

NOIP2019 模拟

Day2 题解

一. 题目概况

| | | | |
|---------|-----------|-----------|----------|
| 题目名称 | doubt | block | road |
| 可执行文件名 | doubt | block | road |
| 输入文件名 | doubt.in | block.in | road.in |
| 输出文件名 | doubt.out | block.out | road.out |
| 时间限制 | 1.0s | 2.0s | 1.0s |
| 空间限制 | 256MB | 128MB | 256MB |
| 测试点数量 | 打包测试 | 20 | 20 |
| 单个测试点分值 | 见题面 | 5 | 5 |
| 题目类型 | 传统 | 传统 | 传统 |

二. 可执行文件名需加后缀

| | | | |
|----------|-----------|-----------|----------|
| 对于 C++语言 | doubt.cpp | block.cpp | road.cpp |
| 对于 C 语言 | doubt.c | block.c | road.c |

三. 编译选项

| | | | |
|----------|-----|-----|-----|
| 对于 C++语言 | -lm | -lm | -lm |
| 对于 C 语言 | -lm | -lm | -lm |

四. 注意事项

1. 文件名(程序名和输入输出文件名)必须使用英文小写。
2. 除非特殊说明, 结果比较方式均为忽略行末空格及文末回车的全文比较。
3. C/C++中函数 main() 的返回值必须是 int, 程序正常结束时返回值必须是 0。

1 doubt

1.1 题意简述

对数组 a 和 b 重新排列，定义一个数组 c ，满足 $c_i = a_i \text{ xor } b_i$ 。求字典序最小的 c 。

1.2 算法 1

$n!$ 暴力枚举 a 。 b 不变，算出 c 之后对 c 排序。

复杂度 $O(n! \log(n!))$ 。期望得分 10。

1.3 算法 2

考虑预处理出 $a[i] \text{ xor } b[j]$ ，从小到大贪心选取，用过的不再选。

考虑当有 $a[i] \text{ xor } b[j] = a[k] \text{ xor } b[t]$ ，选取顺序无所谓；当有 $a[i] \text{ xor } b[j] = a[i] \text{ xor } b[t]$ ，那么 $b[j] = b[t]$ ，选取顺序也无所谓。所以贪心的正确性可证。

复杂度 $O(n^2 \log(n^2))$ ，期望得分 30。

1.4 算法 3

由于 a_i, b_i 范围很小，在算法 2 的基础上，考虑对于相同的 (a_i, b_j) 一起计算处理。

结合算法 2，期望得分 50。

1.5 算法 4

对于 a 和 b 分别建一颗字典树，一起 dfs 使得每个 a_i 和一个 b_j 相消。考虑将 a 字典树上的子树 A 和 b 字典树上的子树 B 尽量合并相消，先合并 $(lson[A], lson[B])$ 、 $(rson[A], rson[B])$ ，再合并 $(lson[A], rson[B])$ 、 $(rson[A], lson[B])$ 。为保证复杂度正确，当某棵子树已经消完时，直接 return 即可，最后需要对 c 重新排序。

复杂度为 $O(n \log(2^{30}))$ ，期望得分 100。

2 block

首先考虑一个矩形如何求答案。

矩形的相邻两行只有两种情况符合要求：

- 两行相应位置的颜色相反
- 两行相应位置的颜色相同，且行内部为一黑一白交替排列

回到原问题，我们可以从高到低 dp。

记 $dp[l][r][h][p=0/1][q=0/1][k=0/1]$ 表示第 l 列到第 r 列，高度大于等于 h 的部分的合法染色方案数，其中高为 h 的行最左端颜色为 p ，最右端颜色为 q ，内部是否为黑白交替用 k 表示。将 h_i 排序后，每次加入一个新列 t ，我们需要合并 $[?, t-1]$ 和 $[t+1, ?]$ 两个区间的答案，在合并前需要将两个区间的答案从旧高度 h 更新到新高度 h_t 。合并后删除原来的两个旧区间。

由于任何时刻任何一列 i 最多属于一个有效区间，故 h 这一维不需要维护。

每次加入一个新列我们只需要修改并删除两个旧状态，添加一个新状态，故 dp 的复杂度为 $O(n)$ 。

一开始需要对 h_i 排序，故时间复杂度为 $O(n \log n)$ 。

3 road

3.1 题意简述

求 n 个点 $n+1$ 条边组成一个边双连通图的方案数，要求无重边和自环。

3.2 算法 1

手动打表 or 暴力枚举每条边取或不取。期望得分 10。

3.3 算法 2

考虑所有点度数和为 $2n+2$ ，而不能有点度数为 1，所以大部分点的度数为 2。

分为以下几种情况讨论：

- 1、有一个点度数为 4，整个图是以它为公共点的两个环。
- 2、有两个点度数为 3，这两个点直接连接，而且还有两条链连接。
- 3、有两个点度数为 3，这两个点未直接连接，中间有三条链连接。

对于这三种情况，预处理组合数，枚举链上的点数求得答案。

时间复杂度 $O(n^2)$ ，期望得分 40。

3.4 算法 3

考虑将算法 2 的式子合并，可得通项公式：

$$Ans = \frac{n!(n-4)}{8} + \frac{n!(n-3)}{4} + \frac{n!(n-3)(n-4)}{24}$$

时间复杂度 $O(n)$ ，期望得分 80。

3.5 算法 4

在算法 3 的基础上，分段打表预处理 $n!$ 即可。期望得分 100。