些冷，我随大队人马一同去了一趟帝国大厦，景色很迷人。第二天休息一下后与几个中国选手打了一会“找朋友”，第一次美国之行就结束了。

总结：

比赛结果虽然不是很理想，但是对于第一次参加世界比赛的我还算可以接受。也算是为今后的比赛留下一些教训吧。

在帝国大厦上见识了大家的拍摄功底，我由于技术差没有拍到任何合适的照片。

在比赛过程中，首次见识了 liympanda 的大将风度，和 panda 在一起总是笑口常开，他无论遇到什么情况都无所畏惧，这一点我一直在努力学习，不过一直做的不好。但是 panda 打牌的时候就不一样了，总是喜欢偷看别人的牌，还炫耀自己会说广东话，被 Ying 和 rocking 两位广东选手狠狠鄙视了一番。

Petr 加上之前的 TCO 和之后的 TCCC，拿到了 2006 年的大满贯，可以算是历史性的突破吧。Tomek 有些可惜，比完了还问我怎么 cha 他 1000 分的，呵呵。

其实这次比赛 Ying 挺可惜的，其实 Petr 的发挥并不很好，如果 Ying 运气再好一些的话，历史从那时就要重写了。不过 Ying 还是体现了他超强的数学功底，让人佩服。另外，来自复旦的同省队友 LemonTree 也获得了好成绩。

这好像是自己最后一次和 xreborner 同场竞技了（由于之后 xreborner 退役了很长时间，忘记 GCJ2008 我们又见面了，谢谢 Savior 的提醒），感谢您在我高中时期教授了我许多编程技巧，我一直沿用至今。

附比赛排名：

Handle Score

Petr 927.02
Ying 811.21
andrewzta 761.56
halyavin 732.46
tomek 677.55
tomekkulczynski 634.69
pashka 590.05
asaveljevs 579.60
ainu7 552.23
misof 537.19
Egor 534.19
xOberon 517.65
reid 516.23
bmerry 498.01
LemonTree 497.27
MikeMirzayanov 490.32
PaulJefferys 481.89
monsoon 478.60
Andrew_Lazarev 475.39
tsjoker 454.86
kalinov 440.75
pparys 440.41
dyatlov 420.64
Michael_Levin 419.19
daveagp 403.39
malcin 399.55
kalmakka 397.15
kedaizd 391.91
cyfra 389.02
Macsy 387.38
Psyho 377.22
mhchan 374.90
jakubr 353.71
otherwise 330.73
kia 283.48
ACRush 282.62
JongMan 261.44
IvanRomanov 251.33
DmitryKorolev 235.14
Revenger 234.54
elizarov 233.79
evgeni 232.44
antimatter 226.70
CatalinT 226.15

Jan_Kuipers 225.73
krijgertje 225.11
wd.h 221.28
w_ 218.82
zhuzeyuan 217.09
falagar 212.75
MegaS 210.88
gawry 209.13
liympanda 207.74
hyyylr 206.37
skatou 205.00
Vintik 204.09
jasonw 199.82
darnley 199.59
NPermyakov 199.54
aengus 196.67
embe 191.62
Yarin 186.89
NSI 185.22
AdrianKuegel 182.40
nicka81 181.01
HeaDacHe 178.77
VitalyGoldstein 178.37
kappa 175.40
HilbertRaum 168.29
DmitryKlenov 168.26
Abednego 165.51
Rocking 164.38
Per 163.41
Emilian_Miron 158.22
Aidin.Kashigar 156.94
lukasP 156.00
grotmol 152.34
gevak 142.55
nhzp339 130.61
NauCoder 107.71
lazyboy 98.73
WSX 56.89
Snail 25.00
Masao 0.00
blackmath 0.00
.Invader 0.00
Mg9H 0.00
smsorin 0.00

Rostislav 0.00
nya 0.00
lyc1977 0.00
xreborner 0.00
goo 0.00
soul-net -25.00
wintokk -25.00
Ulan -25.00
Bankevich -25.00
madking -25.00
fpmc -25.00
Soultaker -25.00

利用假期空闲之时，将这几年 GCJ，ACM，TopCoder 参加的一些重要比赛作个回顾。今天到了 2007 年初的东京，回顾一下 2007 世界总决赛发生的趣事吧。

ACM-ICPC World Final 2007——Mobile Robot 东京决战

2007 年的东京 ACM-ICPC 全球总决赛在樱花盛开的 3 月初拉开序幕。成立了一年的 Mobile Robot 凭借 2006 年 ACM 上海赛区的冠军，代表清华参加了此次 ACM 盛会。

记得黄金雄教授在杭州 2008 时说，ACM 总决赛的实力分布由原先的美洲独霸逐渐转向了现在的亚欧争霸。2007 年，同样来自亚洲的上海交大具有很强的夺冠实力，欧洲 2007 年虽然没有顶尖高手 Petr 和 tomek 的参与，但是 ACM 传统名校 St. Petersburg, St. Petersburg IMFO, Warsaw, Saratov, Petrozavodsk 等都派出了极其豪华的阵容。虽然在 2000 年前后美洲队伍成绩不佳，但是近些年由于众多欧洲选手的加盟，美洲 MIT 等顶尖名校也在总决赛中表现得非常强势。

记得，每次世界总决赛之前，TopCoder 的论坛上都会罗列出所有参加总决赛的 TopCoder 选手名单。但是我不是很看重这些数据，因为在很多次与欧洲选手切磋之后，我发现了自己与欧洲选手相比的一个重大缺陷：我参加各类赛事以来，起初比赛过程中常常受压力的影响很大，很难正常发挥自己的水平。后来情况有所好转，在大多数比赛中都能正常发挥自己的水平。可是，令我感到意外的是，许多来自西方的选手在巨大的压力下，反而表现得极其兴奋而能超常发挥出自己的水平。来自西方的各队，我相信他们只要达到了兴奋的状态，都拥有获得冠军的实力。去年上海交大总决赛总结中，他们也提到了自己没有发挥出应有的水平，而 IMFO 即使在比赛压力下仍然能够做出 8 题，可见他们平时训练实力之强。但是我觉得现场比赛发挥受影响可能是少数中国选手的坏习惯，可能不适合用同样的思路分析欧洲的顶尖高手。

抵达东京：

出发的前一天晚上，我仍然熬夜参加了 TopCoder 上的 SRM 比赛，竟然是 Petr 出的题目。当时我与 Petr 的 Rating 差距很小，当时我 3 道题目都交出了很高的分数，在 System Tests 之前遥遥领先，但是 500 和 1000 分的题目都由于一些很小的粗心而失败了。我也失去了在总决赛之前超过 Petr 的大好机会。结果到达日本之后的第二天，吃早餐的时候，我就碰到了作为教练来到东京的 Petr，他一看到我就扯前天比赛的事情，汗。现在回想起来，那场 SRM 对我的总决赛之旅确实有不小的负面影响。

抵达东京之后才发现，所有队伍中，只有我们选择了与所有志愿者衣服颜色相同的清华校色紫色，开幕式过程中，许多队伍都把我们当成志愿者了。

练习赛前一天的晚会很丰盛，大多食物都是中国风格的，水果也非常好吃。晚会期间，我见到了众多大陆学校的队伍，当年大陆至少有 15 支队伍参加总决赛，随处可以感觉到说着国语的选手。同时还见到了许多 TCCC 上出现过的面孔，随后发现 ardiankp 也来参加了，我们还聊起了 ACM 在新加坡（ardiankp 是代表南洋理工大学参加的）的情况。类似总决赛这样的比赛，我觉得选手之间的交流则更重要

了，因为每次总决赛都会集结众多熟悉的 ID 但陌生的面孔。晚一些之后，我们与北京大学的 T2 一起打牌，队友 geworm 和 wd.h 都抽签到了另一方，他们的牌太猛了，在加上我和李文新老师的牌都不好，结果我们惨败。

从正式比赛的前一天的中午开始，主办方组织我们游玩当地的 Disney 乐园。日本 3 月的景色很美，当地人也很热情，唯一的缺点就是无论用日语还是日式英语都很难交流。我们在 Disney 乐园中主要以观看表演为主，没有参与过多的活动。东京到了晚上有些冷，我嘴唇都有些结冰了，可是发现路上许多日本女高中生还穿着裙子，仰慕。

正式比赛：

总决赛的队伍是按照学校的音序排座位的，练习赛时我们发现自己坐在来自荷兰的上届亚军 Twente 大学旁边，刚打招呼就发现他们 3 人的最低身高也有 190，据说荷兰女子的平均身高也有 180 以上，似乎觉得自己是从小人国来的。

