

第六届「帆软杯」

东南大学大学生程序设计竞赛（冬季）

专业组

题目编号	题目名称	时间限制	内存限制	评测方式
A	宠物店	1.0 s	256 MiB	文本比较
B	广义线段树	3.0 s	384 MiB	文本比较
C	礼物	0.5 s	256 MiB	Special Judge
D	概率论	1.0 s	256 MiB	文本比较
E	可靠的猪宝宝	2.0 s	256 MiB	文本比较
F	XCPC 重开器	0.5 s	256 MiB	Special Judge
G	孤独之歌	0.5 s	256 MiB	文本比较
H	地球上最后的夜晚	1.0 s	256 MiB	文本比较
I	心意不尽，前程已近	1.0 s	256 MiB	文本比较
J	新鲜的日子和可爱的日子	0.5 s	256 MiB	Special Judge
K	无聊的猪宝宝	1.0 s	256 MiB	文本比较
L	量子计算程序设计	1.0 s	256 MiB	Special Judge

时空限制仅对 C++ 语言作保证。

A. 宠物店

时间限制： 1.0 s

内存限制： 256 MiB

题目描述

宠物店店长发现店里好吃懒做的宠物越来越多了。特别是猪，只会睡大觉，出题费被扣一半；还有猫，逐渐只会唱哈基米。为了整治宠物店的不良风气，让宠物们有真才实学，店长决定开设算术课，让宠物们学习算术。

某天，店长让 n^2 只宠物坐成 n 行 n 列，行和列的编号均为 $1, 2, \dots, n$ 。店长决定玩一个小游戏，来测试宠物们的算术能力。每只宠物被要求想着自己的数，初始时都是 0。店长会依次发出 m 条命令，命令分四种：

- 1 k ：要求所有宠物都把各自的数加 k 。
- 2 x k ：要求第 x 行的宠物都把各自的数加 k 。
- 3 y k ：要求第 y 列的宠物都把各自的数加 k 。
- 4 x y ：要求第 x 行第 y 列的宠物说出自己的数。

对于每条第 4 种命令，请你告诉店长正确答案应该是多少。

在你成功解决本题后，店长决定再次提醒你，题目不是按难度排列的。

输入格式

第一行两个整数 n, m ($1 \leq n \leq 2025$, $1 \leq m \leq 10^6$)。

接下来 m 行，每行 2 或 3 个整数，表示一条命令，格式见题目描述。

在命令中，保证 $1 \leq x, y \leq n$, $-10^9 \leq k \leq 10^9$ 。

输出格式

对于每条第 4 种命令，输出一行一个整数，表示第 x 行第 y 列的宠物此时的数应该是多少。

样例 1

输入

```
3 5
1 9
2 3 -1
4 1 3
3 3 4
4 3 3
```

输出

```
9
12
```

解释

命令	解释
1 9	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 9 & 9 & 9 \\ 9 & 9 & 9 \\ 9 & 9 & 9 \end{bmatrix}$
2 3 -1	$\begin{bmatrix} 9 & 9 & 9 \\ 9 & 9 & 9 \\ 9 & 9 & 9 \end{bmatrix} \rightarrow \begin{bmatrix} 9 & 9 & 9 \\ 9 & 9 & 9 \\ 8 & 8 & 8 \end{bmatrix}$
4 1 3	$\begin{bmatrix} 9 & 9 & 9 \\ 9 & 9 & 9 \\ 8 & 8 & 8 \end{bmatrix}, \text{ 答案是 } 9$
3 3 4	$\begin{bmatrix} 9 & 9 & 9 \\ 9 & 9 & 9 \\ 8 & 8 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 9 & 9 & 13 \\ 9 & 9 & 13 \\ 8 & 8 & 12 \end{bmatrix}$
4 3 3	$\begin{bmatrix} 9 & 9 & 13 \\ 9 & 9 & 13 \\ 8 & 8 & 12 \end{bmatrix}, \text{ 答案是 } 12$

