

省选联考 2024 模拟赛 Day1

题目名称	来自冥王星的外星旅人	雷城	花に亡霊
题目类型	传统型	传统型	传统型
评测方式	全文比较	全文比较	全文比较
目录	alien	thunder	undead
可执行文件名	alien	thunder	undead
输入文件名	alien.in	thunder.in	undead.in
输出文件名	alien.out	thunder.out	undead.out
每个测试点时限	1.0s	2.0s	1.0s
内存限制	512 MiB	512 MiB	512 MiB
测试点（子任务）数目	10	7	4
是否有子任务捆绑	否	是	是
测试点是否等分	是	否	否

注意事项（请仔细阅读）：

- 文件名（程序名和输入输出文件名）必须使用英文小写。
- C/C++ 中函数 `main()` 的返回值类型必须是 `int`，程序正常结束时的返回值必须是 0。
- 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
- 只提供 Linux 格式附加样例文件。
- 使用 410 的机器评测 (Ubuntu 20.04 Desktop LTS, Powered by [CMS](#))。
- 如果 `Evaluating` 超过 2min，请善用 CMS 的 Communication 功能。
- 如果您不幸见过原题，请告知我。
- 祝大家好运。

来自冥王星的外星旅人 (alien)

• 题目背景

来自冥王星的外星旅人喜欢棋类，因为他的家乡没有棋。

• 题目描述

今天 Alice 不和 Bob 玩游戏了，她要来自冥王星的外星旅人玩一种新的棋类游戏。

棋盘的大小是 $n \times n$ ，每次操作的一方选择一个没有棋子的 $k \times k$ 的正方形，然后在 **棋盘上任意** 一个位置落下一枚棋子。当有一方无法操作时，TA 就输了。

由于来自冥王星的外星旅人第一次玩这个游戏，因此 Alice 让冥王星人先手。但是，Alice 和来自冥王星的外星旅人都绝顶聪明。那您，我的朋友，拥有比他们还高的智慧，一定已经知道游戏的结果了吧！

• 输入格式

从文件 `alien.in` 输入数据。

本题有多组测试数据。

第一行一个正整数 T ，表示数据组数。

对于每组测试数据，第一行两个正整数 n, k 。接下来 n 行，每行一个长度为 n 的 01 字符串，代表这个位置是否有棋子。

• 输出格式

输出到文件 `alien.out` 中。

共 T 行，每行一个字符串，如果 Alice 赢了输出 `Alice`，否则输出 `Pluto`。

• 输入输出样例

- 样例 1 输入

```
5
1 1
0
1 1
1
```

```
2 2
00
00
2 2
01
00
3 2
100
000
001
```

- 样例 1 输出

```
Pluto
Alice
Pluto
Alice
Pluto
```

- 样例 1 解释

在第 5 组测试样例中，来自冥王星的外星旅人只需要在 (2, 2) 下一个棋子即可赢得游戏。

- 样例 2 输入 / 输出

见选手目录下的 `alien/alien2.in` 与 `alien/alien2.ans`。

除了 $T = 30$ 外，该样例满足测试点 2 的性质。

- 样例 3 输入 / 输出

见选手目录下的 `alien/alien3.in` 与 `alien/alien3.ans`。

该样例满足测试点 8 的性质。

• 数据范围

对于 100% 的数据，保证 $10 \leq T \leq 30, 1 \leq k \leq n \leq 10^3$ 。

测试点编号	$T =$	$n \leq$	特殊性质
1	10	3	
2	10	5	
3	10	10^3	$k = n$
4	10	100	初始时最多存在一个满足条件的 $k \times k$ 矩形
5	10	100	初始没有棋子
6	20	200	$2 \times k > n$
7	30	200	0 的个数最多只有 4 个
8	30	200	0 的个数最多只有 9 个
9	30	200	0 的个数最多只有 15 个
10	30	10^3	

雷城 (thunder)

• 题目背景

巫峡中宵动，沧江十月雷。
 龙蛇不成蛰，天地划争迴。
 却碾空山过，深蟠绝壁来。
 何须妒云雨，霹雳楚王台。
 ——杜甫《雷》

雷城是一座坐落在希斯塔附近的神秘城市。它有时可见，有时不可见。它吸引着世界各地的冒险家前来探索，但是很少听说过谁真正走进了雷城。

作为一名旅行者，今天，你来到了雷城，才知道了人们不想进入雷城的原因。原来，雷城的每个城市都存有一定的雷电能量。但是如果能量的分配不符合自然的规律，那么雷城就会毁灭。

