

Learning strategies of ACM-ICPC

2018/5/30

PPT中可能会插播广告（timu）

- 请在大人陪同下小心观看

Typical way of being talented

- Concentration \times Time
- A little success
- Enthusiasm
- More and more time spent on it and efforts pay off
- Positive Feedback

ACM-ICPC is something about “AC”

How can problems be solved

- prerequisite
- experience of similar problems
- think in the same way of the “author of the problem”

给你一个简单的多边形的顶点坐标，请回答此多边形是顺时针还是逆时针。

- $3 \leq N \leq 30$
- $-1000 \leq x_i, y_i \leq 1000$
- 数据保证，这个简单多边形的面积不为零。

- 一种是算凸包 自带一个时钟方向
- 另一种是有向面积 凸包板子容易抄错 有向面积就一个叉积 手写就好了 方便
- $x_1y_2 - x_2y_1$ 夹角小于180就是正的 那么如果所有相邻点叉乘 加起来如果正的 就是逆时针
- 选的那个切分多边形为三角形的点 放哪里无所谓

这题体现了

- 足够多的知识/经验（知道有向面积，凸包），有时候弥补想法（其实是数学功底和思考能力）的不足
- 反过来思维能力也可以弥补经验的不足（现场推呗，大不了自己造轮子）

Too much plan will eliminate your passion

- Just choose “not too strange” problems
- And solve it

给你一个正整数 S ，若有若干个正整数的和为 S ，则这若干个的数的乘积最大是多少？

• 举例来说，当 $S = 5$ 时

• 1. $1 * 1 * 1 * 1 * 1 = 1$

• 2. $1 * 1 * 1 * 2 = 2$

• 3. $1 * 1 * 3 = 3$

• 4. $1 * 2 * 2 = 4$

• 5. $1 * 4 = 4$

• 6. $2 * 3 = 6$

• 7. $5 = 5$

• 其中乘积最大的是 $2 * 3 = 6$ 。

$$1 \leq T \leq 100$$

$$1 \leq S \leq 2000$$

- 如果有一个数是 n 那么 肯定拆的平均更好 二次函数对称轴了解一下 那么偶数的话 $n^2/4 \geq n$ 算一下是 $n \geq 4$ 所以超过3的肯定拆掉乘起来更大啊 当然如果是奇数是 $(n^2-1)/4 \geq n$
- 解一下 奇偶合并一下 发现结论就是大于3就要拆
- 接下来看看2合算还是3合算
- $3^{(n/3)}$ 和 $2^{(n/2)}$ 比较
- 三次根号三 和根号二比较
- 那发现3更好 所以能3就3
- 如果mod 3 为1 搞两个二
- 如果mod 3 为2 搞一个而
- 为0正好全部是3

Shorten the Feedback Cycle

- Separate learning to smaller processes which could give you positive feedback in a short time

Show some problems

- learn them together to illustrate our learning methods

Really Interresting problems

- Enjoy the process of thinking and solving problems

几个类似的适合享受 钻研难题的人的比赛吧

- 中考数学压轴题
- 高考数学压轴题
- 数学竞赛
- 信息学竞赛 / ACM
- CTF?