练习赛过程中，我已经丝毫感受不到娱乐的气息了，现场的紧张气氛已经笼罩了我们全队。所有队伍都在抓紧一分一秒熟悉比赛环境，赛场中敲击键盘的声音已经完全覆盖了观众鼓掌的声音。比赛中使用的 PC² 提交系统比想象得稳定，我们努力尝试各种功能以熟悉机子上的编程环境。东京的总决赛使用了一个形状奇特的键盘，由于当时早已养成了自带键盘的习惯，这次总决赛中奇形怪状的键盘对我编程的速度影响非常大。

总决赛正式比赛在第二天 9 点左右开始，Bill 想尽各种办法活跃气氛，不过比赛开始前几分钟现场还是静得可怕，比赛开始 5 分钟之后，现场就被键盘声笼罩直到结束。我们回顾一下比赛的过程吧，底纹的文字是我比赛后写下的总结：

这次 World Final 的题目又基本由编程题组成，可能是由于比赛时不够兴奋，比赛全程都非常不顺利。

大概从 2003 年开始，世界总决赛的题目风格已经完全倒向以编程题为主的特点，对此我们早有准备。不过由于时差问题，还有几天前 SRM 比赛由于错两题导致 Rating 跌停对我信心的影响，使我比赛中一直不是很兴奋。不过比赛过程中，我们仍然坚定的采用前面提到过的常用组队模式：

- (1) geworm 全程负责读题，思考算法和出数据；
- (2) wd.h 和我在比赛前 2 个小时一起攻简单的题目；
- (3) 2 小时后 wd.h 就开始死磕难题，我主写程序一直到 3 个半小时左右，结合 wd.h 对难题的把握，大家开始合攻难题。



25 分钟: Problem A, 简单地枚举。可是我生物没有学好, 没有考虑父母基因的顺序问题, 错了一次。

比赛开始时, 正常情况我会从 B-I 中间寻找容易上手的题目。可是由于有些紧张, 直到 geworm 给我翻译 A 题目内容时, 我还没有读懂任何题目, 这种情况很少发生。

题目 A 的描述, 需要一些必要的生物知识帮助理解, 可是这些东西我早已忘记。geworm 花了不少时间帮助我理解这题, 我还是由于没有考虑父母基因的顺序 WA 了一次。不过改过来之后, 我们竟然是所有队伍中第一个通过 A 题的, 可见当时很多队伍也没有完全放开。

43 分钟: Problem B, 最长上升子序列。开始算法没有想好, 莫名其妙地错了一次。

如果说题 A 的 WA 是生物问题, 那 B 的 WA 简直就是莫名其妙。B 就是最长上升子序列问题, 好像刚开始写时我和 wd.h 都没有想清楚, 写了一个神鬼莫测的程序, WA 一次之后才改成正确算法。可是当时我们都没有想到的, 总决赛中我们队伍莫名其妙的 WA 噩梦才刚刚开始。

97 分钟: Problem G, 枚举+模拟。这是很扯淡的一题, 题目很容易看错, 我们由于看错题目错了两次, 等看到 Twente 大学过了之后才重读题目, 找到了正确的理解, 浪费了大量的时间。

G 的题目描述确实不是很清楚, 许多队伍都发生了理解错误, 我们也不例外。不过第 2 次提交错误就不能理解了, 当时也不知道出于什么原因又提交了第二次, 难道是想先抢一个提交冠军吗? 当时我们确实受到了开局不顺利的影响, 这样做在罚时本身就落后的情况更是下雪上加霜。

146 分钟: Problem F, BFS。其实这题是我发挥编程能力的机会, 但是我开始用了个很奇怪的搜索方法, 错了一次才改用 BFS 过了。

在 G 题迷茫而放弃之后, 我又尝试实现了 F。F 的第一次 WA 是我们 Final 之行的第三次“莫名其妙”了, 我也不知道自己用了什么一种奇怪的搜索方法竟然过了样例, 还马上提交了, 面对这种情况我有些着急, 表现得很不冷静。好在 geworm

及时提醒，我马上改成 BFS 过了。在这期间，wd.h 已经实现出了 I 题，并提交了一次，结果是 WA。

178 分钟：Problem C，排序+枚举。这题有一个阴险的地方，就是 $\theta=0$ 的情况，还好我们考虑到了，这也是我们唯一一次 AC 的题目了。

C 题的算法其实非常清楚，阴险的情况我们也考虑到了，我终于没有再搞笑一次，这也是我们唯一一次 AC 的题目了。从通过 C 的时刻讲，我们的形式还是很有利的，因为难度很大的 I 我们已经实现得差不多了。

224 分钟：Problem D，数学题。这题本是一道很简单的数学题目，但是不知出题人怎么想的，搞了一些没有任何意义的东西，真是这次题目的一大败笔。我们开始由于没有注意三点共线的情况错了 3-4 次，然后由于 int64 越界又错了 3-4 次，最后错了 7 次才 AC。这题一共浪费了 1 个多小时。

在 BGF 各一次奇怪的 WA 之后，我们又完全陷在了 D 题的陷阱之中，如果顺利的话 D 题只需要 15 分钟就可以写完，可是我们忘记考虑了 D 题中很多的阴险情况，拖延了 1 个多小时，贡献了 7 个莫名其妙的 WA。可是，当时我并没有想到，这已经是我 AC 的最后一道题目了。

227 分钟：Problem I，数学+模拟。这题是 Jelly 写的，有很多特殊情况。

平心而论，我在总决赛上的状态不是很好，编程速度受到影响，而且有 10 次以上的错误提交。最后我们 7 题的罚时高达 1200 多，而上海赛区同样 7 题的罚时只有 700 多，从这一点上也可以看出当时实在不在状态。不过，wd.h 很好地执行了我们预定的组队模式，顺利完成了拖后中卫的角色。在我通过 D 题之后，他改正了 I 程序中的最后一个 bug。I 题最终也只有我们和华沙两支队伍通过，可是说是我们最终能够获得亚军的杀手锏。记得在颁奖仪式之前，基本上所有选手见到我都问 I 怎么做，我都统一回答：是胡伟栋做的。

我们依靠 I 题的 AC 首次排在了榜首。比赛进行了 227 分钟，能够在 200 分钟之后获得领跑的机会，我首次看到了夺冠的希望，上海和西安赛区的欢呼场面一次又一次从我眼前闪过。当时只有华沙大学通过 6 题，其他队伍都还不超过 5 题。

可是幸福只持续了短暂的 3 分钟，我们由于罚时太多而被华沙反超，华沙大学通过第 7 题时华沙队员的反应几乎疯狂，ICPC 的工作人员也用照片记录了这一时刻。

Problem E，我们的算法应该是正确的：二分答案+最短路。但是不知程序犯了什么错误，没有 AC。

Problem H，很复杂的几何题目，我们的算法是：扫描。但是不知程序又哪里写错了，结果是 WA，不是 TLE。

虽然在接下来的 73 分钟时间内我们没有再过题，不过我们仍然拚杀到了最后一刻，拼尽全力而无怨无悔。无论是 E 还是 H，我们都想出了正确的算法，并且成功写完了程序，但是 Judge 给出的结果一直是 WA。我们不断测试数据，并修正了一些 bug，但仍然不能通过第 8 题。在这种情况下的稳定过题能力我们确实特别没有训练过，华沙能够通过 8 题的超强实力确实很让人敬佩。比赛刚结束时，Petr 还特地赶来问我们有没有通过第 8 题，ICPC 的工作人员碰巧留下了照片。



当时我很希望能够借他的运气得到一个 Yes，不过 PC^2 还是不断返回 WA 直到最后。

后来，E 题就成了我写计算几何题目的一个巨大的心理障碍，直到 2 个月前在 Proxima 的一次训练中，在队友的支持下，我终于成功通过了一个更强版本的 E 题（题目在 UVA 上，题号是 11425，这题至今 2009.1 也还只有我和东京冠军队的 marek 通过）。

Problem J，这是一道很复杂的算法题目，现在我还不能证明算法的正确性。更重要的是这题很容易实现一些看似正确的算法，可能没有做这题是我们这次比赛的唯一成功之处。

I 的算法大致如下：

```
(1)  $X_i$  = the mininum cut between  $V_i$  and  $V_0$ .  
(2) while (the graph is not empty)  
{  
  (3)  $m = \min(X_i)$ .  
  (4) remove all nodes  $V_i$  whose  $X_i=m$ .  
  (5) let  $X_i = \min(X_i, m + \text{the mininum cut between } V_i \text{ and } V_0)$ .  
}  
(6) return  $X_1$ .
```

