

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

构造

吴清月

2020/3/9

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

写在前面

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

写在前面

众所周知，构造题几乎毫无套路可言，也几乎分不出什么知识点。

写在前面

众所周知，构造题几乎毫无套路可言，也几乎分不出什么知识点。

所以没法分模块讲，只好硬扯。

写在前面

众所周知，构造题几乎毫无套路可言，也几乎分不出什么知识点。

所以没法分模块讲，只好硬扯。

还是众所周知，构造题从来不好判断难度，因为不知道会不会降智。

写在前面

众所周知，构造题几乎毫无套路可言，也几乎分不出什么知识点。

所以没法分模块讲，只好硬扯。

还是众所周知，构造题从来不好判断难度，因为不知道会不会降智。

所以一千个人眼中有一千个难度，和难度标签可能没啥特别大的关系。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

写在前面

众所周知，构造题几乎毫无套路可言，也几乎分不出什么知识点。

所以没法分模块讲，只好硬扯。

还是众所周知，构造题从来不好判断难度，因为不知道会不会降智。

所以一千个人眼中有一千个难度，和难度标签可能没啥特别大的关系。

仍然是众所周知，CF 构造题多。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

写在前面

众所周知，构造题几乎毫无套路可言，也几乎分不出什么知识点。

所以没法分模块讲，只好硬扯。

还是众所周知，构造题从来不好判断难度，因为不知道会不会降智。

所以一千个人眼中有一千个难度，和难度标签可能没啥特别大的关系。

仍然是众所周知，CF 构造题多。

所以很多题是 ~~CF 近期的比赛题~~，做过原题的同学请不要秒的太快。（雾

写在前面

众所周知，构造题几乎毫无套路可言，也几乎分不出什么知识点。

所以没法分模块讲，只好硬扯。

还是众所周知，构造题从来不好判断难度，因为不知道会不会降智。

所以一千个人眼中有一千个难度，和难度标签可能没啥特别大的关系。

仍然是众所周知，CF 构造题多。

所以很多题是 ~~CF 近期的比赛题~~，做过原题的同学请不要秒的太快。（雾

课件中涉及到的所有例题都会下发参考程序。

Very Easy

●○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○○
○○○
○○○
○○

总结

○○○
○

AtCoder agc030C

Coloring Torus

Coloring Torus

对于一个 $n \times n$ 的网格，称 (r, c) 为处于第 $r+1$ 行，第 $c+1$ 列的格子。一个对网格的好的 k 染色是满足下述条件的染色方案：

- 每个格子被染成 $1, 2, 3, \dots, k$ 这 k 种颜色之一。
- k 种中的每种颜色都被至少一个格子使用。
- 对于任意颜色 $i, j (1 \leq i \leq k, 1 \leq j \leq k)$ ，满足所有颜色为 i 的格子，其相邻的颜色 j 的格子个数相同。这里，与 (r, c) 相邻的格子为 $((r-1) \bmod n, c)$ ， $((r+1) \bmod n, c)$ ， $(r, (c-1) \bmod n)$ ， $(r, (c+1) \bmod n)$ （如果一个格子多次出现，那么会算多次）。

给出 $k (1 \leq k \leq 1000)$ ，选择一个 $1 \leq n \leq 500$ ，任意构造一个对 $n \times n$ 的网格的好的 k 染色。

