NOIP2016模拟题 Day1
解题报告

ZYF & WMJ

October 2, 2016

1 小P的2048

认真读题，细心模拟即可。

请谨慎使用x_1, x_2, y_1, y_2作为变量，在某些版本C++编译器中可能会出现重名导致编译错误。

时间复杂度：$O(n^2m)$ / $O(n^3m)$

期望得分：100分

2 小P的单调数列

2.1 算法一

暴力枚举所有2^n个子序列，对于每个序列$O(n)$求出其艺术价值。

使用DFS可以节省求艺术价值的时间。

时间复杂度：$O(n \cdot 2^n)$ / $O(2^n)$

期望得分：20分

2.2 算法二

考虑动态规划。

设$f[i][j]$表示以$a[i]$结尾，且单调区间个数不超过j的最大子序列和，那么最后的答案便是$Answer = \max\limits_{1 \leqslant j \leqslant i \leqslant n} f[i][j]/j$，而状态转移方程我们也不难想到：

$$f[i][j]=\begin{cases}\max\limits_{1\leqslant k<i,a[k]<a[i]} \max(f[k][j-1],f[k][j])+a[i], & j\text{为奇数}\\ \max\limits_{1\leqslant k<i,a[k]>a[i]} \max(f[k][j-1],f[k][j])+a[i], & j\text{为偶数}\end{cases}$$

状态$f[i][j]$可以由$f[k][j-1]$转移过来，因为$a[k]$可能是两个单调区间的分界点。

时间复杂度：$O(n^3)$

期望得分：40分

2.3 算法三

经过思考，我们可以发现一个性质：**必定存在一个最优子序列，它的单调区间数不超过2**。具体证明如下：

假设最优子序列有$m(m\geqslant 3)$个单调区间，序列数字总和为S，则其艺术价值为$\frac{S}{m}$。

若m为奇数，此时最后一个单调区间必然为增区间。令该增区间的数字和为T，那么去掉最后一个增区间后，新的子序列的艺术价值为$\frac{S-T}{m-1}$，而显然$\frac{S}{m}$的大小在$\frac{S-T}{m-1},T$之间，所以我们保留艺术价值较大的一部分序列即可，此时序列单调区间数必然减少。

若m为偶数，则$m\geqslant 4$，此时我们令最后两个单调区间的和为T，那么去掉这两个单调区间后，新的子序列的艺术价值为$\frac{S-T}{m-2}$，而显然$\frac{S}{m}$的大小在$\frac{S-T}{m-2},\frac{T}{2}$之间，同理保留艺术价值较大的一部分即可。

综上所述，我们总可以找到一个单调区间数为1或2的序列，使得其艺术价值最大。

套用算法二，设状态$f[i][1/2]$类似转移即可。

时间复杂度：$O(n^2)$

期望得分：60分

2.4 算法四

进一步分析，我们可以发现，最优子序列的形态只可能为以下两种：

（1）单调增序列；（2）单调增序列+单调减序列。

于是，我们枚举分界点，求出以它为左端点的最大下降子序列和以它为右端点的最大上升子序列即可。离散化后，用树状数组轻松优化。

时间复杂度：$O(n\log n)$

期望得分：100分

3 小P的生成树

3.1 算法一

暴力枚举所有2^m个边集，$O(m)$判断该边集是否为一棵生成树，并求出其边权之和的模长，更新答案。

时间复杂度：$O(m \cdot 2^m)$

期望得分：10分

3.2 算法二

使用DFS枚举所有的生成树，在搜索过程中利用并查集进行可行性剪枝，同时使用估价函数进行最优化剪枝，这样可以轻松通过$n \leqslant 12$的数据。

根据$|z_1 + z_2| \leqslant |z_1| + |z_2|$，我们可以设计一个边集的估价函数为：将每条边对应复数的模长直接作为这条边新的权值，此时该边集的最大生成树的值。

时间复杂度：$O(???)$

期望得分：30分

3.3 算法三

当所有边的边权都为实数时，根据等式$|x| = \max\{x, -x\}$，我们只需要做两遍最大生成树，一次边权不变，一次边权取相反数，然后取两者最优值即可。

时间复杂度：$O(m \log m)$

期望得分：20分

3.4 算法四

本题的做法与最小方差生成树非常相似，没有做过最小方差生成树的选手可以参考一下这篇题解：**[BZOJ3754] Tree之最小方差树**。

假设我们已经知道生成树中各边求和所得复数的单位方向向量，那么要使得生成树的边权和模长最大，只需各复数在该方向向量上的投影之和最大。于是，我们将每条边对应复数在该方向向量上的投影作为其新的权值，做一遍最大生成树即可解决。

现在一个简单的想法就是，我们去枚举所有可能的方向向量，重赋权后得出该图的最大生成树，再判断这棵生成树的边权之和是否与当前这个方向向量对应，然后更新答案。但是，这样的方向向量的数目很显然非常多，我们无法全部枚举出来。

