黎锦灏　518021910771　0413作业

1、给定平面上一组点，已知每个点的坐标，求最远点对之间的距离，即点集的直径。（不得穷举，文献查阅，然后用自己的语言进行算法思想的描述，包括时间复杂性分析）

解：先构建这一组点的凸包，点集的直径一定是位于凸包上的两个点的距离，即凸包的一个对角线。那么对于每个点来说，可以求出与其距离最远的点，并求得最远距离，对所有点的最远距离取 Max 即可得到点集直径。

旋转卡壳算法：先任取凸包上的一个点 p 作为起点，用遍历求出与这个距离最远的点 q ，将距离作为 ans 的初值。过p、q分别做距离的垂线，同时旋转这对平行线， 直到某一条线与凸包的边重合，则对于新的点对更新最远距离，旋转一周可以遍历所有节点的最远点对，取最大的即为点集直径。

构建凸包的复杂度为 $O(nlogn)$，使用旋转卡壳法分析点集直径时只需要先得到一个点的最远距离，再依次使用旋转卡壳算法，遍历每个点一次，故复杂度为 $O(n)$，所以总时间复杂度为 $O(nlogn)$。

2、给定测度空间中位于同一平面的 n 个点，已知任意两点之间的距离 dij，存储在矩阵 D 中，求这组点的直径。

该问题的直观解法就是把 D 扫描一遍，选择其中最大的元素即可。由于是在一个测度空间中，因此dij满足距离的基本要求，即非负性、对称性和三角不等式，我们就可以给出一种时间亚线性的近似算法。算法很简单，由原来确定性算法的检查整个矩阵改为只随机检查 D 的某一行，这样时间复杂性就由原来的 O(n2)减少为 O(n)，相对于输入规模 n2而言，这是一个时间亚线性的算法。

证明时间代价减小的同时，解不会小于最优值的一半

证明：不妨设检查的某一行为 i ，而直径的两个点为 设s 和 t ，设 $maxdis(i) = max(dis(i,s), dis(i,t)) = dis(i,s)$，在$i$、$s$、$t$构成的三角形中有：$dis(s,t) < dis(i,s) + dis(i,t)$，由于$dis(i,s) > dis(i,t)$，则$dis(s,t) < 2dis(i,s)$。

即$dis(i,s) > \frac{1}{2}dis(s,t)$，随机检查某一行得到的解，不会小于最优值的一半。

3、在平面上给定一个有 n 个点的集合 S，求 S 的极大点。极大点的定义：设 p1=(x1,y1)和p2=(x2,y2)是平面上的两个点，如果 x1≤x2 并且 y1≤y2，则称 p2 支配 p1，记为p1<p2。点集 S 中的点 p 为极大点，意味着在 S 中找不到一个点 q，q≠p 并且p ≺ q，即 p 不被 S 中其它点支配。

解：注意到极大点都是分布在平面点集的右上部分。先对所有节点按x坐标从小到大排序，若x坐标相同则按y从小到大排序（即第一关键字为x，第二关键字为y的排序）。

然后从最后一个节点开始从后往前遍历，若当前节点的y坐标大于之前遍历的最大y坐标，则当前节点是一个极大点（因为满足yi>yj,即不存在能支配i的点），将当前点加入极大点集合即可。

排序复杂度为$O(nlogn)$，遍历复杂度为$O(n)$，故总复杂度为$O(nlogn)$。

黎锦灏　518021910771　0415作业

1、对凸多边形，1）有多少种三角划分的方法？2）如何使对角线长度之和最小？
如果觉得递推式不好求，写出递归式就可以了

解：1）不妨设凸N边形的三角划分方案数为$f(N)$，则

$f(1)=0, f(2)=0, f(3)=1, f(4)=2, f(5)=4$

且对于凸N边形做第一次三角划分，考虑分为凸i边形和凸$(N+2-i)$边形，这两个结果都是先前已经得到的，可以递归求得，

故 $f(N)=\sum_{i=3}^{N-1} f(i)*f(N+2-i)$

2）对角线长度之和最小，可以转换为使得所有三角形边的长度之和最小（原凸多边形的边算入一次，每一条对角线被算入两次）。

若凸$(N+1)$边形$P=(V_0, V_1 \ldots\ldots V_n)$的最优三角剖分 T 包含三角形 $V_0V_kV_n$,$1<=k<=n$，则 T 的所有三角形边之和可分为为三个部分长度之和：三角形 $V_0V_kV_n$ 的，多边形$(V_0, V_1 \ldots\ldots V_k)$的所有三角形边之和和多边形$(V_k, V_{k+1} \ldots\ldots V_n)$的所有三角形边之和。
设 $t[i][j], 1<=i<j<=N$ 为凸多边形$(V_{i-1}, V_i \ldots\ldots V_j)$的最优三角剖分所对应的所有三角形边的长度之和，即其最优值。最优剖分包含三角形 $V_{i-1}V_kV_j$ 的边长度之和，子多边形$(V_{i-1}, V_i \ldots\ldots V_k)$的三角形边的长度之和，子多边形$(V_k，V_{k+1} \ldots\ldots V_j)$的三角形边的长度之和。

若$i=j$，则$t[i][j]=0$；

若$i \neq j$，则$t[i][j]=min(t[i][k]+t[k+1][j]+S(v_{k-1}v_kv_j))$

综上，要使得所进行的三角划分出来的凸多边形对角线长度之和最小，可利用动态规划求解。

2、给定平面上 n 条线段，设计算法用 O(nlogn)时间确定其中是否有两条线段相交。

解：与扫描线算法求所有线段的交点做法类似，由于只需要判断是否存在两个线段相交，故不需要搜索每一个交点的信息，只要对每一条线段查看与相邻的两条线段是否相交即可。存在则直接返回有两条线段相交。

总时间复杂度为$O(nlogn)$

3、用扫描线算法求解最近邻点对问题

解：先将所有节点按x坐标排序，然后从左往右扫描。一开始先对相邻节点求距离，取最小值为初始的最近距离d。然后扫描时，使用平衡树维护一个序列，若当前节点为i，该序列满足$|x_i-x_j| \leq d$，则每次搜索满足$|y_i-y_j| \leq d$的区间的点，求最小距离并更新d。最后得到的d为最近距离。

总时间复杂度为：$O(nlogn)$

4、有 n 种液体 S1,S2,...,Sn，都含有 A,B 两种成分，含量分别为{a1,b1},{a2,b2},...,{an,bn}，ai+bi<100%。现欲利用这 n 种液体配制目标液体 T，使之 A 和 B 的含量分别为 x 和 y。设计算法判别能否成功配制，并给出算法时间复杂性。

解：考虑配制成功的条件，即需要 n 种液体中至少有 2 种液体的 $\frac{a}{b}$ 等于 $\frac{x}{y}$，且其中存在至少一种的含量大于 x，存在至少一种的含量小于 x。