Very Easy

●○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○○
○○○
○○○
○○

总结

○○○
○

AtCoder agc030C

小提示

尝试构造一下 $k = 4, n = 4$ 。

Very Easy

●○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○○
○○○
○○○
○○

总结

○○○
○

AtCoder agc030C

小提示

尝试构造一下 $k = 4, n = 4$ 。

尝试构造一下 $k = 8, n = 4$ 。

尝试构造一下 $k = 7, n = 4$ 。

Very Easy

●○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○○
○○○
○○○
○○

总结

○○○
○

AtCoder agc030C

小提示

尝试构造一下 $k = 4, n = 4$ 。

尝试构造一下 $k = 8, n = 4$ 。

尝试构造一下 $k = 7, n = 4$ 。

会做了吗？

大提示

方案其实有很多，但是下面这一种比较妙：

1	2	3	4
2	3	4	1
3	4	1	2
4	1	2	3

1	2	3	4
6	7	8	5
3	4	1	2
8	5	6	7

大提示

方案其实有很多，但是下面这一种比较妙：

1	2	3	4
2	3	4	1
3	4	1	2
4	1	2	3

1	2	3	4
6	7	8	5
3	4	1	2
8	5	6	7

扩展到 $k = 7, n = 4$ 也是可以的：

1	2	3	4
6	7	4	5
3	4	1	2
4	5	6	7

Very Easy

```

○○●
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

AtCoder agc030C

题解

对于 $k = n$ 的情况，我们构造 $a_{i,j} = (i + j - 2) \bmod n + 1$ 。

题解

对于 $k = n$ 的情况，我们构造 $a_{i,j} = (i + j - 2) \bmod n + 1$ 。

对于 $k = 2n$ 的情况，我们对奇偶进行分类讨论：

- 奇数行： $a_{i,j} = (i + j - 2) \bmod n + 1$
- 偶数行： $a_{i,j} = (i + j - 2) \bmod n + n + 1$

题解

对于 $k = n$ 的情况，我们构造 $a_{i,j} = (i + j - 2) \bmod n + 1$ 。

对于 $k = 2n$ 的情况，我们对奇偶进行分类讨论：

- 奇数行： $a_{i,j} = (i + j - 2) \bmod n + 1$
- 偶数行： $a_{i,j} = (i + j - 2) \bmod n + n + 1$

对于 $n < k \leq 2n$ 的情况，直接在 $k = 2n$ 的基础上把所有大于 k 的 x 替换成 $x - n$ 即可。

题解

对于 $k = n$ 的情况，我们构造 $a_{i,j} = (i + j - 2) \bmod n + 1$ 。

对于 $k = 2n$ 的情况，我们对奇偶进行分类讨论：

- 奇数行： $a_{i,j} = (i + j - 2) \bmod n + 1$
- 偶数行： $a_{i,j} = (i + j - 2) \bmod n + n + 1$

对于 $n < k \leq 2n$ 的情况，直接在 $k = 2n$ 的基础上把所有大于 k 的 x 替换成 $x - n$ 即可。

所以接下来的问题就是找到一个 n 满足 $n \leq k \leq 2n$ 。

令 $n = 2 \lceil \frac{k}{4} \rceil$ 即可。特判 $k = 1$ 。

代码长度 100b。

A 4x4 grid of circles. The top row has 4 circles, the second row has 3 circles, the third row has 3 circles, and the bottom row has 2 circles. The circle at the top-right position (row 1, column 4) is missing.

○ ○ ○
○

Robot Arms

Robot Arms

一台机械臂由 m 段和 $m + 1$ 个关节构成，第 i 段的长度为一个整数 d_i ，段与段之间由关节连接。其中第一个关节位于 $(0, 0)$ ，机械臂运行的过程中所有的段都必须水平或者竖直。

有 n 个目标点，第 i 个目标点位于 (x_i, y_i) 。

构造一台最多 40 段的机械臂，满足机械臂的最后一个关节能够到达所有的目标点，并且输出方案。

$1 \leq n \leq 1000, -10^9 \leq x_i, y_i \leq 10^9$ ，你需要保证构造的 $1 \leq d_i \leq 10^{12}$ 。

Very Easy

○○○○
○●
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

AtCoder arc103D

题解

首先 $x_i + y_i$ 的奇偶性必须都相同，否则不合法。

题解

首先 $x_i + y_i$ 的奇偶性必须都相同，否则不合法。

先令 $m = 40$ ，接下来我们构造 $d_i = 2^{m-i}$ 。容易发现这时能够到达的点一定满足 $|x| + |y| \leq 2^m - 1$ 。

题解

首先 $x_i + y_i$ 的奇偶性必须都相同，否则不合法。

先令 $m = 40$ ，接下来我们构造 $d_i = 2^{m-i}$ 。容易发现这时能够到达的点一定满足 $|x| + |y| \leq 2^m - 1$ 。

注意到，所有满足上述性质且奇偶性正确的点都可以被到达，证明可以用数学归纳法。

如果奇偶性不相同，就先在开头添加一段长度为 1 的段。

Very Easy

○○○○
○○
●○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 1227G

Not Same

Not Same

大小为 n 的集合 a_1, a_2, \dots, a_n , 满足 $1 \leq a_i \leq n$ 。

每次你可以选择一个子集的元素并且把这个子集的元素都 -1 , 你需要在至多 $n+1$ 次操作之内把所有数变成 0。

每次选择的子集不能相同。

$1 \leq n \leq 1000$

Very Easy

○○○○
○○
○○●○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

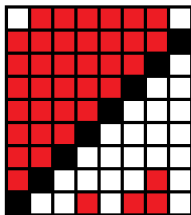
○○○
○

CodeForces 1227G

题解

我们把操作描述为一个宽度为 n 高度为 $n+1$ 的 01 矩阵 c , 其中第 i 行第 j 列表示第 i 次操作中 a_j 是否被选中。

先按照降序排序，然后我们这样填充：


$$a = \{8, 8, 7, 7, 5, 5, 5, 1\}$$

接下来我们证明这种方案一定合法。

Very Easy

○○○○
○○
○○●
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 1227G

证明

Proof.

证明

Proof.

首先，无论 a 数组如何，一定都能通过这种方式变成全 0。

证明

Proof.

首先，无论 a 数组如何，一定都能通过这种方式变成全 0。
如果存在 $1 \leq i < j \leq n + 1$ ，满足第 i 行和第 j 行相同，接下来分类讨论：

证明

Proof.

首先，无论 a 数组如何，一定都能通过这种方式变成全 0。
 如果存在 $1 \leq i < j \leq n+1$ ，满足第 i 行和第 j 行相同，接下来分类讨论：

若 $i = 1$ ，则

$$a_1 \leq n \implies c_{1,1} = 0 \implies c_{j,1} = 0 \implies a_1 \leq n+1-j < n$$

$$c_{1,2} = c_{j,2} \implies a_2 \leq n-j \text{ 或 } a_2 \geq n \implies a_2 \leq n-j < n-1$$

$$\dots$$

$$c_{j,n+2-j} = c_{1,n+2-j} \implies a_{n+2-j} \leq 0, \text{ 矛盾。}$$

证明

Proof.

首先，无论 a 数组如何，一定都能通过这种方式变成全 0。
 如果存在 $1 \leq i < j \leq n+1$ ，满足第 i 行和第 j 行相同，接下来分类讨论：

若 $i = 1$ ，则

$$a_1 \leq n \implies c_{1,1} = 0 \implies c_{j,1} = 0 \implies a_1 \leq n+1-j < n$$

$$c_{1,2} = c_{j,2} \implies a_2 \leq n-j \text{ 或 } a_2 \geq n \implies a_2 \leq n-j < n-1$$

$$\dots$$

$$c_{j,n+2-j} = c_{1,n+2-j} \implies a_{n+2-j} \leq 0, \text{ 矛盾。}$$

否则，先抠掉第 1 行，第 i 行和第 j 行不可能都是全 1，选择一个 0 的位置，同理可证。 □

A 4x4 grid of circles. The top row has 4 circles, the second row has 3 circles, the third row has 3 circles, and the bottom row has 2 circles. The circle at the top-right position (row 1, column 4) is missing.

○ ○ ○
○

Inverse of Rows and Columns

Inverse of Rows and Columns

给定一个 $n \times m$ 的 01 矩阵，你可以选择任意行翻转任意列翻转。

求最后能否将整个矩阵变成有序的。

所谓有序就是满足 $a_{1,1} \leq a_{1,2} \leq \dots \leq a_{1,m} \leq a_{2,1} \leq a_{2,2} \leq \dots \leq a_{2,m} \leq \dots \leq a_{n,m-1} \leq a_{n,m}$

$1 \leq n, m \leq 200$

Very Easy

○○○○
○○
○○○
○●

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 1157G

题解

首先特判 $n = 1$ 。

题解

首先特判 $n = 1$ 。

注意到，最终的矩阵要么满足第一行全 0 要么满足最后一行全 1，并且一旦确定了一行的情况，所有的列怎么翻转就是确定的了。

题解

首先特判 $n = 1$ 。

注意到，最终的矩阵要么满足第一行全 0 要么满足最后一行全 1，并且一旦确定了一行的情况，所有的列怎么翻转就是确定的了。

接下来问题就是确定每一行怎么翻转。这个非常简单，只需要让前面一半全 0，后面一半全 1，中间一行先一段 0 后一段 1 即可。

题解

首先特判 $n = 1$ 。

注意到，最终的矩阵要么满足第一行全 0 要么满足最后一行全 1，并且一旦确定了一行的情况，所有的列怎么翻转就是确定的了。

接下来问题就是确定每一行怎么翻转。这个非常简单，只需要让前面一半全 0，后面一半全 1，中间一行先一段 0 后一段 1 即可。

时间复杂度 $O(n^3)$ 或者 $O(n^2)$ 。

也不难啊，为啥难度是 2800 呢？

A 4x4 grid of circles. The top row has 4 circles, the second row has 2 circles, the third row has 3 circles, and the bottom row has 2 circles. The circle at the top-right position (row 1, column 4) is missing.



Construct the Binary Tree

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

●○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○○
○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

CodeForces 1311E

Construct the Binary Tree

给定 $d \leq 5000$ ，构造一个 n 个点的二叉树，满足所有节点的深度之和等于 d 。

1000 组询问。

Very Easy

○○○○
○○
○○○
○○

Easy

●●
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 1311E

题解

首先构造一棵完全二叉树，令 $f_i = \lfloor \frac{i}{2} \rfloor$ 。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

●●
○○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○○
○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

CodeForces 1311E

题解

首先构造一棵完全二叉树，令 $f_i = \lfloor \frac{i}{2} \rfloor$ 。

然后随便找出来一条链，从后到前枚举不在链上的点，每次让父节点深度 +1。

题解

首先构造一棵完全二叉树，令 $f_i = \lfloor \frac{i}{2} \rfloor$ 。

然后随便找出来一条链，从后到前枚举不在链上的点，每次让父节点深度 +1。

每次总深度会 +1，所以一定能构造出解。

时间复杂度 $O(Tn)$ 。

Very Easy

○○○○
○○
○○○
○○

Easy

○○
●○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 1305E

Kuroni and the Score Distribution

Kuroni and the Score Distribution

新鲜热乎的例题。

对于一个序列 $1 \leq a_1 < a_2 < \cdots < a_n \leq 10^9$ ，我们定义三元组 (i, j, k) 是优美的，当且仅当 $a_i + a_j = a_k$ 。

给定 n 和 m ，构造一个长度为 n 的有 m 个优美的三元组的序列。

$$1 \leq n \leq 5000, 0 \leq m \leq 10^9$$

Very Easy

```

oooo
oo
ooo
oo

```

Easy

```

oo
o●
oo
ooo

```

Medium

```

oooo
ooo
ooo
ooo
oo

```

Not Hard

```

ooo
ooo
oo
ooo

```

Hard

```

oooooo
ooo
ooo
oo

```

总结

```

ooo
o

```

CodeForces 1305E

题解

最大元素为 a_i 的优美的三元组最多有 $\lfloor \frac{i-1}{2} \rfloor$ 个。构造方法可以令 $a_i = i$ 。

Very Easy