这里提一个公开的秘密，最后显示华沙大学的结果时，他们成功通过了 E 题，可是比赛过程中，我们并没有看到他们挂起蓝色的气球，不知道来自浙江大学或者中山大学的选手能不能仔细回忆一下，当时你们应该坐在他们旁边。

颁奖：

最终，华沙大学以通过 8 题的成绩获得冠军，Mobile Robot 通过 7 题总用时 1200 分钟获得亚军。整场比赛，我们克服了开局的种种不利因素，成为全场第一支通过 7 题的队伍，亚军也是一个非常可喜的成绩了。由于华沙大学不来自亚洲，我们同时也获得了亚洲冠军。



颁奖仪式之后的表演很精彩，印象最深的要数那位“神偷”了，他在观众面前不断施展“妙手空空”，观众掌声不断。记得表演结束后大家等电梯时，那位演员从我们身边走过，我们都连忙确认自己的钱包和手机。ACM-ICPC 东京总决赛在一片片掌声中落下帷幕。

总结：

ACM-ICPC 总决赛结束后，Mobile Robot 又恢复了平静。Mobile Robot 成立以来共获得了两个分区赛冠军和一个总决赛亚军，从那之后 Mobile Robot 就宣布解散了，也许唯一的遗憾就是没能获得一个真正的世界冠军。赛后，黄金雄教授也来向我们祝贺，从他的言语中，我们也感受到了一丝挥之不去的遗憾。

东京总决赛的几天里，我有机会结识了许多国内外朋友，也是这次日本之行的一大收获。同时也感谢众多 ACM 选手一年来对我们的关心和支持，当时 bbs.pku 上留下了一个很长的帖子，让我永生难忘。

在现场比赛中，我数次与欧洲选手直接交手，对他们的特点有一定的了解：

- (1) 欧洲选手的编程能力很强，很适应总决赛现有的题目风格。有些欧洲选手在 notepad 里写程序，然后直接提交的事迹绝非传说。
- (2) 欧洲选手对于算法的灵活运用能力强，但是对于一些比较深的算法了解不多。例如此次总决赛的 J 题。
- (3) 许多欧洲选手的现场抗压能力很强，即使在最后时刻仍然可以发挥出自己的水平。

在总结过复旦和 Srbga 出题的风格之后，总结一下我理解的总决赛题目风格吧：

- (1) Srbga 大哥出的题目和世界总决赛的题目风格近似，题目对编程能力提出了极高的要求。相比之下大多数题目对算法的要求不高。
- (2) 总决赛题目对算法的考察范围非常广，但是对于某特殊的算法要求不高。
- (3) 总决赛题目的时间限制很宽，出题人很提倡一题多解。而且数据没有想象得苛刻，随机算法有用武之地。

东京的总决赛已经结束快 2 年，今年寒假结束之后，我又要准备踏上总决赛征程了，希望这次我们 Proxima 能做的更好，将总决赛名次提高一位。

附 Final2007 排名

Rank Name Solved Time

- 1 Warsaw University 8 1405
- 2 Tsinghua University 7 1200
- 3 St. Petersburg University of IT, Mechanics and Optics 6 866
- 4 Massachusetts Institute of Technology 6 866
- 5 Novosibirsk State University 6 868
- 6 Saratov State University 6 957
- 7 Twente University 6 1011
- 8 Shanghai Jiao Tong University 6 1026
- 9 University of Waterloo 6 1103
- 10 Moscow State University 6 1192
- 11 University of Auckland 6 1210
- 12 California Institute of Technology 6 1241

利用假期空闲之时，将这几年 GCJ，ACM，TopCoder 参加的一些重要比赛作个回顾。最后是 2008 年的杭州复出。

2008 年 ACM-ICPC——杭州复出

2006 年 ACM-ICPC 总决赛结束后，Mobile Robot 就宣布解散了，也许唯一的遗憾就是没能获得一个真正的世界冠军。宣布退役 ACM 之后，我并没有完全与 ACM 绝缘，每次 TopCoder 大赛之前还常常做一些 ACM 比赛调整状态。记得 08 年初，我也全程观看了总决赛，不过没有想过复出。

杭州复出：

一切事情要从一个 zhuzeyuan 的电话说起，时间是 11 月 8 日晚上 10 点左右，当时我正在参加 UVA 在线比赛而为 GCJ2008 作准备。zhuzeyuan 在电话里首先告知我 Loner 车祸的事情，好在现在 Loner 已经痊愈了，当时确实很担心。随后，zhuzeyuan 向我介绍了 2008 年 ACM 比赛的进行情况，当时北京和哈尔滨赛区已经结束。然后，邀请我加入 Proxima 参加杭州赛区的比赛。我想当时答应的原因主要有 3 个：

(1) 我个人很喜欢 Coding，虽然退出 ACM 已经快两年了，但是还经常参加个人比赛。刚刚结束的 GCJ2008 中国区半决赛，出人意料的夺冠增强了我的信心。另外，ACM 这样长达 5 个小时的团队比赛造就了很特别的环境，赛场上的气氛和激情是做裁判教练或者参加个人比赛中无法体会到的。

(2) 3 年前的 2005 ACM 杭州赛区，我留下了我大学生生活中的一大遗憾。对于杭州 2005 的惨败，我一直想寻找机会从那个跌倒的地方爬起来，彻底摆脱紫金港校区留下的阴影。

(3) 其实还有一个原因就是我家在杭州，而且在本科期间我也曾经到杭州电子科技大学做过关于 ACM 的报告，Icy 老师的热情给我留下了深刻的印象。

对于 Loner 的车祸，我也觉得非常意外。这也是对于我们常年在校园骑自行车里横冲直撞的警示。Loner 现在能够恢复得这么好，我们都很高兴，祝你明年 ACM 好运。

加入 Proxima 的手续很顺利，教练邬老师对我复出想法的回答简单扼要：研一学生可以参加 ACM 比赛。

Proxima 的另外两名队友分别是 zhuzeyuan 和 zhouyuan（周源），我加入 Proxima 之后，新 Proxima 先后进行了 3 次训练比赛，随后就出发到杭州电子科技大学参加 2008 年 ACM 杭州赛区的比赛了。

当时，我通过许多网上资料和 zhuzeyuan 的描述了解了当时清华的战绩。到杭州赛区之前，清华的 What's Up 和 IronGods 已经分别获得了哈尔滨和北京赛区的冠军。其中 IronGods 还获得了哈尔滨赛区的亚军，What's Up 则一起来到杭州参加比赛。Proxima 在杭州赛区之前已经参加了北京赛区的比赛，成绩是第二名。就当时的形势讲，我们没有资格考虑太多事情，如果想保留悬念就必须获得杭州赛区的冠军。

杭州赛区现场赛：

在杭州赛区练习赛那天的上午，我们抓紧一切时间进行了模拟训练，选择的题目是 NEERC 的题目。题目难度有些大，我们做满整整 5 小时，直到 12 点 50 才急忙去吃午饭。结果很晚才到达比赛场地，到时候练习赛已经开始很久了。希望我们的迟到没有影响旁边队熟悉比赛环境。

杭电赛场的环境很好，在赛场里我找回了 2006 年上海赛区的感觉。队伍之间的空间很宽敞，电脑桌也很大，足以让 3 个人在上面一起推导公式。马上就见到了 icy 老师，不过他带来了一个不太好的消息——不允许自带键盘。好在杭电提供的键盘很标准，对我们影响不大。

正式比赛在第二天早上 9 点开始，回顾一下比赛的过程吧：

在 Proxima 队中，比赛开始时，仍然由我准备编程环境，然后从中间开始读题。我马上发现了 D 是一道看似简单的题目，并且也注意到了这句话：

WARNING: a naive algorithm might not be sufficient to solve this problem.