样例 2

输入

2025 10
4 1394 821
1 998244353
3 985 123456789
2 1024 987654321
3 996 996412345
4 1024 996
1 -1000000000
1 -1000000000
2 2025 -1919810
4 2025 985

输出

0
2982311019
-880218668

B. 广义线段树

时间限制： 3.0 s

内存限制： 384 MiB

题目描述

给定一棵有 $2n - 1$ 个节点的二叉树，节点的编号为 $1, 2, \dots, 2n - 1$ 。

编号为 p 的节点（下称节点 p ）有两个给定的整数属性值 L_p, R_p ，满足 $L_p \leq R_p$ 。

该二叉树还满足以下性质：

- 节点 1 是根， $L_1 = 1$ ， $R_1 = n$ 。
- 对于任意一个节点 p ：
 - 若 $L_p = R_p$ ，则节点 p 没有子节点。
 - 若 $L_p < R_p$ ，则节点 p 有两个子节点。在这里，区分左右子节点。设节点 p 的左子节点是 u ，右子节点是 v ，有 $L_u = L_p$ 且 $R_u + 1 = L_v$ 且 $R_v = R_p$ 。

因此，称它是一棵广义线段树。

给定区间 $[l, r]$ 满足 l, r 均为整数且 $1 \leq l \leq r \leq n$ ，考虑以下算法：

```

function visit( $p, l, r, id$ )
  1  if  $r < L_p$  or  $R_p < l$ 
  2    return
  3   $cnt_1[id] \leftarrow cnt_1[id] + 1$ 
  4   $cnt_2[p] \leftarrow cnt_2[p] + 1$ 
  5  if  $l \leq L_p$  and  $R_p \leq r$ 
  6    return
  7   $u \leftarrow$  left child of  $p$ 
  8   $v \leftarrow$  right child of  $p$ 
  9  visit( $u, l, r, id$ )
 10 visit( $v, l, r, id$ )

```

给定 m 个区间 $[l_i, r_i]$ ($i = 1, 2, \dots, m$)，对于每个 i 都执行 $visit(1, l_i, r_i, i)$ 恰好一次。数组 cnt_1 （下标为 $1, 2, \dots, m$ ）和 cnt_2 （下标为 $1, 2, \dots, 2n - 1$ ）的所有元素初始时都为 0，求它们的最终状态。

输入格式

第一行两个整数 n, m ($1 \leq n, m \leq 10^6$)。

接下来 $2n - 1$ 行中的第 i 行表示节点 i 的情况：

- 每行先是两个整数 L_i, R_i ($1 \leq L_i \leq R_i \leq n$)。
- 若 $L_i < R_i$ ，又有两个整数 u, v ($1 \leq u, v \leq 2n - 1$)，分别表示节点 i 的左子节点和右子节点。