```

oooo
oo
ooo
oo

```

Easy

```

oo
o●
oo
ooo

```

Medium

```

oooo
ooo
ooo
ooo
oo

```

Not Hard

```

ooo
ooo
oo
oo
ooo

```

Hard

```

oooooo
ooo
ooo
ooo
oo

```

总结

```

ooo
o

```

CodeForces 1305E

题解

最大元素为 a_i 的优美的三元组最多有 $\lfloor \frac{i-1}{2} \rfloor$ 个。构造方法可以令 $a_i = i$ 。

取 k 满足 $\sum_{i=1}^k \lfloor \frac{i-1}{2} \rfloor \leq m$ 且 $\sum_{i=1}^{k+1} \lfloor \frac{i-1}{2} \rfloor > m$ 。

题解

最大元素为 a_i 的优美的三元组最多有 $\lfloor \frac{i-1}{2} \rfloor$ 个。构造方法可以令 $a_i = i$ 。

取 k 满足 $\sum_{i=1}^k \lfloor \frac{i-1}{2} \rfloor \leq m$ 且 $\sum_{i=1}^{k+1} \lfloor \frac{i-1}{2} \rfloor > m$ 。

设 $d = m - \sum_{i=1}^k \lfloor \frac{i-1}{2} \rfloor$ ，则

$$a_i = \begin{cases} i & i \leq k \\ a_k + a_{k-2d+1} & i = k+1 \\ \infty & i > k+1 \end{cases}$$

题解

最大元素为 a_i 的优美的三元组最多有 $\lfloor \frac{i-1}{2} \rfloor$ 个。构造方法可以令 $a_i = i$ 。

取 k 满足 $\sum_{i=1}^k \lfloor \frac{i-1}{2} \rfloor \leq m$ 且 $\sum_{i=1}^{k+1} \lfloor \frac{i-1}{2} \rfloor > m$ 。

设 $d = m - \sum_{i=1}^k \lfloor \frac{i-1}{2} \rfloor$ ，则

$$a_i = \begin{cases} i & i \leq k \\ a_k + a_{k-2d+1} & i = k+1 \\ \infty & i > k+1 \end{cases}$$

注意 ∞ 的取值需要满足不能形成三元组。

时间复杂度 $O(n)$ 。

A 4x4 grid of circles. The top-left circle (row 1, column 1) contains a black dot. All other circles are empty.

○ ○ ○
○

AtCoder agc027D

Modulo Matrix

Modulo Matrix

构造一个 $n \times n$ 的矩阵 a ，满足如下性质：

- $1 \leq a_{i,j} \leq 10^{15}$ 。
- 所有的 $a_{i,j}$ 互不相同。
- 存在一个正整数 m ，满足对于任意两个行相邻或列相邻的元素 x, y ，有 $\max(x, y) \bmod \min(x, y)$ 都是 m 。

$2 \leq n \leq 500$ ，保证有解。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
o●
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

AtCoder agc027D

题解

考虑构造 $m = 1$ 的方案。

Very Easy

```

○○○○
○○
○○○
○○○
○○

```

Easy

```

○○
○○
○○
○●
○○○

```

Medium

```

○○○○
○○○
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○○
○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

AtCoder agc027D

题解

考虑构造 $m = 1$ 的方案。

将整个矩阵黑白染色，令黑色格子中的数大于周围白色格子中的数。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○●
○○○

```

Medium

```

○○○○
○○○
○○○
○○○

```

Not Hard

```

○○○
○○○
○○○
○○○

```

Hard

```

○○○○○○○
○○○
○○○
○○○
○○○

```

总结

```

○○○
○

```

AtCoder agc027D

题解

考虑构造 $m = 1$ 的方案。

将整个矩阵黑白染色，令黑色格子中的数大于周围白色格子中的数。

假设我们已经确定了所有白色格子中的数，那么对于一个黑格子 (x, y) ， $a_{x,y} = \text{lcm}(a_{x1,y}, a_{x+1,y}, a_{x,y1}, a_{x,y+1}) + 1$ 。

接下来唯一的问题就是使得相邻四个数的 lcm 不超过 10^{15} 。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○●
○○○

```

Medium

```

○○○○
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○
○○

```

Hard

```

○○○○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

AtCoder agc027D

题解

考虑构造 $m = 1$ 的方案。

将整个矩阵黑白染色，令黑色格子中的数大于周围白色格子中的数。

假设我们已经确定了所有白色格子中的数，那么对于一个黑格子 (x, y) ， $a_{x,y} = \text{lcm}(a_{x1,y}, a_{x+1,y}, a_{x,y1}, a_{x,y+1}) + 1$ 。

接下来唯一的问题就是使得相邻四个数的 lcm 不超过 10^{15} 。

筛出前 $2n$ 个质数，为白色格子的每一个主对角线和副对角线分配一个质数，白色格子中的数等于两个对角线上质数的乘积。

打表会发现这样整个矩阵的最大值在 4×10^{14} 左右，可以通过本题。



○ ○ ○
○ ○ ○
○ ○
○ ○ ○

○ ○ ○
○

AtCoder agc035C

Skolem XOR Tree

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
●○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

AtCoder agc035C

Skolem XOR Tree

给定 n ，构造一棵 $2n$ 个节点的树，其中 $v_i = v_{n+i} = i$ ，要求 i 和 $i+n$ 路径上所有点权的异或和恰好是 i （包含端点）。

$$1 \leq n \leq 10^5$$

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
●○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

AtCoder agc035C

题解

我们用 \oplus 代表异或运算。

题解

我们用 \oplus 代表异或运算。

注意到，当 $k \geq 2$ 时， $0 \oplus 1 \oplus 2 \oplus \dots \oplus 2^k - 1 = 0$ 。

所以当 $n = 2^k - 1$ 时，从 1 到 $2n$ 排成一条链就好了。

题解

我们用 \oplus 代表异或运算。

注意到，当 $k \geq 2$ 时， $0 \oplus 1 \oplus 2 \oplus \dots \oplus 2^k - 1 = 0$ 。

所以当 $n = 2^k - 1$ 时，从 1 到 $2n$ 排成一条链就好了。

当 $n = 2^k$ 时，除了 n 和 $2n$ 之外所有点的权值二进制第 k 位都是 0，所以无解。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
oo●

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

AtCoder agc035C

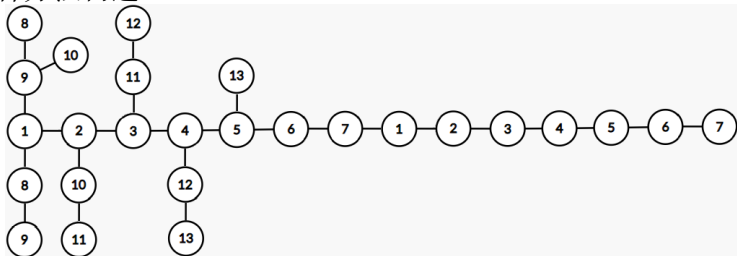
题解

否则，注意到 $(x + 2^k) \oplus x + 1 \oplus x = x + 1 + 2^k$ 。

题解

否则，注意到 $(x + 2^k) \oplus x + 1 \oplus x = x + 1 + 2^k$ 。

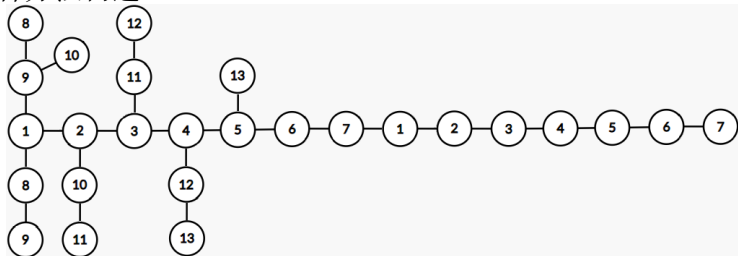
令 k 为最大的满足 $2^k \leq n$ 的数。以 $n = 13$ 为例，我们按照这种方法构造：



题解

否则，注意到 $(x + 2^k) \oplus x + 1 \oplus x = x + 1 + 2^k$ 。

令 k 为最大的满足 $2^k \leq n$ 的数。以 $n = 13$ 为例，我们按照这种方法构造：



顺着构造就行了。时间复杂度 $O(n)$ 。

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

●○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 611H

New Year and Forgotten Tree

New Year and Forgotten Tree

给你一棵 n 个点的树，第 i 条边连接了 u_i 和 v_i ，只不过除了 n 以外，所有的数位都被替换成了'?'，也就是你只知道 u_i 和 v_i 的位数。

求一棵合法的树。

$$1 \leq n \leq 2 \times 10^5$$

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○●○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 611H

性质

两个点位数如果相同那就没有区别。同时我们注意到位数很小，最大只有 6。

性质

两个点位数如果相同那就没有区别。同时我们注意到位数很小，最大只有 6。

对于每一位，我们选择一个关键点，为了方便直接选取 10 的整数次幂：1, 10, 100, 1000, 10000, 100000。

性质

两个点位数如果相同那就没有区别。同时我们注意到位数很小，最大只有 6。

对于每一位，我们选择一个关键点，为了方便直接选取 10 的整数次幂：1, 10, 100, 1000, 10000, 100000。

接下来我们证明存在一组解，满足任意一条边连接的两个点中都至少有一个是关键点。

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○●○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 611H

证明

Proof.

证明

Proof.

若存在 (u, v) 满足 u, v 均不是关键点，则我们先断开 (u, v) 这条边。这样整棵树被分成了两个连通块。

证明

Proof.

若存在 (u, v) 满足 u, v 均不是关键点，则我们先断开 (u, v) 这条边。这样整棵树被分成了两个连通块。

设和 u 位数相等的关键点是 x ，和 v 位数相等的关键点是 y ，则下面三种情况必居其一：

- x 和 v 不在一个连通块内。连接 (x, v) 。
- y 和 u 不在一个连通块内。连接 (y, u) 。
- 否则， x 和 y 必定不在同一个连通块内。连接 (x, y) 。



Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○●
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 611H

题解

6 个关键点两两之间有 5 条路径，这个可以直接暴搜。
暴搜结束后，剩下的所有边都一定连接关键点和非关键点。

题解

6 个关键点两两之间有 5 条路径，这个可以直接暴搜。

暴搜结束后，剩下的所有边都一定连接关键点和非关键点。

跑一个网络流，左边的点代表每一个位数，右边的点代表两个位数之间边的数量。

(方法不唯一，欢迎吐槽)

○ ○
○ ○
○ ○
○ ○ ○

A 4x4 grid of circles. The second circle in the second row is filled black, while all other circles are white with black outlines.

○ ○ ○
○

AtCoder agc025D

Choosing Points

Choosing Points

给定 n, D_1, D_2 , 选择 n^2 个横纵坐标在 $[0, 2n)$ 之间的点, 满足任意两个点之间的欧几里得距离既不是 $\sqrt{D_1}$ 也不是 $\sqrt{D_2}$ 。

$$1 \leq N \leq 300, 1 \leq D_1, D_2 \leq 2 \times 10^5$$

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○●○
○○○
○○

