

音符 (music)

条件即为一首乐曲中音高的极差不超过 k ，若选取的最小值为 x ，则一定会将音高在 $[x, x + k]$ 内的音符都选上。将序列从小到大排序后，等价于选取两段不交的极差 $\leq k$ 的区间，最大化区间长度和。枚举分界点，在前后各选出一个区间，处理一下前缀中能选的最大区间长度即可。

送信卒 (msg)

随着 k 变大，最短路长度也会不断变大，因此可以考虑二分 k 的取值，然后跑最短路，根据最短路长度与 s 的关系调整二分的边界即可。

字符串 (string)

枚举 x 。若 $x \mid n$ ，将原串划分成若干段，处理出 cnt_s 表示划分后 s 的出现次数，方案数即为 $\frac{(\frac{n}{x})!}{\prod cnt_s}$ 。

否则，枚举多出来的长为 m 的串 t 的位置，这里的枚举量是 $O(\frac{n}{x})$ 的，总枚举量是调和级数 $O(n \ln n)$ 。枚举后按照上面的方法算方案数就行，但是不同的 t 也可能让最终拼出来的串相等，而这种情况只会在枚举 t 后剩余划分出来的集合 $\{s\}$ 相等时出现，再使用集合哈希判定 $\{s\}$ 这个集合是否已经出现过去重即可。使用手写哈希表，时间复杂度 $O(n \ln n)$ 。

树 (tree)

分别建出最小、最大生成树的 Kruskal 重构树（注意这里是关于点的重构树）。此时一条路径 (u, v) 满足条件等价于在一棵重构树上 u 是 v 的祖先，在另一棵重构树上 v 是 u 的祖先。

考虑在第一棵重构树上 DFS，走到 v 时， v 到根链上的点可以成为 u 。可以维护到根链的信息，DFS 进入 v 时在第二棵树上加入 v ，回溯时删掉 v ，对于点 v 查询在第二棵树上有多少个点是 v 的后代即可，可以用树状数组维护单点加子树和，时间复杂度 $O(n \log n)$ 。