但是没有想到的是 BFS 算法也算是 naive algorithm，我交出了全场第一个提交，结果是理所当然的 TLE。不过那句 WARNING 稍微有些飘逸。

zhuzeyuan 发现 A 是简单题目，于是我马上写 A。

19 分钟，A：判断两张图的修改距离。枚举全排列，统计即可。

A 是最简单的题目，由于开始 D 的耽搁，我们大概是全场第 4 个出题的队伍。

接着，zhouyuan 发现 J 也很简单，于是我转向 J。

28 分钟，J：允许删点的并查集问题。通过添加新点的方法实现删点。

过了 J 之后，排名暂时上升到第一位。随后，zhuzeyuan 发现没有新题可写，于是就开始写 C，过程中，我和 zhouyuan 发现 G 比较简单，于是插空写 G。

50 分钟，G：简单图论问题。开始删点判断错误造成 WA 了一次。

59 分钟，C：高精度计算和素数判定问题。这题是 zhuzeyuan 写的。

不到一个小时就通过了 4 题，Proxima 获得了一个很好的开局。对于杭州赛区难度的题目，能够在第一个小时通过 4 题已经很顺利了。对于许多分区赛中会出现更多的简单题目的情况，有时能够做到一小时 5 题。但是一小时 6 题实在太难了，记得我们在一次训练比赛中做到了一小时 6 题，已经是我们的能力极限了。

接下来我实现了一下 B，可是由于发生了理解错误，计算结果与题目要求计算的结果直接存在重复排列问题，只好把程序放在一边。

随后，zhuzeyuan 开始实现 H，提交之后我开始写 F。

95 分钟，H：计算几何，如果使用 $O(n^2)$ 的算法需要注意常数不易太大。

105 分钟，F：自动机判断相等问题，通过计算差乘的方法能够在 $O(n^2 * |\Sigma|)$ 内解决

H 的提交等了很久，H 的 Yes 出来后不久我就写完了 F，提交之后也 Yes 了。大概在 2 个小时左右我们做出了 6 题，其实如果不在 B 上浪费时间能够更早一些。在 2008 杭州赛区，我们又一次获得了 6-4 的领先优势。

下面我们面临一个比较困难的状况，E 和 I 看似都比较复杂，但明白题意的 B 和 D 都没有想出算法。2008 年杭州赛区的题目中，基本没有中等难度的题目，所以我们通过 6 题之后就直接进入了比赛后期。当时我们分了一下工，我决定死磕 D 题，zhouyuan 负责推 B 题的公式。zhuzeyuan 尝试新题目 E 或者 I。

我的工作进行很不顺利，先实现了一个普通的 A* 算法，由于优化得不好还是 TLE。现在回想起来，D 题标准 A* 算法中使用的那个优化还是挺巧妙的，至少很有艺术感。我放弃 A* 算法之后，zhouyuan 似乎已经推好了 B 题的公式，开始帮助我实现 D 题。

163 分钟，D：状态最短路径问题，通过 A* 算法加一些优化可以轻松通过。

zhouyuan 提出了一个很重要的优化方法，先通过解方程的方法判断是否有解，在确认有解的情况下使用双向广度优先搜索，程序写好之后又 TLE 了。不过我觉得运行时间已经差不多了。于是，我使用了卡节点的方法，终于在第 5 次提交通过了 D。D 题我们用了大概一个小时左右。这时 What's Up 早已通过 5 题，不过由于他们卡在 H 题上，我们仍然以 7-5 领先。

zhuzeyuan 确认 E 和 I 比较复杂之后，我们开始合攻 B 题。zhouyuan 其实受到了我原先错误算法的误导，他得到一些公式来计算繁衍函数，通过繁衍以及原先程序的结果得到正确结果。不过，从当时的形式看，这样也是很不错的选择。

程序很快就写好了，提交之后又是奇怪的 TLE。B 题的 TLE 和 D 的 TLE 本质完全不同，B 题我们算法的复杂度是 $O(n^4)$ 的，对于 $n \leq 20$ 的数据范围，时间上应该没有问题。于是，我生成了 100 组测试数据，发现总共只需要 1 秒左右。

在 B 题的这一点上，我觉得命题人做的很不合理，虽然此题存在 $O(n^3)$ 的算法，但是既然把范围出到 20，就应该允许 $O(n^4)$ 的算法通过。可是命题人一共叠出了 6000 组测试数据，使得我们的程序超时了。而且在 Clarify 中的回答是 1000 多组，我们优化程序之后还是一直 TLE，当时我们怎么会想到是 6000 多。至少这里的范围 20 极具误导性。幸好，zhuzeyuan 及时想出了一个解决方法——打表。由于对程序没有信心，打表的 15 分钟时间内我们 3 人都只得通过手工计算简单数据来确认程序的正确性。

236 分钟，B：比较复杂的动态规划，需要考虑 4 种情况。

打完表之后提交终于得到了第 8 个 Yes，时间是 236 分钟，距离封版只有 4 分钟。由于 6000 组的阴险数据，我们从第一次提交 B 题到通过 B 整整用了 50 分钟，而且是 3 个人一直在一起做。

封版时，我们仍保持了 8-6 的领先优势。但是接下来，我们犯下了杭州 2008 最大的错误，如果类似的错误在总决赛中出现，我们将很可能失去领先地位。当时我们没有看到港大挂起 E 的气球，于是在 E 和 I 中选择了 I，结果深深地陷在了 I 的无底洞中，直到结束都不能自拔。

I：模拟题，需要考虑的情况比较多。

E：计算几何。计算半平面的交。

现在回想起来，E 题的难度远没有 I 题大，我们错误估计了 I 的难度。非常敬佩赛场上通过 E 题的港大和 I 题的湖南大学，你们不愧为射雕英雄。

清华 2008 战况：

2008 年，清华延续自己在 ACM 大陆赛区中的霸主地位，4 支不同的队伍获得了创纪录的 4 个不同赛区的冠军。分别是：

1. What's Up——哈尔滨赛区冠军
2. IronGods——北京赛区冠军
3. Proxima——杭州赛区冠军
4. ZCS——成都赛区冠军

从 ACM 的规则上讲 4 支队伍都获得了进军总决赛的资格，清华总决赛队伍的选拔过程在成都赛区结束的第二天就开始了。

从我的角度描述另 3 支队伍的情况吧：

What's Up 是清华第一个获得冠军的队伍，杭州赛区的过程中，他们以 amber 主写程序的模式进行，在比赛开始阶段体现出了很强的冲击力，不过卡住 H 后的慌乱略显出组队模式的缺陷。虽然他们在杭州赛区之后就选择放弃了总决赛资格的争夺，但是我们都深知他们的实力。后来 What's Up 的成员担任了 PK 比赛的裁判工作。

ZCS 由刚进入清华学习的三名大一学生组成，成员是 yuhch123, Cheryl 和 ScaleRhyme。我参加 Proxima 之后没有和 ZCS 交过手，不过在 Ural 和 SGU 上比赛时看到过 ZCS 的身影。在杭州赛区之后，ZCS 在成都赛区创造了 7/7(7 提交 7 通过)奇迹，不过和北京赛区相似的是后期略显经验不足。随后，ZCS 没有参加校内 PK 赛。

IronGods 的组成是 OpenGL, ahyangyi 和 ghy。在 IronGods 成立之初，我一直很看好这支队伍。哈尔滨赛区结束后，记得 ahyangyi 还来和我抱怨比赛中的失误，那道高精度题目确实有些过于复杂（呵呵，不过至少数据没有错误）。北京赛区的情况，我是事后听 dzx 介绍的，IronGods 依靠最后一小时的稳定发挥，通过 3 题，一举压倒 Proxima, Carriage 和 ZCS 获得冠军。可是几天后，我惊奇的发现自己需要面对强大的 IronGods 了。

IronGods 的组合与新 Proxima 惊人得相似，IronGods 的 OpenGL 与 ahyangyi 还有我和 zhuzeyuan 都是 TopCoder 上的 Target（中国一共有 7 个 Target，另外 3 个是前辈 haha, lympanda 和 ZCS 队中的 yuhch123，看好 zhoujie 成为第 8 个，加油呀！），他们的编程能力与我和 zhuzeyuan 不相上下。从 TopCoder 的成绩上看，我们两人的速度略快。

另一名队员 ghy 和 zhouyuan 都很擅长思考算法，ghy 结束 OI 时间比较短，状态保持得很好，zhouyuan 对于深入的算法了解比较扎实（北京的 A 很赞呀！）。

从配合上说，IronGods 组队时间长，配合方面比我们默契许多。我们重组后虽然也进行了一些训练，不过在比赛中普遍交流偏少，特别是我和 zhouyuan 的交流，在后几场比赛中才有些成功的配合。

不过从稳定性角度看，我们稍占上风，TopCoder 上的 Volatility 值至少可以说明一些。而且 ACM 比赛时间长达 5 小时，稳定性的要求应该比 TopCoder 还高一些。

清华校内 PK:

后来, zhuzeyuan 代表 Proxima 与 IronGods 协商之后, 大家决定采用三局两胜的赛制, 并定下了 3 场比赛的时间和题目安排。

关于总决赛队伍的选拔, 我个人非常不赞成直接指定, 可能与我的一些经历有关吧。已经进入研究生学习的我, 对参加总决赛已经没有两三年前的激情了。不过我个人的观点是, 如果学校指定, 我对于 4 种结果都可以接受; 如果进行 PK 选拔, 赛场上我一定拼尽全力。

两场 PK 过程中, 我们都在 [bbs.pku](http://bbs.pku.edu.cn) 上发布了现场的即时排名情况。由于清华 ACM 团队有严格规定要求对两次 PK 中使用的题目保密, 我这里就只留下了比赛的大致过程。

第一场 PK, 时间和吉隆坡赛区完全相同, 过程大致如下:

Proxima 启动比较快, 到 2 小时左右就获得了 5:2 的领先优势。

题 F 是这场比赛中我们最大的失误, F 浪费了很多时间, 而且最后都没有过。

IronGods 利用 Proxima 卡住 F 的时机连追 4 题, 以 6:5 反超。

发现 IronGods 反超之后, 我又尝试了几次 F 题, 但还是不能通过。比赛还有 70 分钟结束, 而且我们手上并没有其他题目。zhuzeyuan 在关键时候毅然决定开始写 J, 记得他说的一句话是“没有时间了, 我必须开始写了”, 当时形势不容乐观。好在 J 成功 1Y, 士气大振。

Proxima 随后连过两题重新占据 7:6 优势。

最后, IronGods 追成 7:7 平, 比赛又打得难解难分。

IronGods 最后时刻也还有机会, 我们又一次目睹了 IronGods 的绝地反击实力, 可能他们最后做 H 的选择值得商榷。

第一场 PK 过程中两支队都有明显失误的时期, 我们由于失误在中期, 所以罚时较少。最后依靠罚时险胜, 在 PK 中占得先机。

第二场 PK, 时间设在 12 月 25 日的晚上进行, 题目编号从 A 到 L, 共有 12 题之多。第二场 PK 比前一场进行得更激烈, 过程中两支队伍都长时间保持了很好的状态, 比赛过程中多次交换领先位置:

开局 Proxima 起步略快, 65 分钟就通过了 5 题 BDEFK。

开局看似顺利, 不过我们都明白: 真真的比拼还没有开始。

Proxima 卡在了 H 和 C 上, IronGods 通过了 BCDFK 追成 5:5 平, 罚时 Proxima 领先。

IronGods 通过了 G, 首次反超 6:5。

Proxima 经过 rejudge 通过了 H, 出现了 6:6 平, 罚时 Proxima 领先。

Proxima 第 10 次提交才通过了 C, 再次获得题数领先 7:6。

如果输掉了这次 PK, 题 C 则是最大的败笔。

IronGods 通过了 J, 追成 7:7 平, IronGods 在罚时上领先。

此时的罚时落后就是 Proxima 在 C 题上出错 9 次的恶果。

Proxima 第 4 次提交才通过 G, 以 8:7 反超, 但罚时还是很大。

IronGods 通过了 H, 又追成 8:8 平, 利用罚时 IronGods 再次获得领先。

这已经是第 6 次出现平分。这时还不到 3 个小时，校内 PK 赛的题目难度并不在 2008 杭州赛区的难度之下，3 小时的 8:8 的高比分平局是现场比赛中很难看到的。而在高比分平局中罚时也是很重要的，此时 IronGods 占据明显的优势。

Proxima 经过 rejudge 通过了 I，再次超出 9:8。

Proxima 通过了 J，优势扩大到 10:8。

记得题 J 的第一次提交开始的返回结果是“Other-Contact Staff”，看到这个回复之后 zhuzeyuan 马上跑到 Judge 室，在被工作人员挡住之后，zhuzeyuan 很奇怪地问道“难道不是你们让我来 Contact 的吗？”，囧死了。不过很快就 rejudge 成 Yes 了，题 J 的通过也从一定意义上逆转了罚时的不利，IronGods 如果想翻盘就必须在最后一小时重新上演北京赛区封版通过 3 题一幕。

Proxima 通过了 A，优势扩大到 11:8。

记得最后提交 A 题的时候，我紧张得手都有些发抖了。当时只剩下 25 分钟，IronGods 还没有开始写 A 和 L 两题，所以在最后的时间里他们已经不可能通过余下的 4 题了。A 题的 Yes 也就成为了这场 PK 的胜利宣言。

IronGods 最后时刻通过了 I，最终题数为 11:9。

此次校内 PK 的激烈程度决不亚于 2006 年上海赛区，能够最终赢得这场 PK 使得我们更有自信地站在总决赛的现场。

首先感谢关心我们的同学，记得第一场 PK 当天正在举行吉隆坡赛区比赛，bbs.pku 上还是出现了如此多的帖子为我们双方加油。第二场 PK 结束时已经是晚上 11 点，我们手机还不断收到祝贺短信。

向 IronGods 三位天王致敬，在 PK 过程中只需略微的变化，出现在斯德哥尔摩的就很可能是你们。棋逢对手是我 ACM 生涯的一大幸事，相信你们明年一定能够做得更好。

我想这是清华第一次使用公开的现场 PK 方式来选拔总决赛队伍，个人觉得 PK 的方式除了公平之外还有许多优点。首先，PK 方式可以使得各队伍能够更从容地选择和准备不同的分区赛赛区，有效提高学校的总体成绩。其次，通过 PK 的过程，可以加强各队之间的交流，队伍各方面水平能够得到全面提高。真是一举两得。

利用假期空闲之时，将这几年 GCJ，ACM，TopCoder 参加的一些重要比赛作个回顾，包括今天一共 10 篇。接下来的重要比赛就是世界总决赛了，纵观世界总决赛各队，虽然形势不容乐观，但我们一定会拼尽全力。

利用假期空闲之时，将这几年 GCJ, ACM, TopCoder 参加的一些重要比赛作个回顾。回顾 GCJ2006, ACM2005, TCCC2006 和 ACM2006 之后，今天简要回顾一下国内个人赛场吧。

国内个人赛场——百度之星

国内个人赛场中最重要的比赛要数每年一度的百度程序设计大赛，到今年为止已经举办了 4 届，每一届我都全身心地参加了比赛的全过程，百度程序设计大赛是中国举办的规模最大的公开程序设计比赛，其参加人数比许多世界范围的程序设计大赛的人数还要多得多。另外在 2006 年初，Google Beijing 举行了 Google Code Jam China 的比赛，刚刚开始参加 TopCoder 的我也加入了这次 GCJC 之旅。

第一届 baidu 程序设计大赛：

最早的国内个人程序设计比赛要回忆到 2005 年 9 月开始的第一届百度程序设计大赛了，源于宿舍走廊中的海报，我以尝试的心态报名参加了第一届百度程序设计大赛。每一届百度程序设计大赛都由初赛，复赛和现场决赛组成。

第一届百度程序设计大赛中，印象最深的复赛题目就是那道规模最小的最小树形图问题了，100000 的数据规模吓退了不少选手，我鼓足勇气提交了一个理论上能够运行的程序，顺利通过了复赛进入决赛。最小树形图算法在大多图论书上就接在最小生成树算法后面，但是其程序量远比最小生成树大，而且用途没有最小生成树广泛，在大多数竞赛中很少出现。我最早接触最小树形图算法是在 2003 年 4 月，当时正在复旦大学训练，记得关于这个问题和 xreborner 讨论了很长时间才得以证明算法的正确性并实现出高效的程序。

现场决赛于 2005 年 10 月底在北京举行，由于当年比赛的知名度不高，时间上还和 GCJ 冲突，没有太多的顶尖高手参加。清华大学除我之外只有 superzn（张宁，我们留 shell 一个人参加 ACM 北京赛区预赛☺），当时 OpenGL 还是以高中生身份参加的，还有复旦大学的 xreborner 和 young（李阳）；中山大学的 magicpig, Savior 和张子臻（不好意思，我不记得您的 ID 了，好像杭州 2008 的时候我们还说起此事）。我一直认为，现场比赛过程的一个重要的意义在于提供了一个老朋友重逢和结实新朋友的机会，选手之间的交流是比赛中最重要的组成部分之一，我很有幸能够在这些比赛中认识了众多牛人。

稍微回顾一下决赛的题目吧：决赛的题目是经典的 8 数码问题，给定初始状态和结束状态，计算最短需要的转移步数。对于分数相同的情况，按照程序的运行速度排名。比较容易想到的方法有：