```

Not Hard

```

○○○
○○○
○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

AtCoder agc025D

性质

对于一个 d ，我们把距离为 \sqrt{d} 的点两两相连，则一定不存在奇环。

性质

对于一个 d ，我们把距离为 \sqrt{d} 的点两两相连，则一定不存在奇环。

Proof.

假设两个点 (x_1, y_1) 和 (x_2, y_2) 之间有边，则
 $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} = d$ ，也就是 $\Delta x^2 + \Delta y^2 = d^2$ 。

性质

对于一个 d ，我们把距离为 \sqrt{d} 的点两两相连，则一定不存在奇环。

Proof.

假设两个点 (x_1, y_1) 和 (x_2, y_2) 之间有边，则
 $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} = d$ ，也就是 $\Delta x^2 + \Delta y^2 = d^2$ 。
 接下来分类讨论：

- 若 d 是奇数，则 $\Delta x + \Delta y$ 是奇数，不存在奇环。
- 若 d 是偶数，则整个点集按照横纵坐标的奇偶性可以被分成互不相交的四部分，递归证明。



题解

先把所有点拿出来，把所有距离为 $\sqrt{D_1}$ 的点两两相连。

由上可知，连出来的图一定是二分图，所以我们可以 01 染色，取较多的一半。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○○●
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○
○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

AtCoder agc025D

题解

先把所有点拿出来，把所有距离为 $\sqrt{D_1}$ 的点两两相连。

由上可知，连出来的图一定是二分图，所以我们可以 01 染色，取较多的一半。

再在这一部分中把所有距离为 $\sqrt{D_2}$ 的点两两相连，染色后再取较多的一半。

题解

先把所有点拿出来，把所有距离为 $\sqrt{D_1}$ 的点两两相连。

由上可知，连出来的图一定是二分图，所以我们可以 01 染色，取较多的一半。

再在这一部分中把所有距离为 $\sqrt{D_2}$ 的点两两相连，染色后再取较多的一半。

则最后的点集大小一定大于等于 $\frac{1}{4} \cdot (2n)^2$ ，即 n^2 。

在连边的时候枚举横坐标，时间复杂度 $O(n^3)$ 。

A 4x4 grid of circles. The top row has 4 circles, the second row has 2 circles, the third row has 3 circles, and the bottom row has 2 circles. The circle at the top-right position (row 1, column 4) is missing.

A 4x3 grid of circles. The bottom-right circle is missing, leaving 11 circles in total.



○ ○ ○
○

Sorting a Grid

Sorting a Grid

$n \times m$ 的网格 A ，格子中的数形成了一个 1 到 $n \times m$ 的排列，你需要顺次进行下面三次操作：

- 1 重排所有的行。
- 2 重排所有的列。
- 3 重排所有的行。

最后的目标是将整个网格排序。输出第一次操作后的网格和第二次操作后的网格。

$$1 \leq n, m \leq 100$$

性质

考虑最后一次重排。这一次操作一定是把所有行排序。换句话说在此之前我们只需要关心元素在哪一行。为了方便令

$$A_{i,j} = \left\lfloor \frac{A_{i,j-1}}{m} \right\rfloor + 1。$$

性质

考虑最后一次重排。这一次操作一定是把所有行排序。换句话说在此之前我们只需要关心元素在哪一行。为了方便令

$$A_{i,j} = \left\lfloor \frac{A_{i,j-1}}{m} \right\rfloor + 1.$$

接下来考虑第二次重排。这一次一定是把所有的列排序，并且满足排完序之后恰好是 1 到 n 。容易发现这要求重排之前所有的列都是 1 到 n 的一个排列。

性质

考虑最后一次重排。这一次操作一定是把所有行排序。换句话说在此之前我们只需要关心元素在哪一行。为了方便令

$$A_{i,j} = \left\lfloor \frac{A_{i,j-1}}{m} \right\rfloor + 1.$$

接下来考虑第二次重排。这一次一定是把所有的列排序，并且满足排完序之后恰好是 1 到 n 。容易发现这要求重排之前所有的列都是 1 到 n 的一个排列。

现在我们只需要考虑第一次重排就好了。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
oo●
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

AtCoder agc037D

题解

先给出一个结论：在题目所给条件下（1 到 n 每个数都有 m 个），第一次重排一定存在合法解。这个我们在构造过程中递归证明。

Very Easy

```

0000
00
00
000
00

```

Easy

```

00
00
00
000

```

Medium

```

0000
000
000
00●
00

```

Not Hard

```

000
000
00
00
000

```

Hard

```

000000
000
000
000
00

```

总结

```

000
0

```

AtCoder agc037D

题解

先给出一个结论：在题目所给条件下（1 到 n 每个数都有 m 个），第一次重排一定存在合法解。这个我们在构造过程中递归证明。

考虑第一列的结果。相当于让你在每一行剩下的元素中选择一个数，使得 1 到 n 各出现一次。

用二分图匹配，左边是每一行，右边是每一个数，左边第 i 行和右边的第 j 个数之间有 $cnt_{i,j}$ 条边。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○●
○○

```

Not Hard

```

○○○
○○○
○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○○

```

总结

```

○○○
○

```

AtCoder agc037D

题解

先给出一个结论：在题目所给条件下（1 到 n 每个数都有 m 个），第一次重排一定存在合法解。这个我们在构造过程中递归证明。

考虑第一列的结果。相当于让你在每一行剩下的元素中选择一个数，使得 1 到 n 各出现一次。

用二分图匹配，左边是每一行，右边是每一个数，左边第 i 行和右边的第 j 个数之间有 $cnt_{i,j}$ 条边。

由于任意 k 行中至少有 k 个互不相同的数，有霍尔定理可知存在完美匹配，能够构造出解。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○●
○○
○○

```

Not Hard

```

○○○
○○○
○○○
○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

AtCoder agc037D

题解

先给出一个结论：在题目所给条件下（1 到 n 每个数都有 m 个），第一次重排一定存在合法解。这个我们在构造过程中递归证明。

考虑第一列的结果。相当于让你在每一行剩下的元素中选择一个数，使得 1 到 n 各出现一次。

用二分图匹配，左边是每一行，右边是每一个数，左边第 i 行和右边的第 j 个数之间有 $cnt_{i,j}$ 条边。

由于任意 k 行中至少有 k 个互不相同的数，有霍尔定理可知存在完美匹配，能够构造出解。

用匈牙利算法，每次画 $O(n^2)$ 的时间寻找一组匹配，总复杂度 $O(n^2m)$ 。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
●o

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

CodeForces 1158D

Winding polygonal line

Winding polygonal line

平面上有 n 个点，不存在三点共线。还有一个长度为 $n - 2$ 的字符串 s ，满足 $s_i = \{'L', 'R'\}$ 。

你需要将这 n 个点连接成一条折线，满足每一个拐点向哪拐刚好符合给定的字符串 s 。

$$3 \leq n \leq 2000$$

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○●

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces 1158D

题解

首先，随便取一个凸包上的点当做起始点。

Very Easy

```

oooo
oo
ooo
oo

```

Easy

```

oo
oo
oo
ooo

```

Medium

```

oooo
ooo
ooo
ooo
o●

```

Not Hard

```

ooo
ooo
oo
oo
ooo

```

Hard

```

oooooo
ooo
ooo
ooo
oo

```

总结

```

ooo
o

```

CodeForces 1158D

题解

首先，随便取一个凸包上的点当做起始点。

每一次，求出剩下的所有点的凸包，如果下一步需要往左拐就找到凸包上最靠右的点，否则找到最靠左的点。

题解

首先，随便取一个凸包上的点当做起始点。

每一次，求出剩下的所有点的凸包，如果下一步需要往左拐就找到凸包上最靠右的点，否则找到最靠左的点。

注意到一旦我们选择了最靠左的点，那么接下来选择任何一个点一定都是向右拐的，靠右同理，所以构造出的方案一定合法。

时间复杂度 $O(n^2)$ 。

A 4x4 grid of circles. The top-left circle (row 1, column 1) contains a black dot. All other circles are empty.

