

OI 训练赛 Day5

[七夕欢乐赛]

题目名称	英文名	时间限制	内存限制	答案比较方式
链式反应	pcr	1s	128MB	全文比较，忽略行尾空白和文末回车
地毯	carpet	1s	128MB	全文比较，忽略行尾空白和文末回车
红石灯	lamp	1s	128MB	全文比较，忽略行尾空白和文末回车
路障	bar	1s	128MB	全文比较，忽略行尾空白和文末回车

注意事项就不用我多说了⑧！

今天过节，我们特意降低了题目难度，玩得愉快 w

七夕快乐，祝您早日脱单！

Best wishes!

阮行止, rxz@luogu.org

A. 链式反应(pcr)

题目描述

PCR（多聚酶链式反应）技术是一种机器自动控制的 DNA 复制技术。

该技术中，根据生成 DNA 两条链的长度，可以将生成的 DNA 分成长中、中短、短短三种类型。

第一次循环的时候，模板链会产生两条长中型 DNA。从第二次循环开始，每条长中型 DNA 可以变成长中和中短型 DNA 各一条，每条中短型 DNA 可以变成中短和短短型 DNA 各一条，而每条短短型 DNA 可以变成两条短短型 DNA。

在 PCR 生成的所有 DNA 产物中，只有短短型 DNA 是我们需要的目的基因，现在问，PCR 技术进行 n 次循环后会产生多少个目的基因？

因为答案可能很大，答案模 19260817 输出。

输入格式

输入仅一行，一个正整数，表示 n 。

输出格式

输出仅一行，一个正整数，表示答案。

样例数据

pcr.in	pcr.out
4	8

第一轮循环结束后，有 2 条长中型。

第二轮循环结束后，有 2 条长中型、2 条中短型。

第三轮循环结束后，有 2 条长中型、4 条中短型、2 条短短型。

第四轮循环结束后，有 2 条长中型、6 条中短型、8 条短短型。

数据规模与约定

对于前 30%的数据，保证 $n \leq 5$ 。

对于前 50%的数据，保证 $n \leq 10^6$ 。

对于 100%的数据，保证 $n \leq 10^{18}$ 。

B. 地毯(carpet)

题目描述

在 $n \times n$ 的格子中有 m 个地毯。

给出这些地毯的覆盖范围，问每个点被多少个地毯覆盖。

输入格式

第一行，两个正整数 n, m 。意义如题所述。

接下来 m 行，每行两个坐标 (x_1, y_1) 和 (x_2, y_2) ，代表一块地毯，左上角是 (x_1, y_1) ，右下角是 (x_2, y_2) 。

输出格式

输出 n 行，每行 n 个正整数。

第 i 行第 j 列的正整数表示 (i, j) 这个格子被多少个地毯覆盖。

样例数据

carpet.in	carpet.out
5 3	0 1 1 1 0
2 2 3 3	0 1 1 0 0
3 3 5 5	0 1 2 1 1
1 2 1 4	0 0 1 1 1
	0 0 1 1 1

样例解释：

0 0 0 0 0	0 0 0 0 0	0 1 1 1 0
0 1 1 0 0	0 1 1 0 0	0 1 1 0 0
0 1 1 0 0	-> 0 1 2 1 1	-> 0 1 2 1 1
0 0 0 0 0	0 0 1 1 1	0 0 1 1 1
0 0 0 0 0	0 0 1 1 1	0 0 1 1 1

数据规模与约定

对于前 20% 的数据，保证 $n \leq 50, m \leq 100$ 。

对于前 100% 的数据，保证 $n, m \leq 1000$ 。

C. 红石灯(lamp)

题目描述

七夕有祈福的习俗。而作为一名 MC 玩家，我们选择使用红石灯来祈福。

如果你没有玩过 MC，也没有关系，我们下面说一些前置知识。红石灯只有两个状态：开和关。 k 个红石灯放成一排，称为一个红石灯序列。我们可以通过扳动红石灯后的拉杆，将这个红石灯的状态翻转。

七夕当晚 8 点，服务器管理员 rxz 会公布一个“幸运序列”——也就是一个长度为 k 的 01 串，0 表示关，1 表示开。如果您的红石灯序列与幸运序列一致，那么您就能获得 rxz 的祝福！

您已经建造了 n 个红石灯序列，初始的开关状态将会在输入数据中给出。在幸运序列公布之前，您并不知道幸运序列是哪个串；所以在幸运序列公布之后，您需要选择自己的一个红石灯序列，然后扳动若干个红石灯的拉杆，从而让这个序列与幸运序列一致。

现在，您想知道，最坏情况下，您至少需要扳动多少次拉杆。

输入格式

第一行，两个正整数，分别表示 n, k 。

接下来 n 行，每行一个二进制串，长度为 k ，表示已有的红石灯序列的状态。

输出格式

输出仅一行，一个整数，表示最坏情况下需要扳动多少次拉杆。

样例数据

lamp.in	lamp.out
2 3 0000 1111	2

最坏情况下，幸运序列为 0110 或 1001。此时您需要扳动两次拉杆。

其余情况下，只需要扳动至多一次拉杆。比如 1101，您可以选择 1111 为初始序列，然后关掉第三个红石灯。

数据规模与约定

对于前 30% 的数据，保证 $n \leq 5, k \leq 3$ 。

对于前 70% 的数据，保证 $n \leq 1000, k \leq 5$ 。

对于 100% 的数据，保证 $n \leq 100000, k \leq 17$ 。

D. 路障(bar)

题目描述

B 君站在一个 $n \times n$ 的棋盘上。

最开始, B 君站在 $(1,1)$ 这个点, 他要走到 (n,n) 这个点。B 君每秒期间可以向上下左右的某个方向移动一格, 但是很不妙, C 君打算阻止 B 君的计划。

每秒结束的那一瞬间, C 君会在 (x,y) 上摆一个路障。在任何时刻, B 君所在的点都不能有路障。

我们知道 C 君准备在哪些点放置路障。现在你需要判断, B 君能否走到 (n,n) 。

输入格式

首先是一个正整数 T , 表示数据组数。

对于每一组数据: 第一行是一个正整数, 表示 n . 接下来 $2n - 2$ 行, 每行两个正整数 x 和 y , 表示在那一秒结束时, (x,y) 将被摆上路障。

输出格式

对于每一组数据, 输出 Yes 或 No, 回答 B 君能否走到 (n,n) 。

样例数据

bar.in	bar.out
2	Yes
2	Yes
1 1	
2 2	
5	
3 3	
3 2	
3 1	
1 2	
1 3	
1 4	
1 5	
2 2	

样例解释:

以下 0 表示能走, x 表示不能走, B 表示 B 君现在的位置。从左往右表示时间。

Case 1:

0 0 0 0 0 B (已经走到了)

B 0 x B x 0

Case 2:

```

0 0 0 0 0    0 0 0 0 0    0 0 0 0 0    0 0 0 0 0
0 0 0 0 0    0 0 0 0 0    0 0 0 0 0    0 0 0 0 0
0 0 0 0 0    0 0 x 0 0    0 0 x 0 0    0 0 x 0 0
0 0 0 0 0    0 0 0 0 0    0 0 x 0 0    0 0 x 0 0
B 0 0 0 0 0    0 B 0 0 0    0 0 B 0 0    0 0 x B 0 ....(B 君可以走到终点)
    
```

数据规模与约定

对于前 20%的数据，保证 $n \leq 3$.

对于前 60%的数据，保证 $n \leq 500$.

对于 100%的数据，保证 $n \leq 1000, T \leq 10$.