- (1) 单向 BFS：最坏情况需要 1s 左右。
- (2) 双向 BFS：如果先判断无解情况，这是 xreborner 使用的方法，平均情况大概 0.002 秒左右。
- (3) A* 或者 IDA*：先判断无解情况，然后通过距离启发函数搜索。平均情况大概 0.002 秒左右。我当时使用了 A* 的方法，但许多地方的实现不是很合理。

(4) 常量表，这是最有挑战的方法，因为决赛的提交量限制在 64K 以内。

现场比赛中，(2)和(3)的使用人数比较多，速度相差无几，选手之间比拼的是各种细节和常数的处理。后来，我想出了一种速度非常快的方法：

首先使用 A*加上“卡节点”技术，就是限制 A*算法搜索过程中每层的节点个数上限，这种算法扩展节点个数在 100 左右。然后，由于上述算法的正确性不能保证，把所有反例打成常量，程序大概 50K 左右。很容易发现，这个程序的速度远比赛过程中所有程序的速度都快得多。

最终我的程序以总时间 0.022 秒获得冠军，xreborner 和 Savior 以 0.026 分并列第二名。xreborner 的程序很可惜，如果加入了无解判断，速度应该比我程序块，superzn 就更可惜了，superzn 的飘逸程序其实只有 0.020 秒，但是有一个数据错了。

记得颁奖之后，主持人邀请获奖选手发言，选手可以通过向前走一步选择优先发言。这时，我突然感觉大家把目光都聚焦到了我身上，向右一看，由于我站在最左边没有注意到右边的情况，可谁知其他选手都后退了一步，把我留在了看似向前一步的位置。

第二届 Astar-baidu 程序设计大赛：

第二届百度程序设计大赛没有等到 10 月，而是在 2006 年 6 月就拉开序幕。没有想到的是，第二届百度程序设计大赛竟然以我在一年前比赛中使用的 A*算法的名字命名，感到非常荣幸。

记得复赛的题目非常正式，印象最深的要数 xreborner 招牌式的 Zuma，我推了两个小时公式才得到了正确的动态规划方程，实现之后还由于 TLE 只有 30 分 (100 满分)。还有 Ying 出的无向图最小割问题，我用网络流算法又超时了。不过最后一题，我的程序竟然比 xreborner 优化过的标程还快，真是不容易呀。清华的舍友 RealPlayer 在复赛中表现很兴奋，可惜由于一个很小的错误没能进入决赛。

参加第二次百度决赛的选手中有许多熟悉的面孔，清华的同学包括 shell，OpenGL，lympanda 还有 Macsy。复旦大学也来了很多选手，其中除了 LemonTree 和 Topkiller（沈毅）之外，还有我刚到复旦时见过的 admin 和 funny 的身影。另外 magicpig 和 flymouse 也参加了，而且 magicpig 和我住一个房间，吃饭时记得他把桌上所有人的 Dev 功底全都鄙视了一遍，可惜 PE 不在场呀。比赛前还看到了 Srbga 的身影，据他说是被邀请一起来玩的，其实稍微用小脑判断一下就知道一定是参与出题的，有了 Srbga 的加盟，相信决赛题目绝对不会是一年前的风格了。

第二年决赛的题目是：著名的俄罗斯方块。写程序玩一个 10 列的标准 2 维俄罗斯方块游戏。

Srbga 设计了很有特色的记分方法和评分标准。对于记分方法，特别的地方是消去 1 行后没有得分，而同时消去 2, 3, 4 行的得分分别是 3, 6, 10，记分方法非常鼓励一次消去多行。评分标准则更奇怪了，有 50 种不同规模的数据，对于每组数据对所有选手的得分进行排名，前 8 名的选手依次得到 10, 7, 6, 5, 4, 3, 2, 1 分，也就是说，是存在可能性在测试结束之后分数仍然为 0 的。

比赛过程中，我花了许多时间来分析这个奇怪的评分标准。

对于这种评分标准，常见的策略有两大类：(1)所有数据的成绩比较平均 (2)在一种数据风格中特别突出呢。

根据数据描述，50 个数据可以分为 10 种不同的风格。参加比赛的总共有 50 名选手，如果所有分数是完全平均分配的话，分数是 31 分，这个数字意义不大。但如果设想分数的 80% 会分配在前 10 名中（根据当时选手的水平，这个假设还是比较合理的），这样前 10 名的平均分数是 124 分左右，也就是说如果想挤进前 4，至少也要 100 分以上，如果想争取冠军估计需要 200 分左右。如果选择一种策略，使得它只在一种数据风格中特别突出，分数只有可怜的 50 分，而且很可能有许多有同样想法的选手，所以(2)不太可取。

在决定选择比较平衡的策略(1)的之后，需要再考虑一个问题，如果最终目标是 150 分，那么平均分数只需要 3 分，也就是说每个数据可以允许有 5 名选手超过自己。这些必要的分析帮助我明确了努力的方向，面对这种开放性的题目，多分析题目的特点往往可以达到事半功倍的效果。

还有一个重要环节是调整估价函数，机器学习其实是一个很好的策略，可惜我当时不会。其实当时我做的事情，本质上就是人工模拟机器学习，手工调整了 1 个半小时，眼都花了。而且我犯下了一个致命的错误，记得记分方法非常鼓励一次消去多行，也就是说对于平坦的数据，一次消去 1-3 行的权值应该可能设置为负数，而我只把他们设成了 0，使得程序对于平坦的数据分数不高。Macsy 就考虑到了这一点，只可惜一个很奇怪的技术问题（在 Linux 和 Windows 下的 CLOCKS_PER_SEC 参数是不一样大的，想使用卡时策略时千万不能事先把这个数字取出来设置成 CONST）使得 Macsy 没能成功。

由于 Macsy 和 LemonTree（同样的技术问题）的出局，我在许多数据中得到了很高的分数，最后的总分达到了是 255，领先了第 2 名有 99 分之多。其实现场的许多选手的程序风格相差并不大，可能我唯一多做的事情就是建立了一个博弈树，多搜索两层，这样比直接贪心的程序看得更远一些。后来事实也证明了，排名靠前的选手大多都是比较平衡的策略。记得 lympanda 洋洋洒洒写了 1800 多行程序，在其中一种数据中拿到了满分 50 分。不过可惜 panda 的程序平衡性稍差，总排名进入了前 10，但最终只有三等奖。

第二届 astar-baidu 程序设计大赛，复旦大学获得了丰收。记得许多复旦的选手由于考试提前回到学校，颁奖仪式的时候二等奖颁奖一片空场。

比赛的住宿条件可以用无与伦比来形容，很感谢 baidu 的大方与细心。记得第一天晚上还有机会和 Ikki 一起打沙壶球，面对球风完全对立的 Ikki 玩得很开心。

Google Code Jam China 2006

大概是 2005 年末，突然看到了名为 GCJC 的比赛，而且使用的是 TopCoder 的比赛模式，于是就报名参加了。当时估计只参加过几场 TopCoder 的比赛，帐号还是蓝色的，GCJC 第二轮预选赛由于经验不足差一点就被淘汰了。好在有惊无险地进入到了北京的现场赛。

GCJC 现场的选手中，我觉得至少认识 80% 吧，清华同学就有 7 人：b142857，fuwenjie，lympanda，Macsy，zig 还有 hyyylr（李老师），复旦的 LemonTree 和

TopKiller 也都来了，浙大也来了许多 TopCoder 上的元老 xuchuan（徐串），sghao126。

记得，就在 GCJC 决赛的前一天晚上，我参加了 TopCoder 的 SRM 比赛，第一次踩住了 Petr，不过也消耗了太多的 RP。晚上的 SRM 比赛中没有人过 3 题，第二天早上 lympanda 还把我们统统鄙视了一遍。随后，b142857 还描述他 Challenge 过程中的囧事，由于 500 分题目的返回结果需要使用 long long 类型，所以 b142857 看到一个人提交的程序计算过程中只使用了 long 就果断 Challenge 了，结果失败了两次之后才发现，那个人用的语言是 Java。

比赛中 250 分题目，简单的概率问题。我写完就交了 224 分，竟然是所有选手中最快的。后面的 500 分，我虽然提交是最快的，不过没有考虑一种情况。打开 1000 分题目之后网络就开始很不稳定了，时断时连，1000 分题目其实算法很清楚，由于网络原因提交只有 600 分左右了。

Challenge 阶段开始时，我打开了房间中 lympanda 的 500 分程序，发现我们两人的程序基本过程完全一样。又打开了一个，也一样。但是在还没有反应过来的时候，lympanda 的 500 分程序被 Challenge 了，接着我的 500 分也被 Challenge 了。然后就没有什么斗志了，在无奈中等待比赛结束。