○ ○ ○
○

Median Replace

Median Replace

我们定义一个长度为 n (n 是奇数) 的 01 串 s 是美丽的, 当且仅当它能够通过进行下面这个操作 $\frac{n-1}{2}$ 次变成 '1':

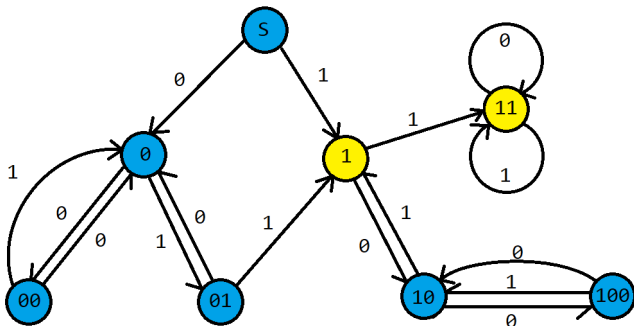
- 选择三个连续的位置 s_{i-1}, s_i, s_{i+1} , 并且用它们的中位数代替这三个字符。

现在给定一个长度为 n 的包含 '0', '1', '?' 三种字符的字符串, 求有多少种将 '?' 替换为 '0' 和 '1' 的方案, 使得最后得到的串是美丽的。对 $10^9 + 7$ 取模。

$$1 \leq n \leq 300000$$

结论

我们直接给出结论：把你的字符串放到下面这个自动机上跑一遍，能跑到终点就是合法的！



Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

oo●
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

agc022E

题解

虽然这道题看起来和构造没啥关系，但是像做出这道题，你的构造水平必须过硬。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

oo●
ooo
oo
ooo

Hard

oooooooo
ooo
ooo
oo

总结

ooo
o

agc022E

题解

虽然这道题看起来和构造没啥关系，但是像做出这道题，你的构造水平必须过硬。

关于上面那个自动机的构造，我们可以这么考虑：

题解

虽然这道题看起来和构造没啥关系，但是像做出这道题，你的构造水平必须过硬。

关于上面那个自动机的构造，我们可以这么考虑：

从开头开始一个一个加字符。那么如果在任意时刻开头出现了两个 1 那就一定合法（对应了 11 那个终点），而如果最后剩下一个 1，也一定合法（对应 1 那个终点）。

注意到，某些情况下删掉一些字符一定不会变劣。比如 000 删成 0，101 删成 1 等等，于是就有了上面那个自动机。

题解

虽然这道题看起来和构造没啥关系，但是像做出这道题，你的构造水平必须过硬。

关于上面那个自动机的构造，我们可以这么考虑：

从开头开始一个一个加字符。那么如果在任意时刻开头出现了两个 1 那就一定合法（对应了 11 那个终点），而如果最后剩下一个 1，也一定合法（对应 1 那个终点）。

注意到，某些情况下删掉一些字符一定不会变劣。比如 000 删成 0，101 删成 1 等等，于是就有了上面那个自动机。

设 $f_{i,j}$ 表示长度为 i ，到达 j 号点的方案数，直接转移就好了。

时间复杂度 $O(n)$ 。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
●oo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

AtCoder agc025E

Walking on a Tree

Walking on a Tree

有一棵 n 个点的树，还有 m 条路径，第 i 条路径是从 u_i 到 v_i 。

定义一条路径的愉悦度为满足如下两个条件之一的边的数量：

- 在之前的路径中，这条边未被经过。
- 在之前的路径中，这条边只被沿相反方向经过。

你需要把所有的 m 条路径重新定向，使得总愉悦度最大。
求一组方案。

$$1 \leq n, m \leq 2000$$

猜想

首先考虑理论最大值。容易发现被经过一次的边至多会带来 1 的贡献，被经过至少 2 次的边至多会带来 2 的贡献。

猜想

首先考虑理论最大值。容易发现被经过一次的边至多会带来 1 的贡献，被经过至少 2 次的边至多会带来 2 的贡献。

经过手玩找规律，我们大胆猜想这就是答案。接下来尝试在构造过程中证明。

Very Easy

```

oooo
oo
ooo
oo

```

Easy

```

oo
oo
oo
ooo

```

Medium

```

oooo
ooo
ooo
oo

```

Not Hard

```

ooo
oo●
oo
oo
ooo

```

Hard

```

oooooo
ooo
ooo
ooo
oo

```

总结

```

ooo
o

```

AtCoder agc025E

题解

考虑增量法，每次删掉树上的一个叶子 x ，并且确定以这个叶子为其中一个端点的所有路径：

题解

考虑增量法，每次删掉树上的一个叶子 x ，并且确定以这个叶子为其中一个端点的所有路径：

- 若这种路径数量为 0，直接删掉这个叶子。
- 若数量为 1，怎么定向答案都一样，将这条路径向内收缩一条边，然后删掉 x 。
- 否则，每次删掉两条路径。设两条路径分别为 $x-u$ 和 $x-v$ ，其中 u 和 v 在以 x 为根时的 LCA 为 d ，则将 $x-d$ 路径上所有边都标记为访问了两次，同时把原先的两条路径替换为 $u-v$ 这一条路径。

题解

考虑增量法，每次删掉树上的一个叶子 x ，并且确定以这个叶子为其中一个端点的所有路径：

- 若这种路径数量为 0，直接删掉这个叶子。
- 若数量为 1，怎么定向答案都一样，将这条路径向内收缩一条边，然后删掉 x 。
- 否则，每次删掉两条路径。设两条路径分别为 $x - u$ 和 $x - v$ ，其中 u 和 v 在以 x 为根时的 LCA 为 d ，则将 $x - d$ 路径上所有边都标记为访问了两次，同时把原先的两条路径替换为 $u - v$ 这一条路径。

在构造的过程中，所有的边都达到了理论最大值，所以理论最大值就是答案。

朴素实现，时间复杂度 $O(nm)$ 。

A 4x4 grid of circles. The top-left circle (row 1, column 1) contains a black dot. All other circles are empty.

A 4x4 grid of circles. The top row has 4 circles, the second row has 3 circles, the third row has 3 circles, and the bottom row has 2 circles. The circle at the top-right position (row 1, column 4) is missing.

○ ○ ○
○

Berserk Robot

Berserk Robot

有一个机器人从 $(0,0)$ 出发，每一秒会向上下左右四个方向中的一个移动 1 单位长度。机器人的移动方式是循环的，循环节长度为 l 。

你知道 n 条信息，第 i 条信息形如在 t_i 时刻机器人位于 (x_{t_i}, y_{t_i}) 。构造一个合法的移动方式。

$1 \leq n \leq 2 \times 10^5, 1 \leq l \leq 10^6, 1 \leq t_i \leq 10^{18}, -10^{18} \leq x_i, y_i \leq 10^{18}$

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
o●
ooo

Hard

oooooo
ooo
ooo
oo

总结

ooo
o

CodeForces 538G

题解

奇偶性不满足条件直接无解。

题解

奇偶性不满足条件直接无解。

假设 l 次移动后机器人位于 $(\Delta x, \Delta y)$ ，则第 k 次移动后机器人一定位于 $(x_{k \bmod l} + \lfloor \frac{k}{l} \rfloor \Delta x, y_{k \bmod l} + \lfloor \frac{k}{l} \rfloor \Delta y)$ 。

题解

奇偶性不满足条件直接无解。

假设 l 次移动后机器人位于 $(\Delta x, \Delta y)$ ，则第 k 次移动后机器人一定位于 $(x_{k \bmod l} + \lfloor \frac{k}{l} \rfloor \Delta x, y_{k \bmod l} + \lfloor \frac{k}{l} \rfloor \Delta y)$ 。

我们把所有的点按照 $t_i \bmod l$ 排序，令

$x'_{t_i} = x_{t_i \bmod l} - \lfloor \frac{t_i}{l} \rfloor \Delta x$ ， y'_{t_i} 同理，则对于相邻的两条信息，我们有：

$$\left| x'_{t_{i+1}} - x'_{t_i} \right| + \left| y'_{t_{i+1}} - y'_{t_i} \right| \leq t_{i+1} \bmod l - t_i \bmod l$$