如果想要安全离开雷城，那么你需要先调整好雷城里的雷电能量才行。

• 题目描述

雷城是一棵 n 个点的树的结构，每条边 (u_i, v_i) 都有边权 c_i 。

现在第 i 个点有 x_i 的雷电能量，每个点需要至少 y_i 的雷电能量，你可以移动这些雷电能量，移动的代价是移动的雷电能量大小 \times 雷电能量经过的边权和。

求满足所有点要求的最小代价。

• 输入格式

从文件 `thunder.in` 输入数据。

第一行一个正整数 n 。

接下来 $n - 1$ 行，每行三个正整数 u_i, v_i, c_i ，表示树上的边。

接下来 n 行，每行两个正整数 x_i, y_i 。

• 输出格式

输出到文件 `thunder.out` 中。

一行一个正整数，表示答案。

• 输入输出样例

- 样例 1 输入

```
3
1 2 5
3 1 5
2 1
5 0
1 3
```

- 样例 1 输出

```
15
```

- 样例 1 解释

1, 2 号点分别向 3 号点分别运送单位为 1 的雷电能量即可，这样运送雷电能量消耗的代价是 $1 \times 5 + 1 \times 10 = 15$ 。

- 样例 2 输入

```
6
1 2 2
1 3 5
1 4 1
2 5 5
2 6 1
0 0
1 0
2 1
2 1
0 1
0 1
```

- 样例 2 输出

```
9
```

- 样例 3 输入 / 输出

见选手目录下的 `thunder/thunder3.in` 与 `thunder/thunder3.ans`。

该样例满足子任务 3 的性质。

- 样例 4 输入 / 输出

见选手目录下的 `thunder/thunder4.in` 与 `thunder/thunder4.ans`。

该样例满足子任务 6 的性质。

- 样例 5 输入 / 输出

见选手目录下的 `thunder/thunder5.in` 与 `thunder/thunder5.ans`。

该样例满足子任务 7 的性质。

• 数据范围

本题采用捆绑测试。

对于 100% 的数据，保证

$1 \leq n \leq 2.5 \times 10^5, 1 \leq c \leq 10^6, 0 \leq \sum y \leq \sum x \leq 10^6$ 。

子任务编号	$n \leq$	特殊性质	分值
1	2		5
2	300	$c = 1, \sum x \leq 300$	10
3	10^3	$c = 1, \sum x \leq 5 \times 10^3$	10
4	10^5	$c = 1, u_i = i, v_i = i + 1$	10
5	10^5	$u_i = i, v_i = i + 1$	10
6	300	$\sum x \leq 300$	20
7	2.5×10^5		35

花に亡霊 (undead)

• 题目背景

- 忘れないように
不要让记忆消逝啊
- 色褪せないように
不要让色彩褪去啊
- 心に響くものが全てじゃないから
因为响彻于心灵的并非全部啊
- 言葉をもっと教えて
告诉我更多的话语吧
- さよならだって教えて
告诉我什么是告别吧
- 今も見ろんだよ
我现在仍在看着哦
- 夏に咲いてる花に亡霊を
那夏日盛开的繁花之上的亡灵啊

• 题目描述

月有 q 次询问，每次询问给定两个正整数 n, m ，求：

$$\left(\sum_{i=0}^n \sum_{j=0}^m \binom{i}{j} [(i+j) \bmod 2 = 0] \right) \bmod 998244353$$

- 输入格式

从文件 `undead.in` 输入数据。

第一行一个正整数 q 。

接下来 q 行，每行两个非负整数 n_i, m_i 。

- 输出格式

输出到文件 `undead.out` 中。

共 q 行，每行一个非负整数表示答案。

- 输入输出样例

- 样例 1 输入

```
10
0 0
2 1
9 2
10 3
26 8
60 8
202 161
2013 2016
10635 1080
75689 22265
```

- 样例 1 输出

```
1
3
80
256
4417154
6105294
565123535
53217606
135348654
87065825
```


- 样例 2 输入 / 输出

见选手目录下的 `undead/undead2.in` 与 `undead/undead2.ans`。

该样例满足子任务 1 的性质。

• 数据范围

本题采用捆绑测试。

对于 100% 的数据，保证 $1 \leq q, n, m \leq 10^5$ 。

子任务编号	限制	分值
1	$n, m \leq 5000$	18
2	所有的 n 都相等	11
3	所有的 m 都相等	31
4		40