比赛结束之后的午饭过程中，我正好坐在 Google 中国掌门人李开复旁边。午餐快结束时，李开复问起 2 个月前的百度程序设计大赛，突然，鬼使神差地直接问我百度大赛的冠军是谁？这可是在 Google 的老巢呀，抖死了。我当时真害怕他听完回答之后直接把我赶下桌:-P。

好在我的 250 分和 1000 分都 Pass 了，由于 TopKiller 的 1000 分超时了，我获得了第 3 名。冠军 xuchuan 和亚军 b142857 都顺利通过了 3 题。

POJ Monthly Contest

大概是从 2004 年 8 月开始，POJ 上开始举行每月一次的有奖月赛。2005 年的月赛中，每次都有机会同 xreborner, Ying 等高手切磋技艺。从 2006 年初开始，我已经比较熟悉了比赛的题风，连续获得了许多次比赛的冠军，并且保持了良好的个人比赛状态。

记得 2006 年 4 月底，在 POJ 的邮箱里突然发现了 hawk 的信，他问我五一长假回家的情况。我告诉 hawk 自己定在周五晚上出发。于是，第二天早上就看到比赛安排中：2006 年 5 月份月赛安排在了周五晚上，太囧了。

后来，POJ 上直接出现了一系列奇怪的定义，但其实结论就是我不能以正式身份参加月赛了。现在这些定义早已成为笑料了，但是我不参加月赛之后，仍然有 ahyangyi 这样的选手夺走了绝大多数的冠军。

后两届 baidu 程序设计大赛：

从第二届开始，我们习惯了在每年 6 月等待 astar-baidu 的开赛。2007 年最出乎意料的就要数 CS 这个决赛题目了，我在关键的买枪环节犯了重要错误，太迷恋 AK47 了。祝贺师兄 lympanda, Macsy 还有 shell，不愧是真金不怕火炼。

第四届百度大赛我参与了预赛和复赛的命题工作，但是没有参与决赛的命题。决赛题目是一道关于直升机的题目，印象最深的是 **ahyangyi** 使用了一个很有进攻性的策略，如果采用淘汰赛，可能就是冠军了。对我来说，通过现场比赛，有机会和老朋友重逢，并结识了许多新选手是我最大的幸事。

利用假期空闲之时，将这几年 GCI, ACM, TopCoder 参加的一些重要比赛作个回顾。今天总结一下国际个人赛场吧。

国际个人赛场——三大赛事

ACM-ICPC 总决赛结束后，Mobile Robot 就宣布解散了，也许唯一的遗憾就是没能获得一个真正的世界冠军。宣布退役 ACM 之后，我仍然连续参加了那之后的每一场世界范围的现场编程比赛，按照时间先后分别是：TopCoder Open(简称 TCO)2007, TopCoder Collegiate Challenge (简称 TCCC)2007, TCO2008 以及 Google Code Jam(简称 GCI)2008。每次比赛，我都度过了一段美好快乐的时光。

TopCoder 公司与三大赛事：

TopCoder 公司大概在 9 年前成立，成立的原因有些让人匪夷所思，据说公司创立者原来是另一家 IT 公司的大股东，在把原来公司的股票转手之后换了一笔钱，开设了 TopCoder 公司。然而 TopCoder 和原来的 IT 公司有一个重要协议，就是 TopCoder 在创立之初的两年内不得从事软件开发的工作。于是 TopCoder 在前两年时间内以类似竞赛的方式从事软件开发的比赛。经过 9 年的发展，现在 TopCoder 公司已经基本由算法竞赛转向软件开发了。

TopCoder 公司除了在网上举办 SRM 之外，每年还举办 TCO 和 TCCC 等现场赛事（当然还有 TCHS，不过规模比较小，参与面也不是很大），TCO 和 TCCC 分别在每年的 6 月和 11 月举行，每次大赛都能汇聚众多国际编程高手。另外，Google 公司从 2000 年开始，先在各大洲举办名为 Google Code Jam 的比赛，从 2002 年开始也举办全球范围的 Google Code Jam。于是这些年来，大家一直把 TCO, TCCC 和 GCI 称为三大赛事。

2007 年之前的 GCI 都是使用 TopCoder 的比赛形式，TopCoder 的算法竞赛有点类似于 IOI, ACM-ICPC 之类的竞赛，题目是同一个类型的。每次比赛三道题目，一般分数分配为 250-500-1000，比赛分为 Coding, Rest, Challenge 和 System Test 四个阶段，时间各是 75 分钟（现场比赛 85 分钟），5 分钟，15 分钟（现场比赛 10 分钟）和不定。

TopCoder 的现场比赛都由 3 个阶段组成：所有选手被分为 3 个组（称为 Room1, 2, 3），每组分别进行半决赛，每组前 2(或 3)名直接晋级决赛，3-6 名晋级 wildcard 比赛，wildcard 比赛 12 人中的前两名填补决赛的最后 2(或 1)个名额，决赛由 8(或 10)名选手参加。由于三大赛事的比赛形式相差不多，每次现场决赛的选手中总是有许许多多熟悉的面孔。

三大赛事的波荡起伏：

可能细心的同学能够发现疑问，在文章最开始的一段中，我表明自己在 2007 年之后没有错过任何现场赛事，那为什么没有 GCI2007 呢？其实原因很简单，

Google 公司在 2007 年全年中只举办了面向美洲的比赛，没有举行面向全世界的公开赛。GCJ2007 的搁浅也使得整个 2007 年只有 TopCoder 公司独自举办世界大赛。

但是，当大家以为 GCJ 将在记忆中淡去的时候，GCJ2008 重新登陆，而且新的比赛环境与形式给选手以焕然一新的感觉。这里先谈谈自己对这种新比赛环境的看法吧：

GCJ2006 仍然使用的是 TopCoder 标准形式，也就是说和 TCO 以及 TCCC 完全一样，用一句话概括就是 Coding-250-500-1000-Challenge-SystemTests。

GCJ2008 比赛环境结合了 ACM，TopCoder 还有 IPSC(ipsc.ksp.sk)等多种比赛的特色。

(1)每道题目分为 Easy-Hard 两组数据，并且数据可以下载到本地，这点好像与 IPSC 很相似，另一个与 IPSC 的共同点就是，不限制选手使用的编程工具，包括肉眼观察或者人工搜索。

(2)Easy 数据则和 ACM 非常近似，即时提交评测，并且也设定了每次失败提交加 4 分钟的罚时。

(3)Hard 数据则更像 TopCoder 的形式了，Hard 数据由于是统一评测，System Tests 可以有效

地把悬念保留到最后一刻。

GCJ2008 的比赛形式是一种大胆的尝试，并且也已经有了很理想的结果。

另外，值得称赞的是，GCJ2008 中首次使用了分各大洲进行当地现场半决赛的赛制。使得排在前 500 名的选手得以参加各大洲的半决赛，也拉近了 Google 公司与选手之间的距离。从另一个角度来说，各大洲半决赛的方法很有效保证了决赛选手的水平。平心而论，TopCoder 现场比赛前的最后一轮网络淘汰赛对选手的压力很大，就连 Petr 在 2007 年都直接来了一个“滑铁卢”，连现场赛都没有进。而现场比赛的公平程度远超过网络赛，所以通过现场赛决定决赛选手可以一定程度上提高决赛选手的水平，至少我个人很赞同这种做法。

搁浅的比赛无独有偶，可能是受到了 2008 年全球经济危机的影响，TCCC2008 也停办了。而且我们都觉得，TCCC2007 很可能是 TopCoder 举行的最后一次 TCCC 了，当然 TopCoder 这样做没有不合理的地方。

TCO 则相对稳定一些，就连每年举行的地点都不变，TCO 连续 3 年在著名的赌城 Las Vegas 举行。今年应该也不会改变地点。

三大赛事的举办，我觉得选手最大的受益就是，比赛提供了一个到美国免费游玩的机会。我先后去过 7 次美国，其中 6 次都是参加编程比赛。通过比赛的机会，我们得以开阔眼界，结交朋友。我个人真心希望三大赛事能够继续举行，但是 2009 年秋天的 TCCC 和 GCJ 很可能同时停办，这也是一个不可避免的问题，让我们拭目以待吧。