题解

奇偶性不满足条件直接无解。

假设 l 次移动后机器人位于 $(\Delta x, \Delta y)$ ，则第 k 次移动后机器人一定位于 $(x_{k \bmod l} + \lfloor \frac{k}{l} \rfloor \Delta x, y_{k \bmod l} + \lfloor \frac{k}{l} \rfloor \Delta y)$ 。

我们把所有的点按照 $t_i \bmod l$ 排序，令

$x'_{t_i} = x_{t_i \bmod l} - \lfloor \frac{t_i}{l} \rfloor \Delta x$ ， y'_{t_i} 同理，则对于相邻的两条信息，我们有：

$$\left| x'_{t_{i+1}} - x'_{t_i} \right| + \left| y'_{t_{i+1}} - y'_{t_i} \right| \leq t_{i+1} \bmod l - t_i \bmod l$$

展开，化简，我们就可以得到关于 Δx 和 Δy 的一组限制。

最后在取值范围里面随便找一组 $\Delta x, \Delta y$ ，就可以顺着构造

A 4x4 grid of circles. The first three rows are complete with 4 circles each. The fourth row has 3 circles, with the last circle missing.

○ ○ ○
○

Excavator Contest

Excavator Contest

有一个 $n \times n$ 的网格，你需要从边界上的某个位置出发，经过所有的格子恰好一次并最终回到边界上，要求拐弯的数量至少有 $n(n-1)-1$ 次。

$$1 \leq n \leq 1000$$

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○
○○
○●○

```

Hard

```

○○○○○○○
○○○
○○○
○○
○○

```

总结

```

○○○
○

```

ZOJ3823

尝试

loopy.exe 已经传到了 QQ 群里，大家可以手玩找规律。

注意原题里面是在格子里走，而这个小游戏是在分界线上，不过本质是一样的。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○
○○

```

Hard

```

○○○○○○
○○○
○○○
○○

```

总结

```

○○○
○

```

ZOJ3823

尝试

loopy.exe 已经传到了 QQ 群里，大家可以手玩找规律。

注意原题里面是在格子里走，而这个小游戏是在分界线上，不过本质是一样的。

可以先手玩一些比较小的情况，找一找规律，然后考虑比较大的情况有什么一般方法。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○○
○○

```

Not Hard

```

○○○
○○○
○○
○○

```

Hard

```

○○○○○○
○○○
○○○
○○
○○

```

总结

```

○○○
○

```

ZOJ3823

尝试

loopy.exe 已经传到了 QQ 群里，大家可以手玩找规律。

注意原题里面是在格子里走，而这个小游戏是在分界线上，不过本质是一样的。

可以先手玩一些比较小的情况，找一找规律，然后考虑比较大的情况有什么一般方法。

我把屏幕分享关了手玩一个 20×20 ，玩完了就开始讲。

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○●

Hard

○○○○○○○
○○○
○○○
○○

总结

○○○
○

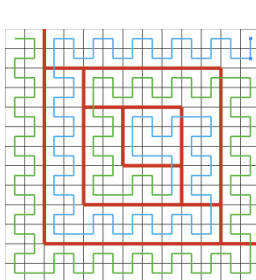
ZOJ3823

题解

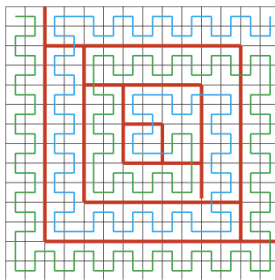
两张图，安排的明明白白：

题解

两张图，安排的明明白白：



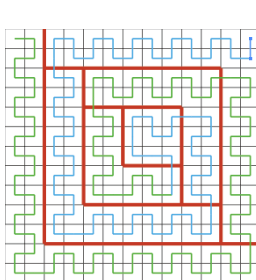
奇数



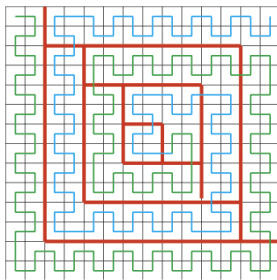
偶数

题解

两张图，安排的明明白白：



奇数



偶数

判就完了。—(说起来容易)—

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

●○○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces Gym 101221A

2014 World Finals Baggage

2014 World Finals Baggage

有一个长度为 $2n$ 的序列 BABA...BA，序列左边还有 $2n$ 个空位。

每一次，你可以选择两个相邻的元素，并且把它们移动到任意两个相邻的空位上。

你的目标是让所有的 A 都在所有的 B 左边，并且这 $2n$ 个字母仍然相邻。求最少步数并构造一组解。

$$3 \leq n \leq 100$$

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○●○○○○
○○○
○○○
○○

总结

○○○
○

CodeForces Gym 101221A

乱搞

$$n = 3$$

$$n = 4$$

乱搞

$$n = 3$$

____BABABA
 ____ABB__ABA
 ____ABBBAA____
 AAABBB_____

$$n = 4$$

____BABABABA
 ABBABAB__A
 ABBA__BBAA
 A__ABBBBAA
 AAAABBBB____

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○●○○○
○○○
○○○
○○

总结

○○○
○

CodeForces Gym 101221A

接着乱搞

$$n = 5$$

$$n = 6$$

接着乱搞

$$n = 5$$

__BABABABABA
 ABBABABAB__A
 ABBA__BABBA
 ABBAABB__BAA
 A__AABBBBBBAA
 AAAAABBBBBB__

$$n = 6$$

__BABABABABABA
 ABBABABABAB__A
 ABBABABA__BBAA
 ABB__ABAABBBBAA
 ABBAAAB__BBBAA
 A__AAABBBBBBBAA
 AAAAAABBBBBBBB__

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○●○○
○○○
○○○
○○

总结

○○○
○

CodeForces Gym 101221A

继续乱搞

$$n = 7$$

继续乱搞

$$n = 7$$

__BABABABABABA
 ABBABABAB__ABABA
 ABBABA__BBAABABA
 ABBABAABBBAAAB__A
 ABBA__ABBBAABBAA
 ABBAAAABBB__BBAA
 A__AAAABBBBBBBBAA
 AAAAAAABBBBBBBB__

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○●○
○○○
○○○
○○

总结

○○○
○

CodeForces Gym 101221A

更复杂的情况？

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○●○
○○○
○○○
○○

总结

○○○
○

CodeForces Gym 101221A

更复杂的情况？



Computer Science will destroy you!

计算机科学将会毁灭你！

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○●
○○○
○○○
○○

总结

○○○
○

CodeForces Gym 101221A

题解

对于 n 更大的情况，是存在规律的：

Very Easy

```

0000
00
00
000
00

```

Easy

```

00
00
00
000

```

Medium

```

0000
000
000
00

```

Not Hard

```

000
000
00
00
000

```

Hard

```

000000●
000
000
00

```

总结

```

000
0

```

CodeForces Gym 101221A

题解

对于 n 更大的情况，是存在规律的：

___BA[BABA...BABA]BABA

题解

对于 n 更大的情况，是存在规律的：

___BA[BABA...BABA]BABA

ABBA[BABA...BABA]B___A

ABBA[___BA...BABA]BBAA

题解

对于 n 更大的情况，是存在规律的：

___BA[BABA...BABA]BABA

ABBA[BABA...BABA]B___A

ABBA[___BA...BABA]BBAA

...

ABBA[AAAA...BB___]BBAA

题解

对于 n 更大的情况，是存在规律的：

___BA[BABA...BABA]BABA

ABBA[BABA...BABA]B___A

ABBA[___BA...BABA]BBAA

...

ABBA[AAAA...BB___]BBAA

A___A[AAAA...BBBB]BBAA

AAAA[AAAA...BBBB]BB___

题解

对于 n 更大的情况，是存在规律的：

___BA[BABA...BABA]BABA

ABBA[BABA...BABA]B___A

ABBA[___BA...BABA]BBAA

...

ABBA[AAAA...BB___]BBAA

A___A[AAAA...BBBB]BBAA

AAAA[AAAA...BBBB]BB___

时间复杂度 $O(n)$ 。

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
●○○
○○○
○○

总结

○○○
○

CodeForces Gym 100512C

Comparator Networks

Comparator Networks

有一个长度为 n 的 01 序列 a ，你需要构造下面的代码：

