

Noip-CSP 2020 模拟赛题解

2020 年 10 月 5 日

1 blocks

1.1 70 %

$O(n^2)$ 处理，枚举两堆积木 i, j ，计算 j 追上 i 至少需要多少天，取最小值即可。

1.2 100 %

分析题目性质，发现 i 每天只比 $i - 1$ 多加1，只比 $i - 2$ 多加2，也就意味着在 i 要追上 $i - 2$ ，相当于 i 每天花1的价值，追上 $i - 1$ ，同时在 $i - 1$ 和 $i - 2$ 的差距上花1的价值追上 $i - 2$ ，即花费的时间比 $\min(i \text{追上} i - 1 \text{的时间}, i - 1 \text{追上} i - 2 \text{的时间})$ 的时间要长，也就意味着答案为相邻两天差距的最小值。

2 sort

2.1 40 %

指数级别大爆搜

2.2 80 %

将序列分为两个部分，前半部分和后半部分，我们发现需要做的即将前半部分排在 $n/2$ 名后的和后半部分排在 $n/2$ 前的交换位置即可。

仔细分析，发现在哈密顿距离下，任意交换花费的值相同，答案即为后半部分排在 $n/2$ 前的位置和 - 前半部分排在 $n/2$ 名后的位置和。

2.3 100 %

对数值分析，发现答案会超过 int 范围，开 long long （提醒一下，noip争取不要再出现同样的错误）

3 *string*

3.1 30 %

指数级别大爆搜，枚举每个串在哪个集合，并计算公共前缀长度

3.2 60 %

在无序的情况下思考比较困难，由于问题与顺序无关，不妨从考虑从有序的角度思考。

排序后，发现前缀存在一个性质，即 i 和 $i + 2$ 的公共前缀长度 = $\min(i$ 和 $i + 1$ 的公共前缀, $i+1$ 和 $i+2$ 的公共前缀),也就意味着，假设现在剩余 n 串，将 n 个串排序后看做一个环形，其中包含第1个串的集合，为环形上连续的一块，采用区间动态规划处理。

$f[i][j]$ 表示区间剩余 $[i,j]$ 的最大美观度， $O(k)$ 枚举第 i 串和哪一串合并在一起，并转移。

时间复杂度 $O(n^2)$

3.3 100 %

由于问题与前缀相关，考虑在trie树上处理，用所有字符串建立一棵trie树，我们通过前序遍历可以得到字符串排序后的结果。

在trie树上思考，对于一个子树，从tire深度最深的deep层开始，如果能够凑出 K 个，肯定凑成一个集合是最优的，否则就会被放置在 $deep - 1$ 层进行统一考虑，也就意味着，在遍历trie时，记录两个数，一个是已经凑出的集合的美观度之和，一个是剩余未放入集合的字符串个数，统计答案即可。

4 *dinner*

4.1 40 %

对于每一个方案，将方案提出的区间标为1，再枚举每一对关系不好的同学，判断是否同时为1，时间复杂度 $O(Q * (n + m))$

4.2 80 %

k 比较小，考虑设计一个与 k 相关的算法，离线处理+线段树维护，考虑关系不好的对 $[i,j]$ ， $i \leq j$ ，从1到 n 枚举每一个同学，枚举到 j 时，将与 j 关系不好的位置 i 在线段树中的值设为 j ，

如果 j 为某个方案中某个区间 $[x,j]$ 的右端点，即我们在线段树中查找该方案中，该区间前的区间最大值，如果存在某个区间最大值大于 x ，则该方案存在冲突，对于每个方案，假设有 k_i 个区间，则需要查询 k_i^2 次，总复杂度 $O(m \log(n) + \sum k_i^2 \log(n))$

4.3 100 %

做法一：

考虑通过差分+位运算压位来解决问题。考虑暴力”将方案提出的区间标为1”，即差分后将两个区间端点左端点加1，右端点+1位置减1，随后，通过前缀和，我们可以得到整个区间的每个位置的表示，发现一个int数值只置为1是非常浪费的，可以考虑位运算压位，long long 60位每一位处理一个方案。

时间复杂度 $O(Q * (n + m) / 60)$

做法二：

考虑分块处理。将40分做法和80分做法结合起来，当 $k \geq lim$ 时，采用40分做法，当 $k < lim$ 时采用80分做法。

时间复杂度 $O((n + m) * Q / lim + (n + m)\log(n) + Q * lim * \log(n))$