建议开题顺序：$T1 \to T3 \to T2 \to T4$。

## 传送门（portal）

- 原创；灵感源于省中 2019 年模拟赛，原出题人：徐翊轩。
- 构造，二进制
- 赛后提交地址：https://www.luogu.com.cn/problem/U149134

**解法零**

输出样例。

预计得分 5 分。

**解法一**

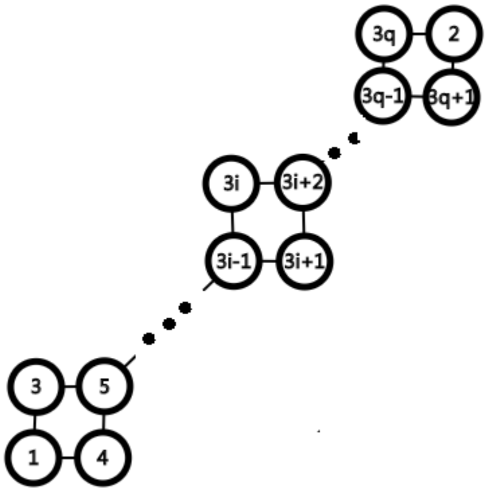
自行设计 $k \le 10$ 的数据。

预计得分 5 ~ 35 分。

**解法二**

本部分分旨在为选手想出正解提供灵感。

观察样例，不难想出以下构造方法，我们称其为「方格形」图，其中 $q = \log_2 k$。



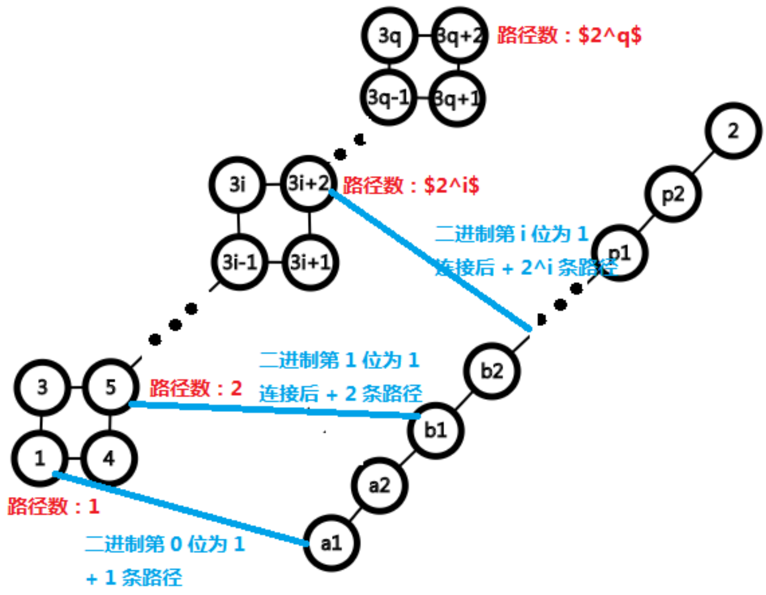

明显，从 1 号传送门走到 2 号传送门有 $k$ 种方法。使用节点数为 $3q+1$。$q_{\max} = 30$，能够拿到全分。

预计得分 25 分。加上解法一可以获得 30 ~ 60 分。

**解法三**

从解法二的方法继续思考。考虑把 $n$ 进行二进制拆分。

如下图，再构造一条含有 $2q+1$ 个节点的链：




将 $k$ 二进制拆分后，记最低位为第 0 位。若第 $i$ 位二进制位为 1，则如图连接对应的两个点。

不难发现此图符合要求，使用节点数为 $5q+2$，极限情况时大约需要使用 150 个节点。

预计得分 100 分。

std 的构造与连接方法与该图有出入，但是思路一致。

# 摆肖像（portrait）

- 来源：IOI 2018 练习赛 T3 Bubble Sort 2。
- 权值线段树
- 赛后提交地址：https://www.luogu.com.cn/problem/U149135

**解法零**

~~为防止爆零，特设此档部分分~~

`freopen`

预计得分 5 分。

**解法一**

每次修改后，都进行冒泡排序。

时间复杂度 $O(Q \times N^2)$，预计得分 30 分。

**解法二**

观察冒泡排序的过程，不难发现：在一轮中，每一个数前面最大的数总会被移到这个数后面，而其他数与这个数的相对位置关系保持不变。

故可得答案为 $\max_{i=1}^{n}\{\ a_i$ 前面比 $a_i$ 大的数的个数 $\}+1$，即为 $\max_{i=1}^{n}\{\ i-(\ a_i$ 前面 $\leq a_i$ 的数的个数 $)\}+1$