是的，方向向量的数目非常多，每条边对应的投影值也非常多。但是，由Kruskal的算法流程我们可以得到一个重要结论：

生成树的形态只与各边边权的相对大小有关，而与具体权值无关。

我们考虑两条边的边权$x_1 + y_1 i$与$x_2 + y_2 i(y_1 \neq y_2)$，当两条边对应复数的投影相等时，方向向量$(\cos\theta, \sin\theta)$需要满足：$x_1 \cos\theta + y_1 \sin\theta = x_2 \cos\theta + y_2 \sin\theta$，化简后得$\tan\theta = \frac{y_2 - y_1}{x_1 - x_2} \Rightarrow \theta_1 = \arctan\frac{y_2 - y_1}{x_1 - x_2}$，$\theta_2 = \arctan\frac{y_2 - y_1}{x_1 - x_2} + \pi$。而当$y_1 = y_2, x_1 \neq x_2$时，等式化为$x_1 \cos\theta = x_2 \cos\theta \Rightarrow \cos\theta = 0 \Rightarrow \theta_1 = \frac{\pi}{2}$，$\theta_2 = \frac{3\pi}{2}$。

我们枚举每一对边，算出两条边投影相等时极角的分界点，此时这些分界点会把$[-\frac{\pi}{2}, \frac{3\pi}{2})$的极角区间分成若干个小区间。由于每条边的投影大小关于极角θ是连续变化的，所以在每个小极角区间内，所有边的投影**相对大小关系不变**。

于是，我们枚举每个区间，在该区间内任取一个方向向量做一遍最大生成树，最后取最优方案即可。

时间复杂度：$O(m^3 \log m)$

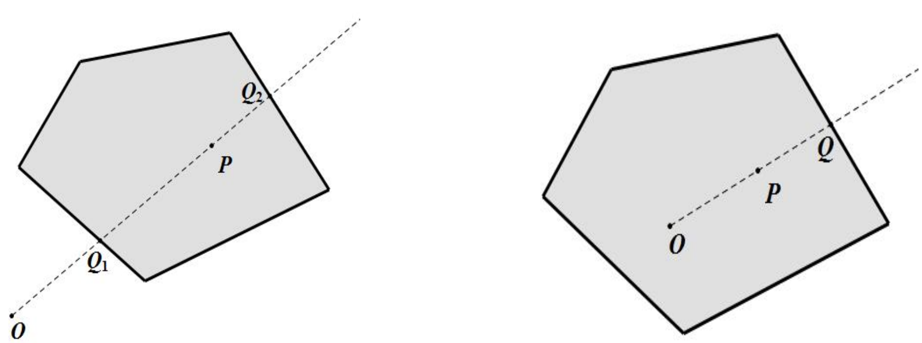
期望得分：100分

3.5 算法五

（这个算法原本是我在出数据时想卡掉的，后面卡着卡着发现它是对的2333）

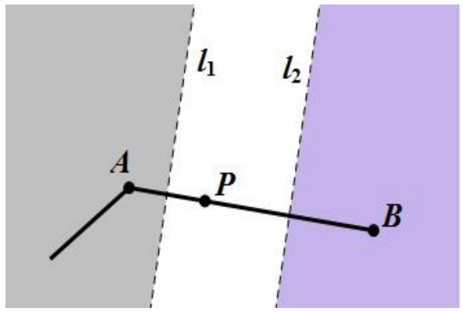
我们将每个生成树的边权之和看做是复平面上的一个点(A, B)，此时模长即为距原点O的距离，下面我们证明最优解必定会落在点集凸包的边上。

首先，根据凸包的定义，最优解对应的点P必定不在凸包的外部。假设P在凸包的内部，那么根据原点O所在的位置（凸包内/凸包外）可以分为如下两种情况：



(a) O点在凸包外　　(b) O点在凸包内

过原点O向P点作射线，此时射线与凸包会有1或2个交点，我们取距原点较远的那个交点，这个交点到O点的距离必然比P点要大，与P点是最优解矛盾。



而假设P点落在凸包的边AB上（不包括顶点），如上图，此时我们作AP, BP的中垂线l_1, l_2。要使得P点比A, B两点更优，那么原点O必定落在l_1左侧和l_2右侧。但$l_1 // l_2$，显然这两块阴影区域不可能有交集。

综上所述，最优解必定为凸包的某一个顶点。于是，我们套用最小乘积生成树问题中使用的Quickhull算法求出点集的凸包，再枚举每个顶点求出最优解即可。每次寻找距直线最远点，我们重赋边权做一遍最大生成树即可，时间复杂度为$O(m \log m)$。而对于一个随机点集S，其凸包上的点数期望为$O(\log |S|)$，所以本题我们求出的凸包期望顶点数只有$O(\log 2^m) = O(m)$。

时间复杂度：$O(m^2 \log m)$

期望得分：100分

更为详细的介绍请参考下列资料：

1. 唐文斌《图论专题之生成树》

2. BZOJ2395解题报告

3. Quickhull算法