```

if(a[i[1]]>a[j[1]])swap(a[i[1]],a[j[1]]);
if(a[i[2]]>a[j[2]])swap(a[i[2]],a[j[2]]);
...
if(a[i[m]]>a[j[m]])swap(a[i[m]],a[j[m]]);

```

其中 $i_k < j_k$ ，并且满足 $m \leq 1000$ 。

你需要保证除了对他给你的一个序列之外，其它的序列都排好序了。无解输出 -1 。

$n \leq 10$

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
o●o
ooo
oo

总结

ooo
o

CodeForces Gym 100512C

题解

若输入已经排好了序显然无解。
我们大胆猜想剩下的情况全都有解。

题解

若输入已经排好了序显然无解。

我们大胆猜想剩下的情况全都有解。

首先将所有 0 的位置排序。由于步数限制很松，排序用 $O(n^2)$ 就可以了。

题解

若输入已经排好了序显然无解。

我们大胆猜想剩下的情况全都有解。

首先将所有 0 的位置排序。由于步数限制很松，排序用 $O(n^2)$ 就可以了。

然后摠住最后一个 0，把其它位置排序。

题解

若输入已经排好了序显然无解。

我们大胆猜想剩下的情况全都有解。

首先将所有 0 的位置排序。由于步数限制很松，排序用 $O(n^2)$ 就可以了。

然后摠住最后一个 0，把其它位置排序。

再然后摠住第一个 1，把其它位置排序。此时序列会长成 $00\dots001011\dots11$ 这个样子。

题解

若输入已经排好了序显然无解。

我们大胆猜想剩下的情况全都有解。

首先将所有 0 的位置排序。由于步数限制很松，排序用 $O(n^2)$ 就可以了。

然后摠住最后一个 0，把其它位置排序。

再然后摠住第一个 1，把其它位置排序。此时序列会长成 $00\dots001011\dots11$ 这个样子。

然后我们惊奇地发现，其它所有的序列都排好了！

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○●
○○○
○○

总结

○○○
○

CodeForces Gym 100512C

证明

Proof.

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○●
○○○
○○

总结

○○○
○

CodeForces Gym 100512C

证明

Proof.

考虑另一个序列。在第一次排序的时候，如果排好了序那就不用管了，否则分情况讨论：

证明

Proof.

考虑另一个序列。在第一次排序的时候，如果排好了序那就不用管了，否则分情况讨论：

- 如果 0 的数量小于输入，那么第一次摠住的 0 的位置一定是 1，这个 1 是排好序的，所以第二次就排好序了。
- 如果 0 的数量等于输入，同理，因为不能完全相同。
- 如果 0 的数量大于输入，那么第二次摠住的 1 的位置一定是 0，这个 0 是排好序的，所以第三次就排好序了。



证明

Proof.

考虑另一个序列。在第一次排序的时候，如果排好了序那就不用管了，否则分情况讨论：

- 如果 0 的数量小于输入，那么第一次摠住的 0 的位置一定是 1，这个 1 是排好序的，所以第二次就排好序了。
- 如果 0 的数量等于输入，同理，因为不能完全相同。
- 如果 0 的数量大于输入，那么第二次摠住的 1 的位置一定是 0，这个 0 是排好序的，所以第三次就排好序了。



总步数 $O(n^2)$ 。

A 4x4 grid of circles. The top-left circle (row 1, column 1) contains a black dot. All other circles are empty.

A 4x4 grid of circles. The top row has 4 circles, the second row has 3 circles, the third row has 3 circles, and the bottom row has 2 circles. The circle at the top-right position (row 1, column 4) is missing.

○ ○ ○
○

Restoring Map

Very Easy

```

0000
00
000
00

```

Easy

```

00
00
00
000

```

Medium

```

0000
000
000
00

```

Not Hard

```

000
000
00
00
000

```

Hard

```

0000000
000
000
●00
00

```

总结

```

000
0

```

CodeForces 566E

Restoring Map

有一棵 n 个点的树，你知道到每一个点距离不超过 2 的点的集合，但是不知道哪个集合对应了哪个点。

构造一棵合法的树。

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
o●o
oo

总结

ooo
o

CodeForces 566E

题解

观察两个点 u, v 的集合的交集的性质：

题解

观察两个点 u, v 的集合的交集的性质：

- 若 u, v 距离为 1，则交集为 u, v 以及所有与 u 或 v 直接相连的点。
- 若距离为 2，则交集为 u, v 的中点以及所有与这个点相连的点。
- 若距离为 3，则交集为 u, v 之间的两个点。
- 若距离为 4，则交集为 u, v 的中点。
- 若距离大于 4，交集为空。

题解

观察两个点 u, v 的集合的交集的性质：

- 若 u, v 距离为 1，则交集为 u, v 以及所有与 u 或 v 直接相连的点。
- 若距离为 2，则交集为 u, v 的中点以及所有与这个点相连的点。
- 若距离为 3，则交集为 u, v 之间的两个点。
- 若距离为 4，则交集为 u, v 的中点。
- 若距离大于 4，交集为空。

枚举两个集合求交集，若交集大小为 2 那么把这两个点相连，这样我们就得到了所有非叶节点之间的边。

Very Easy

```

0000
00
000
00

```

Easy

```

00
00
00
000

```

Medium

```

0000
000
000
00

```

Not Hard

```

000
000
00
00
000

```

Hard

```

0000000
000
000●
00
00

```

总结

```

000
0

```

CodeForces 566E

题解

令原树为 T_1 ，非叶节点形成的树为 T_2 。

当 T_2 的大小为 1 或 2 时特判。

Very Easy

```

0000
00
00
000
00

```

Easy

```

00
00
00
000

```

Medium

```

0000
000
000
000

```

Not Hard

```

000
000
00
00
000

```

Hard

```

0000000
000
000
00●
00

```

总结

```

000
0

```

CodeForces 566E

题解

令原树为 T_1 ，非叶节点形成的树为 T_2 。

当 T_2 的大小为 1 或 2 时特判。

否则，每次拿出 T_2 中的一个叶节点 x ，并尝试找出所有和 x 相连的点。

题解

令原树为 T_1 ，非叶节点形成的树为 T_2 。

当 T_2 的大小为 1 或 2 时特判。

否则，每次拿出 T_2 中的一个叶节点 x ，并尝试找出所有和 x 相连的点。

设在 T_2 中与 x 直接相连的点为 y ，我们找出一个在 T_2 中只包含 x, y 的集合，则这个集合的中心点一定和 x 直接相连，且所有未被访问的点都和 x 有边。做完后从 T_2 中删掉 x 。

题解

令原树为 T_1 ，非叶节点形成的树为 T_2 。

当 T_2 的大小为 1 或 2 时特判。

否则，每次拿出 T_2 中的一个叶节点 x ，并尝试找出所有和 x 相连的点。

设在 T_2 中与 x 直接相连的点为 y ，我们找出一个在 T_2 中只包含 x, y 的集合，则这个集合的中心点一定和 x 直接相连，且所有未被访问的点都和 x 有边。做完后从 T_2 中删掉 x 。

不断重复这个操作，就可以得到整棵树。注意各种实现细节。

集合之间的操作可以用 bitset 加速，复杂度为 $O\left(\frac{n^3}{w}\right)$ 。

A 4x4 grid of circles. The top-left circle (row 1, column 1) contains a black dot. All other circles are empty.

A 4x4 grid of circles. The top row has 4 circles, the second row has 3 circles, the third row has 3 circles, and the bottom row has 2 circles. The circle at the top-right position (row 1, column 4) is missing.



○ ○ ○
○

Construction of a Tree

Construction of a Tree

给定 $\{1, 2, 3, \dots, n\}$ 的 $n-1$ 个子集，令第 i 个子集为 E_i ，其中 $|E_i| \geq 2$ 。

在每一个集合中选择两个元素并且连边，使得最后形成一棵树。构造一组解。

$$2 \leq n \leq 10^5, \sum |E_i| \leq 2 \times 10^5$$

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
o●

总结

ooo
o

AtCoder agc029F

题解

首先任取一个点当根，并且把一条边视作确定一个父子关系。这样如果我们从根节点开始 bfs，相当于依次遍历每一个点集，并且在这个点集中选择了一个点。

题解

首先任取一个点当根，并且把一条边视作确定一个父子关系。这样如果我们从根节点开始 bfs，相当于依次遍历每一个点集，并且在这个点集中选择了一个点。

考虑用二分图匹配求解，左边是除了根节点之外的 $n-1$ 个点，右边是 $n-1$ 个点集，如果 E_i 这个点集包含了点 x ，就连接从 x 到 E_i 的边。

题解

首先任取一个点当根，并且把一条边视作确定一个父子关系。这样如果我们从根节点开始 bfs，相当于依次遍历每一个点集，并且在这个点集中选择了一个点。

考虑用二分图匹配求解，左边是除了根节点之外的 $n-1$ 个点，右边是 $n-1$ 个点集，如果 E_i 这个点集包含了点 x ，就连接从 x 到 E_i 的边。

接下来求一个完美匹配，如果不存在显然无解。

否则从根节点开始 bfs，如果 x 号点遍历到了，就把所有包含 x 的点集标记为遍历到并且把点集匹配的点连到 x 的下方。最后如果有点没有被访问到就无解，否则就找到了一组解。

Very Easy