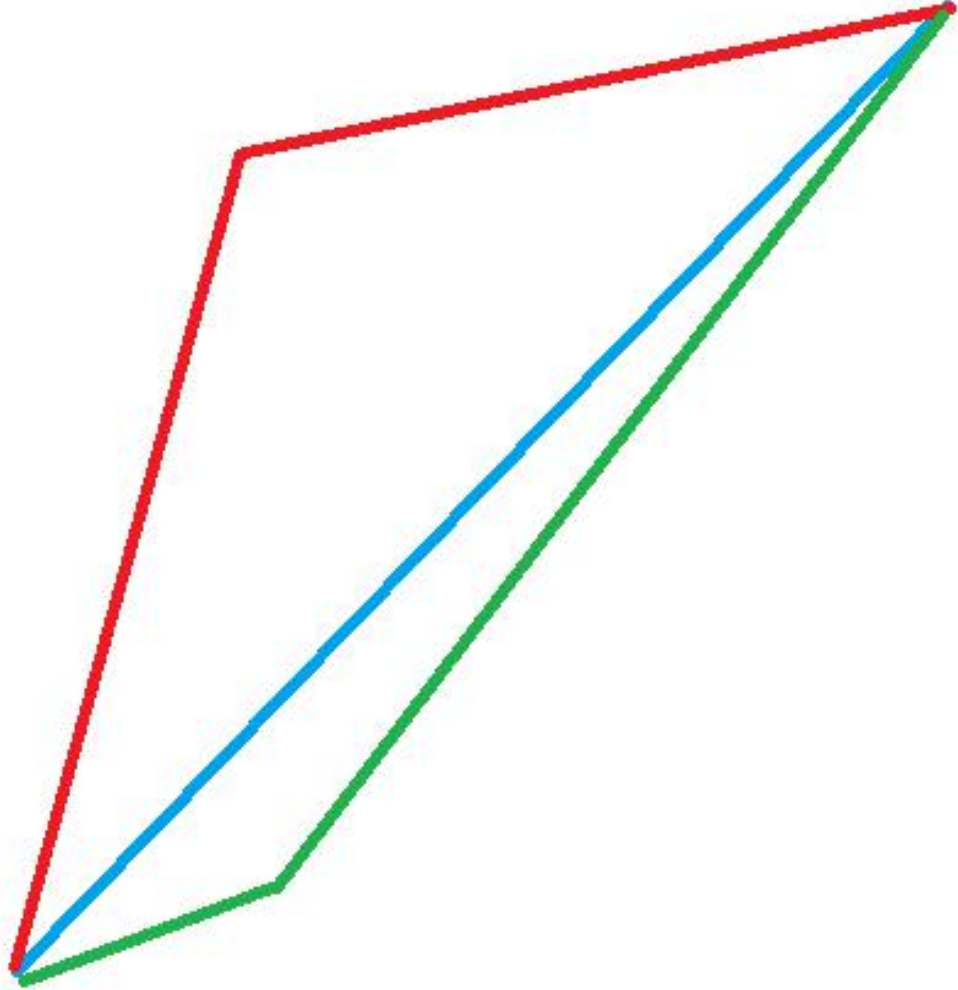
给出 n 张扑克牌，每张正面和反面分别有一个数字

- 每张扑克牌的数字只能选一个
- 请问，最多可以凑出 $1, 2, 3, 4, 5, \dots$ 到多少呢？
- ($n \leq 1e6$)

- 这题题面上看似乎挺好做的 实际上还是有点麻烦的
- 但这道题确实非常巧妙，而且以前没遇到过。
- 每张牌把正面的数字和反面的数字连一条边。
- 然后就会发现带环的大小是 p 的联通块，所有 p 个节点都可以满足，不带环的，可以满足 $p-1$ 个节点
- 那么实际上就做完了。这边还有个贪心，如果我必须要舍弃一个节点，让它不能满足。
- 一定是舍弃权值最大的。否则小的崩的，打的赢的没用了。
- 最后实现上，与其去dfs找联通块，不如直接用并查集，每次把权值小的往权值大的连。同时权值小的被满足。出现环，也就是连接的两个点在同一个并查集中的时候把并查集根(即权值最大节点)满足。

平面上有 N 个点，任意2个点确定一条直线，求出所有这些直线中，斜率最大的那条直线所通过的两个点。

- 第1行，一个数 N ， N 为点的数量。 ($2 \leq N \leq 10000$)
- 第2 - $N + 1$ 行：具体 N 个点的坐标， X Y 均为整数 ($-10^9 \leq X, Y \leq 10^9$)



讲故事环节

- 从讨厌学算法 到 喜欢算法题

1-n的全排列中逆序对为k的个数

- 第1行：一个数T，表示后面用作输入测试的数的数量。 ($1 \leq T \leq 10000$)
- 第2 - T + 1行：每行2个数n, k。中间用空格分隔。 ($2 \leq n \leq 1000, 0 \leq k \leq 20000$)

- $dp[n][k]$ 表示1-n的全排列中有k个逆序对有多少种
- 递推
- $dp[n][k]=\sum dp[n-1][k-i] \quad (0 \leq i \leq n-1)$

- 决定 n 个数排列的第一个是多少，剩下的 $n-1$ 个数，按照 $n-1$ 个数的排列方案来放
 - 就把一种 $n-1$ 个数的排列方案转换成了 n 个数的排列方案
 - 那么第一个数如果放最小的，不增加逆序对
 - 放最大的，增加 $n-1$ 个逆序对（和后面所有数都形成逆序对）
 - 所以有增加 $0、1、2\dots n-1$ 这么多种转移的方法
 - 就有了那个递推式了
-
- 好的，那么这种 $O(n^2*k)$ 复杂度太高
 - 前缀和优化一下，因为你要的就是连续的 $dp[n-1][k-(n-1) \dots k-0]$