美国之旅：

从 2007 年以来的 4 次现场比赛，虽然每次比赛过程中都有一些遗憾，但是现在回想起来都有不尽的乐趣。

TCO2007 是我第一次到达赌城，一下飞机就看到很多赌场(CASINO)，可谁知 TCO2007 整个比赛过程就是一场巨大的赌博。我当时由于不熟悉 Texas Hold'em 的规则，在半决赛中搞错了 Flush 和 Straight 的大小关系，结果初上赌场就倾家荡产而被淘汰出局。TopCoder 比赛中竟然出赌博有关的题目，果然有 Las Vegas 的特色呀。不过在赌场里，我仔细研究了许多赌博游戏的规则，然后写了几个程序计算赌博的期望，但是发现标准概率模型下所有游戏的期望值全是负数（其实挺显然的），于是，也就以娱乐为目的和 lympanda 切磋了一下。

如果说 TCCC2006 的 Room1 是中国的胜利，TCO2007 的 Room1 则是中国的失败了，虽然 Ying 和 lympanda 都进入了 wildcard，可是都由于一些小失误输掉了这次赌博。赛后 lympanda 请我去牛排馆吃饭，后来那个牛排馆也成为每次 TCO 比赛我们中国选手的主要聚会地点。

TCCC2007 的小组赛还比较顺利，我轻松击败了 gawry, Per, marek.cygan 获得小组第一挺进决赛。可是决赛中，我为了提高速度以超过 Petr，再加上有些紧张，最后 500 分和 1000 分两题又都挂了，落到了第 5 名。TCCC2007 地点设在了奥兰多，比赛结束后我们到附近的 Disney Land 去玩，那里的惊险游戏比国内刺激得多，有些远远超过我的极限，我们一行人一直玩到深夜才返回。许多选手还一起到奥兰多魔术队主场观看了 NBA 现场比赛，可惜最后一节成为了垃圾时间。

TCO2008 我也依靠飘逸的 1000 分题中 800+分的提交闯入决赛。决赛前我还和 visualage 聊天，夸耀自己从来没有所有题目全挂，更没有拿过负分。可是在随后的决赛中，这两个“梦想”就都实现了，PE 对我的评价是太紧张了。基本每次 TopCoder 现场比赛都能见到 PE，谁知他每次怀疑我某些题目的正确性的时候，我的程序就一定是错的，如果下次我参加决赛，您就不要再看我程序了吧（呵呵，开个玩笑）。

不过在决赛 Challenge 阶段的最后时刻，我从第一视角目睹了 Petr 和 Tomek 的巅峰对决。在还有 15 秒钟结束时 Petr 还落后 Tomek 大概 30 分左右，Petr 成功 Challenge 了一个超过了 Tomek，但是 Tomek 利用短短的 10 秒钟也提交了一个成功 Challenge 又超了回来，谁知 Petr 得到这个信息之后又提交了一个 Challenge，可是运气稍差，如果那个数据用来 Challenge 我的程序的话，Petr 就能够在最后 1 秒再次夺回冠军的位置。能够到最后一秒还能有机会成功翻盘的一定是神一般的人物，能够把神一般的人物逼到最后一秒的也一定是神一般的人物，两个神一般的人物你来我往，为大家上演了一场精彩的比赛。

欧洲独霸：

又一次引用黄金雄教授在杭州 2008 时说的话，ACM 总决赛的实力分布由原先的美洲独霸逐渐转向了现在的亚欧争霸。但是，我根据这些年的比赛结果发现，从 2006 年开始，团体比赛和个人比赛，特别是个人比赛，欧洲选手一直保持着绝对的霸主地位，亚欧争霸的说法实在有些牵强。

从 2005 年开始，几乎所有三大赛事的冠军都是欧洲选手。成绩最好的要数俄罗斯，俄罗斯选手以 Petr, andrewzta 等为代表。俄罗斯选手训练刻苦，编程能力

极强。欧洲的另一霸主就是波兰，波兰选手具有很强的灵气，以 torek, marek 以及 Eryx 为代表，程序设计在他们手中体现出了艺术气息。

前几天我也看到关于取消 NOIP 保送资格的文章，我没有发表评论，因为我没有看懂，为什么文章里把保送和保送资格混为一谈，让人觉得哭笑不得。这里我对保送资格还是想法不多，不过想比较一下我们中国选手与欧洲选手思维能力上的差别。

在高中时，吴文虎老师就常说中国选手的 IOI 成绩很优秀，的确这几年从 IOI 成绩上看，中国是绝对的霸主。可是 ACM-ICPC 的成绩，俄罗斯和波兰等强队的成绩却远在中国之上。于是我们总结的原因是：欧洲选手的编程能力强。我非常同意这个说法。

但是“欧洲选手的编程能力强”的说法并不说明他们的算法能力弱，相反他们的思维素质非常高，他们具有非常正统和严密的思维方式，体现出经过长期训练的思维能力和素质。

我觉得中国的“高手”和许多通过高考进入名校的“神人”，在大学之前接受的教育都是以选拔为目的的，并没有太多针对思维方式和能力的训练。记得小学要考重点 初中，初中则拼搏重点高中，高中期间则梦想名牌大学，而在学习期间，我们并没有太多机会训练自己的思维能力，至少在我的中学阶段是这样的。虽然很多高中已经竭尽全力通过类似研究性学习的方法锻炼我们的创新能力，但是仍然不能改变选拔性考试“高考”这一事实。而在与西方选手交流的过程中，我觉得许多思维能力优秀的学生很早就有机会接受系统的思维能力训练，寻找最适合自己的思考方法。我一次有机会看 Eryx 留下的草稿，发现他考虑问题有非常严密的过程，从理解题目到想出算法每步都有根有据，并不是随机碰撞的结果。

现在欧洲选手与我们相比，思维能力上也并没有劣势。我有幸在投身 OI 竞赛之后，得到许多机会与其它选手交流，学习他们的思考方法，努力锻炼自己这方面的能力，试图与众多欧洲选手对抗。

Mountain View 登顶：

GCI2008 在 Google 总部 Mountain View 举行，赛前我想用 Ying 的一句话来表达我对比赛夺冠的渴望，“我虽然获过很多奖，但是缺少一个世界冠军”。早在 GCI2006，我就拥有机会获得冠军，但是在失去那次机会之后一等就是整整的两年。

比赛开始不久，bmerry 的强势起跑使我逐渐失去了夺冠的念头，只得一心做好眼前的题目。bmerry 在不到 2 个小时的时间里就做出了除了 C-Hard 以外的所有题目，他只要在最后一小时做出 C-Hard，就基本上可以锁定冠军了。

不过我克服开场的不顺利之后，磕磕碰碰地在 2 小时过 5 分顺利通过了 E-Easy 和 E-Hard。摆在我面前的只有 B-Hard 和 C-Hard。B 题和 C 题相比之下，B 题我已经有了自己的想法，可是 C 则是完全没有想法。于是我决定先做 B，GCI2008 的 B 题简直是我的克星，我先后用了 100 分钟时间做这题都没有结果，可以说当时状态很差。大概到了 2:40 的时候，我查看 board 时突然发现了一件令人窒息的事情，bmerry 已经尝试了 C-Hard 并且超时了。由于 C-Hard 的分数略高于 B-Hard，我最后想要超过 bmerry 就必须做出 C-Hard。果断放弃 B-Hard 之后，并没有想出 C-Hard

的方法，写了一个搜索程序但是心里很没底，Hard 数据的提交时限是 8 分钟，于是到了 2 小时 52 分的时候，我毅然打开 C-Hard，用搜索的程序运行 C-Hard，在焦急的等待之后，程序在运行了 1 分多钟以后神奇地运行结束了。我依靠搜索方法通过了 C-Hard，一举超过了 bmerriy。1 分钟后 zhuzeyuan 也做出了同样的题目，超过了 bmerriy，由于罚时排在第 2 名。我和 zhuzeyuan 还有 bmerriy 比赛过程中都有不小的失误，我很有幸把失误的损失降到了最低点，终于获得了第一个世界比赛的冠军。

这次 GCJ 的题目有非常详细的解答，可以在比赛的链接里找到。GCJ2008 的比赛结果从一定意义上，打破了欧洲选手多年的独霸场面。加上原籍南非的 bmerriy，前五名中都没有出现欧洲选手的名字，这也是在多年现场比赛中没有出现过的。

这一年，我很高兴看到 OI 选手中出现了 ahyangyi, yuhch123, Loner 等各方面都极为出色的新人，真心希望你们能够早日适应大学的学习生活，再创佳绩。

众多新人的加盟，大大提高了清华 ACM 团队的实力。在 2008 年，清华大学 ACM 队创纪录地获得了 4 个分区赛的冠军。明天最后一篇回忆中将分享 ACM-2008 中发生的趣事。