先暴力 $O(N^2)$ 算出初始序列的答案。每次单点修改时，就只需要用 $O(N)$ 算出被修改处的答案，$O(N)$ 更新这个被修改处对后面的影响即可。

时间复杂度 $O(N^2+QN)$，预计得分 55 分。

**解法三？？**

注意到值域很小，考虑从值域入手。

~~应该是可以做的，但是出题人没想到什么解法，就设了这个部分分，希望看看各位同学有什么做法。~~

如果你想到了这个部分分的解法欢迎联系出题人。

~~预计得分 20 分。加上解法二可得 75 分。~~

**解法四**

$\max_{i=1}^{n}\{\ i-(\ a_i$ 前面 $\leq a_i$ 的数的个数 $)\}+1$ 可以用 KD Tree 或者树套树维护。

时间复杂度两个 $\log$，预计得分 85 分。常数优秀的话可以拿到 100 分。

**解法五**

可以发现，若存在 $x,y\in[1,n]$，使得 $x<y$，$a_x\geq a_y$，那么，$x$ 一定不会成为答案。

把答案式子改为 $\max_{i=1}^{n}\{\ i-(\ \leq a_i$ 的数的个数 $)\}+1$。那么，如果 $x$ 位置处后面还有比它小的数，会多减导致不优，但是此时 $x$ 位置一定不是答案。故所求可以变为 $\max_{i=1}^{n}\{\ i-(\ \leq a_i$ 的数的个数 $)\}+1$

离线用权值线段树维护即可。时间复杂度 $O((N+Q)\log(N+Q))$，预计得分 100 分。

**道理我都懂，但为啥 std 要跑 3s 却只开 4s？**

std 的处理方法并不优秀，可以继续优化。

## 审疑犯（possibility）

- 来源：Project Euler 288。PE 难度系数 35%
- 简单数论
- 赛后提交地址：https://www.luogu.com.cn/problem/U160376

**解法零**

~~为防止爆零，特设此档部分分~~

输出样例。

预计得分 4 分。

**解法一**

算出 $Num(p,q)!$ 后直接质因数分解。

预计得分 15 分。

**解法二**

记 $\nu_p(x)$ 为 $x$ 中因子 $p$ 的次数，易有：

$$\nu_p(n!) = \sum_{i=1,p^i\le n} \left\lfloor \frac{n}{p^i} \right\rfloor$$

记 $N = Num(p,q) = \sum_{i=0}^{q} D_i p^i$ ，带入上式有：

$$\begin{aligned}\nu_p(N!) &= \sum_{i=1,p^i\le N} \left\lfloor \frac{N}{p^i} \right\rfloor \\ &= \sum_{i=1,p^i\le N} \left\lfloor \frac{\sum_{j=0}^{q} D_j p^j}{p^i} \right\rfloor \\ &= \sum_{i=1,p^i\le N} \left\lfloor \sum_{j=0}^{q} D_j p^{j-i} \right\rfloor \\ &= \sum_{i=1,p^i\le N} \left\lfloor \sum_{j=i}^{q} D_j p^{j-i} + \sum_{j=0}^{i-1} D_j p^{j-i} \right\rfloor\end{aligned}$$

注意到 $0 \le D_j \le p-1$，故有：

$$\begin{aligned}\sum_{j=0}^{i-1} D_j p^{j-i} &\le \sum_{j=0}^{i-1} (p-1)\times p^{j-i} \\ &= 1 - p^{-1} + p^{-1} - p^{-2} + \cdots + p^{i-1} - p^i \\ &= 1 - p^i\end{aligned}$$

故有 $\sum_{j=0}^{i-1} D_j p^{j-i} \le 1 - p^i < 1$。（其实这里用 $p$ 进制数更好理解，因为 $(D_{i-1}D_{i-2}\cdots D_1 D_0)_p < (1\underbrace{00\cdots 00}_{i-1\text{个零}})_p$ ）

又 $\sum_{j=i}^{q} D_j p^{j-i}$ 一定是个整数，记 $\alpha = \lfloor \log_p(N) \rfloor$，故：

$$
\begin{aligned}
\nu_p(N!) &= \sum_{i=1}^{\alpha}\left\lfloor\frac{\sum_{j=0}^{q} D_j p^j}{p^i}\right\rfloor \\
&= \sum_{i=1}^{\alpha}\sum_{j=i}^{q} D_j p^{j-i} \\
&= \sum_{j=1}^{q}\left(D_j\sum_{i=0}^{j-1} p^i\right) \\
&= \sum_{j=1}^{q} D_j\frac{p^j-1}{p-1}
\end{aligned}
$$

直接计算即可，预计得分 100 分。

**形形色色的部分分：**

**子任务 1、2、3（48 分）：** 推到蓝色式子推不下去了。$\alpha$ 虽大，但是大于 $q$ 的 $i$ 对答案没有任何贡献。于是只要从 $1$ 至 $q$ 枚举 $i$ 即可。双重循环 $O(q^2\log q)$。如果你写光速幂的话就是 $O(q^2+\sqrt{q})$。