```

○○○○
○○
○○○
○○

```

Easy

```

○○
○○
○○
○○○

```

Medium

```

○○○○
○○○
○○○
○○○

```

Not Hard

```

○○○
○○○
○○○
○○○

```

Hard

```

○○○○○○
○○○
○○○
○○○
●

```

总结

```

○○○
○

```

AtCoder agc029F

题解

首先任取一个点当根，并且把一条边视作确定一个父子关系。这样如果我们从根节点开始 bfs，相当于依次遍历每一个点集，并且在这个点集中选择了一个点。

考虑用二分图匹配求解，左边是除了根节点之外的 $n-1$ 个点，右边是 $n-1$ 个点集，如果 E_i 这个点集包含了点 x ，就连接从 x 到 E_i 的边。

接下来求一个完美匹配，如果不存在显然无解。

否则从根节点开始 bfs，如果 x 号点遍历到了，就把所有包含 x 的点集标记为遍历到并且把点集匹配的边连到 x 的下方。最后如果有点没有被访问到就无解，否则就找到了一组解。

用 Dinic 算法求解二分图匹配，时间复杂度 $O(\sqrt{n} \sum E_i)$ 。

Very Easy

A 4x4 grid of circles. The top-right circle (row 1, column 4) contains a black dot. All other circles are empty.

Easy

Medium

Not Hard

Hard

A 4x5 grid of circles. The first row has 5 circles, the second row has 4 circles, the third row has 3 circles, and the fourth row has 2 circles. The circles are arranged in a staircase pattern, with the number of circles decreasing by one in each subsequent row.

总结

●○○○
○

特点

构造题都有哪些特点呢?

Very Easy

oooo
oo
ooo
oo

Easy

oo
oo
oo
ooo

Medium

oooo
ooo
ooo
oo

Not Hard

ooo
ooo
oo
ooo

Hard

oooooo
ooo
ooo
oo

总结

●oo
o

特点

构造题都有哪些特点呢？

- 最大的特点当然是降智辣！构造题**几乎**没有方法可循。

特点

构造题都有哪些特点呢？

- 最大的特点当然是降智辣！构造题**几乎**没有方法可循。
- 可以和多种算法结合，比如二分图匹配，网络流，bfs/dfs，凸包等，~~好像二分图匹配~~尤其多。比如这个课件里面就涉及到了 5 道题：CF611H，agc025D，agc037D，CF1158D，agc029F。

特点

构造题都有哪些特点呢？

- 最大的特点当然是降智辣！构造题**几乎**没有方法可循。
- 可以和多种算法结合，比如二分图匹配，网络流，bfs/dfs，凸包等，好像二分图匹配尤其多。比如这个课件里面就涉及到了 5 道题：CF611H，agc025D，agc037D，CF1158D，agc029F。
- OI 中涉及比较少（可能是因为不好区分难度），ACM 中偏多一些。

特点

构造题都有哪些特点呢？

- 最大的特点当然是降智辣！构造题**几乎**没有方法可循。
- 可以和多种算法结合，比如二分图匹配，网络流，bfs/dfs，凸包等，好像二分图匹配尤其多。比如这个课件里面就涉及到了 5 道题：CF611H，agc025D，agc037D，CF1158D，agc029F。
- OI 中涉及比较少（可能是因为不好区分难度），ACM 中偏多一些。
- 由于考到的主要是选手的思维能力，所以如果出到比赛里面，可能会狙掉一些神仙，也可能让一些人占便宜。

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

方法

几乎没有方法可寻不代表完全没有方法可寻！

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
○

方法

几乎没有方法可寻不代表完全没有方法可寻！
构造题都有哪些方法呢？

方法

几乎没有方法可寻不代表完全没有方法可寻！

构造题都有哪些方法呢？

- 1) 增量法。每次考虑增加一个元素或者删除一个元素，将问题分解。如 CF1305E, agc025E, ZOJ3823, Gym101221A。

方法

几乎没有方法可寻不代表完全没有方法可寻！

构造题都有哪些方法呢？

- 1) 增量法。每次考虑增加一个元素或者删除一个元素，将问题分解。如 CF1305E, agc025E, ZOJ3823, Gym101221A。
- 2) 如果题目要求构造一个方案使得权值等于某个数，可以考虑构造理论最大值，理论最小值并且尝试在这种构造方法的基础上构造一般情况。如 CF1311E, 1305E, agc025E。

方法

几乎没有方法可寻不代表完全没有方法可寻！

构造题都有哪些方法呢？

- 1) 增量法。每次考虑增加一个元素或者删除一个元素，将问题分解。如 CF1305E, agc025E, ZOJ3823, Gym101221A。
- 2) 如果题目要求构造一个方案使得权值等于某个数，可以考虑构造理论最大值，理论最小值并且尝试在这种构造方法的基础上构造一般情况。如 CF1311E, 1305E, agc025E。
- 3) 对于构造树的情况，考虑链/菊花图/二叉树等特殊情况，正解很有可能是这几种情况的组合。如 CF1311E。

方法

几乎没有方法可寻不代表完全没有方法可寻！

构造题都有哪些方法呢？

- 1) 增量法。每次考虑增加一个元素或者删除一个元素，将问题分解。如 CF1305E, agc025E, ZOJ3823, Gym101221A。
- 2) 如果题目要求构造一个方案使得权值等于某个数，可以考虑构造理论最大值，理论最小值并且尝试在这种构造方法的基础上构造一般情况。如 CF1311E, 1305E, agc025E。
- 3) 对于构造树的情况，考虑链/菊花图/二叉树等特殊情况，正解很有可能是这几种情况的组合。如 CF1311E。
- 4) 对于完全找不到思路的题，可以先考虑手玩构造特殊情况，通过特殊情况寻找普遍规律。如 agc030C。

方法

- 5) 考虑质数、2 的整数次幂等特殊的数字，如 arc103D，
agc027D，agc035C。

方法

- 5) 考虑质数、2 的整数次幂等特殊的数字，如 arc103D，agc027D，agc035C。
- 6) 如果有集合，可以考虑它们的并、交等等，尝试寻找性质。如 CF566E。

方法

- 5) 考虑质数、2 的整数次幂等特殊的数字，如 arc103D，agc027D，agc035C。
- 6) 如果有集合，可以考虑它们的并、交等等，尝试寻找性质。如 CF566E。
- 7) 和贪心等思路相结合，从而构造出解。如 agc022E。

方法

- 5) 考虑质数、2 的整数次幂等特殊的数字，如 arc103D，agc027D，agc035C。
- 6) 如果有集合，可以考虑它们的并、交等等，尝试寻找性质。如 CF566E。
- 7) 和贪心等思路相结合，从而构造出解。如 agc022E。
- 8) 拆出来限制，构造出解。如 CF538G。

方法

- 5) 考虑质数、2 的整数次幂等特殊的数字，如 arc103D，agc027D，agc035C。
- 6) 如果有集合，可以考虑它们的并、交等等，尝试寻找性质。如 CF566E。
- 7) 和贪心等思路相结合，从而构造出解。如 agc022E。
- 8) 拆出来限制，构造出解。如 CF538G。
- 9) 最重要的还是看脑洞。(雾)

方法

- 5) 考虑质数、2 的整数次幂等特殊的数字，如 arc103D，agc027D，agc035C。
- 6) 如果有集合，可以考虑它们的并、交等等，尝试寻找性质。如 CF566E。
- 7) 和贪心等思路相结合，从而构造出解。如 agc022E。
- 8) 拆出来限制，构造出解。如 CF538G。
- 9) 最重要的还是看脑洞。(雾

多良心的讲题人，此处应有掌声！

Very Easy

○○○○
○○
○○○
○○

Easy

○○
○○
○○
○○○

Medium

○○○○
○○○
○○○
○○

Not Hard

○○○
○○○
○○
○○○

Hard

○○○○○○
○○○
○○○
○○

总结

○○○
●

Q&A

谢谢大家!