Solutions to NOIP Simulation(day 2)

NickH

2018 年 10 月 1 日

1 distance

注意到这其实是一棵树，需要找一个点使得离这个点最远的点到这个点的距离最小。答案是树的直径长度除以2取上整，将树的直径画出来稍加分析即可证明。实在不行猜一个结论也肯定是猜这个结论。

求树的直径方法也非常经典，使用dp或者两遍dfs都可以。

2 graph

2.1 算法1

暴力dfs搜索所有S到T的路径然后取最小值，期望得分20分。

2.2 算法2

当图的点数，边数，边权都在几百的范围内时，可以使用暴力dp来做这个题。令$f_{i,j,k}$表示从S到i号点，是否存在一条最大值为j，最小值为k的路径，利用队列来存储等于1的f值，然后正向更新即可。

复杂度$O(100^2(n+m))$，和算法1结合可以得到50分。

2.3 算法3

注意到边数比较小，只有5000。我们可以把边按照权值从小到大排序$e_1, e_2, ..., e_m$，枚举S到T路径上的最小权值边e_k，那么我们只能够用$e_k, e_{k+1}, e_{k+2}, ...e_m$这些边来使得$S$到$T$连通，并且要找到一条$S$到$T$的最大值最小的路径。

最大值最小的路径在最小生成树上，所以只需要用$e_k, e_{k+1}, ..., e_m$这些边做一次最小生成树即可。假设在kruskal的过程中，加入某条边e_p的时候发现S和T第一次连通，那么这个最大值最小的边就是e_p，所以我们可以用$\frac{w(e_p)}{w(e_k)}$来更新答案。

复杂度$O(m^2 \log n)$，期望得分100分。

3 sweet

3.1 算法1

暴力枚举每种糖果是大的还是小的还是不买，然后验证。

期望得分30分。

3.2 算法2

题目给了这样一个条件，$2 \times A_i < B_i$。如果我们把每种糖果拆成两个，贡献都为1，价格分别为$A_i, B_i - A_i$。从小到大选，那么我们肯定会先选A_i，就保证了我们的选法是肯定合法的(因为价格为$B_i - A_i$的糖果必须要在选完A_i之后才能选)。贪心即可。

总复杂度为$O(n \log n)$，期望得分50分。

3.3 算法3

由于题目中没有了$2 \times A_i < B_i$的条件，若我们直接贪心会导致方案不合法，所以我们调整贪心策略。首先我们将糖果分成两类，X类满足$2 \times A_i < B_i$，Y类则不满足。我们枚举Y类的贡献q，X类当然直接贪心选前$P-q$个，将Y按照B_i 排序，同样令两种新糖果的价格$T_i = A_i, R_i = B_i - A_i$。

1. 若q为偶数，那么我们肯定会选取Y中的前$\frac{q}{2}$种糖果的大糖果。

   证明：明显选不在前$\frac{q}{2}$种中的一个大糖果不会比上述方案优。若我们选了前$\frac{q}{2}$种中的一个小糖果j和不在前$\frac{q}{2}$种中的一个小糖果k，现在有$T_j + R_j \le T_k + R_k, T_j > R_j, T_k > R_k$，要证明$T_j + R_j \le T_j + T_k$，即$R_j \le T_k$。反证法，若$R_j > T_k$，则有$T_j > R_j > T_k > R_k$，即$T_j + R_j > T_k + R_k$，矛盾。

2. 若q为奇数，我们有两种选法。要么选前$\frac{q-1}{2}$种大糖果加后面一个花费最小的小糖果，要么选前$\frac{q+1}{2}$种大糖果并在前$\frac{q+1}{2}$种除去一种花费最大的大糖果换成小糖果。这种方案正确性的证明与1方法相同，不再赘述。

总复杂度$O(n \log n)$。