**子任务 1、2、3、4（72 分）：** 注意计算过程中可能会出现 `long long` 溢出。如果没有写两个 `long long` 的模乘只会得到 72 分。

**子任务 5（17 分）：**

回到上一步：

$$
\nu_p(N!) = \sum_{j=1}^{q}\left(D_j\sum_{i=0}^{j-1} p^i\right)
$$

$n=p^5$ 时，在 $\mod n$ 意义下，对于 $j>5$，$p^j \mod n = 0$。

故：

$$
\nu_p(N!) = \sum_{j=1}^{q}\left(D_j\sum_{i=0}^{\min(j-1,5)} p^i\right)
$$

这样可以少算点东西提升程序效率，PE 原题也是 $p=61, n=61^{10}$。

你也可以通过这个性质，通过特殊处理避免掉两个 `long long` 的模乘。预计得分 17 分，加上前面四个子任务就是 89 分。

## 断电源（power）

- 来源：省中 2017 年模拟赛。原出题人：周润龙。
- 单调队列优化动态规划
- 赛后提交地址：[https://www.luogu.com.cn/problem/U149148](https://www.luogu.com.cn/problem/U149148)

**解法零**

输出样例、或全部输出 `Bad Luck`

以上解法都是 0 分。

**解法一**

暴力枚举每个拉杆是否拉下。

时间复杂度 $O(2^n)$，预计得分 17 分。

**解法二**

对问题进行形式化描述，不难发现本问题等价于：

寻找合适的 $n$ 元 0-1 数组 $(x_1, x_2, \cdots, x_n)$，使得：

$$\begin{cases} \sum_{i=1}^n r_i x_i \le P \\ \sum_{i=1}^n p_i x_i \ge P \end{cases}$$

且使得 $w = \sum_{i=1}^n h_i x_i$ 最小。

考虑 DP。

设 $f(i, l, r)$ 为 $x_1, \cdots, x_i$ 已经确定，$\sum_{j<i} r_j x_j = l$，$\sum_{j<i} p_j x_j = r$ 的情况下，$w$ 的最小值。

不难得出转移方程：

$$f(i, l, r) = \min \begin{cases} f(i-1, l, r) \\ f(i-1, l-r_i, r-p_i) + h_i \quad (l \ge r_i, r \ge p_i) \end{cases}$$

边界：

$$\begin{cases} f(0,0,0) = 0 \\ f(0, j, k) = +\infty \quad (k \ge j, k > 1) \end{cases}$$

答案即为 $\min_{0 \le i \le P, j \ge P}(f(n, i, j))$。

时间复杂度 $O(nP^2)$，空间复杂度 $O(nP^2)$，使用滚动数组可以优化至 $O(P^2)$。预计得分 48 分。

**解法三**

这一部分分有一个非常无耻的通过方法：在解法一的基础上加上两个剪枝：（1）当前选的 $h_i$ 之和大于已经搜到的最优解（2）按照目前的方案不管后面怎么选都不符合要求。~~应该是能卡的，但是我实在卡不掉，就放它过了... 注：其实如果子任务 2 的数据弱的话这个方法能过掉子任务 2，但是我特意构造了子任务 2 的最后一个测试点。~~ 这是子任务 3 只有 19 分的原因之一。

首先各种普通贪心应该都卡掉了。如果有贪心过了，请联系出题人。

不难看出，当 $\forall i \in [1, n], h_i = 0$ 时，只要存在 0-1 数列满足题目条件，答案一定是 $0$。问题就转化为判断是否存在 0-1 数列满足题目所述条件。

同样考虑 DP。设 $g(i, l)$ 为 $x_1, \cdots, x_i$ 已经确定，$\sum_{j<i} r_j x_j = l$ 的情况下，$r = \sum_{j<i} p_j x_j$ 的最大值。

$$g(i,l)=\max\begin{cases}g(i-1,l)\\ g(i-1,l-r_i)+p_i & l\ge r_i\end{cases}$$

边界：

$$g(0,0)=0$$

时间复杂度 $O(nP)$，预计得分 19 分，结合做法二可得 67 分。

**解法四**

解法二中的 DP 明显复杂度太高，需要优化。但是 $i$ 这一维时间明显约不掉，所以这意味着只能优化掉 $l$ 或者 $r$ 一维。

设 $h(i,s)$ 为 $x_1,\cdots,x_i$ 已经确定，$\sum_{j<i} r_j x_j \le s \le \sum_{j<i} p_j x_j$ 的情况下，$w$ 的最小值。

那么可以得出状态转移方程：

$h(i,s)=\min(h(i-1,s),\min_{s-p_i\le t\le s-r_i}(h(i-1,t))+h_i)$

很明显，这玩意可以用单调队列优化搞过去。时间复杂度 $O(nP)$，空间复杂度 $O(nP)$，使用滚动数组可优化至 $O(P)$。预计得分 100 分。

那如何证明这种 DP 的正确性？

对于状态 $L\le x\le R$，去掉 $i$ 位置的状态后变为 $L-r_i\le y\le R-p_i$。

对于任何状态 $L-r_i\le y\le R-p_i$，它是否都可以转化到 $L\le x\le R$？由于转移时 $x$ 是确定的，正确性是显然的。（如果你想不通，请注意对于任意 $i$ 均有 $r_i\le p_i$。）