接下来 m 行中的第 i 行输入两个整数 l_i, r_i ($1 \leq l_i \leq r_i \leq n$)。

输出格式

第一行 m 个整数，其中第 i 个数表示 $cnt_1[i]$ 最终的值。

第二行 $2n - 1$ 个整数，其中第 i 个数表示 $cnt_2[i]$ 最终的值。

样例

输入

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

输出

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

C. 礼物

时间限制： 0.5 s

内存限制： 256 MiB

题目描述

S 君设计了一道题目，准备作为礼物送给大家。

给定一个长度为 n 的整数序列 $a_1, a_2, \dots, a_n$ 。

输出它的任意一个满足以下要求的非空子序列 $a_{b_1}, a_{b_2}, \dots, a_{b_k}$ ：

$$\left(\sum_{i=1}^k a_{b_i} \right) \bmod n = 0$$

可以证明，这样的子序列一定存在。

小 L 看完题目后，向大家解释了一些基本概念。

任取整数 k 满足 $1 \leq k \leq n$ ，以及 k 个整数 $b_1, b_2, \dots, b_k$ 满足

$$1 \leq b_1 < b_2 < \dots < b_k \leq n$$

则称 $a_{b_1}, a_{b_2}, \dots, a_{b_k}$ 是 $a_1, a_2, \dots, a_n$ 的一个非空子序列。

关于求和号 $\sum$ ，它的含义是

$$\sum_{i=1}^k a_{b_i} = a_{b_1} + a_{b_2} + \dots + a_{b_k}$$

在这里， $\bmod$ 运算的定义是

$$x \bmod y = x - \left\lfloor \frac{x}{y} \right\rfloor y$$

其中 $\lfloor z \rfloor$ 表示 $\leq z$ 的最大整数。 $x \bmod y = 0$ 等价于 x 是 y 的倍数。

输入格式

第一行一个整数 T ($T \geq 1$)，表示该测试点有 T 组数据。

接下来，对于每组数据：

- 第一行一个整数 n ($n \geq 1$)。
- 第二行 n 个整数 $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^9$)。

保证 T 组数据中的 n 之和 $\sum n \leq 10^6$ 。

输出格式

对于每组数据，输出两行：

- 第一行一个整数 k 。
- 第二行 k 个整数 $b_1, b_2, \dots, b_k$ 。

你需要保证：

- $1 \leq k \leq n$ 。
- $1 \leq b_1 < b_2 < \dots < b_k \leq n$ 。
- $(a_{b_1} + a_{b_2} + \dots + a_{b_k}) \bmod n = 0$ 。

若有多组解，只需输出任意一个。

样例

输入

```
2
6
1 1 4 5 1 4
6
1 1 4 5 1 4
```

一个满足要求的输出

```
2
2 4
5
1 2 3 4 5
```

解释

第一组数据中， $a_2 + a_4 = 1 + 5 = 6$ ， $6 \bmod 6 = 0$ ，满足要求。

第二组数据中， $a_1 + a_2 + a_3 + a_4 + a_5 = 1 + 1 + 4 + 5 + 1 = 12$ ， $12 \bmod 6 = 0$ ，满足要求。

可见，答案应为 $\frac{1}{16} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8} + \frac{1}{4} = \frac{11}{16}$ 。

现在，有一个 n 级台阶，某人要从第 0 级台阶上到第 n 级台阶。每次都以 $\frac{1}{2}$ 的概率上 1 级台阶，以 $\frac{1}{2}$ 的概率上 2 级台阶。特别地，在第 $n-1$ 级台阶时，只能上 1 级台阶。求经过第 m 级台阶的概率。

答案对质数 $10^9 + 7$ 取模。

可以证明，答案能被表示为 $\frac{p}{q}$ ，其中 p, q 均为正整数，且 q 不是 $10^9 + 7$ 的倍数。

你需要输出 $x = p \cdot q^{-1} \bmod (10^9 + 7)$ ，其中 q^{-1} 满足 $q \cdot q^{-1} \equiv 1 \pmod{10^9 + 7}$ 。

因为模数是质数，易证 q^{-1} 存在则唯一。由费马小定理， $q^{-1} \equiv q^{10^9+5} \pmod{10^9 + 7}$ 。

输入格式

第一行一个整数 T ($1 \leq T \leq 5$)，表示该测试点有 T 组数据。

接下来，对于每组数据，输入一行两个整数 n, m ($1 \leq m \leq n \leq 10^{1000000}$)。

输出格式

对于每组数据，输出一行一个整数，表示答案对 $10^9 + 7$ 取模后的结果。

样例

输入

```
4
7 4
7 1
7 7
9876543210123456789 54321012345
```

输出

```
187500002
500000004
1
376896505
```

解释

前三组数据的答案依次是 $\frac{11}{16}, \frac{1}{2}, 1$ 。

E. 可靠的猪宝宝

时间限制： 2.0 s

内存限制： 256 MiB

题目描述

为了确认猪宝宝是一名可靠的中级算法竞赛选手，小 L 给它出了这样一道题。

任意正整数 x 都能被唯一表示为 $\overline{d_m d_{m-1} \cdots d_0}$ ，满足：

- $x = \sum_{i=0}^m 10^i d_i$ 。
- $1 \leq d_m \leq 9$ 。
- 对于 $i = 0, 1, \cdots, m-1$ ，都有 $0 \leq d_i \leq 9$ 。

称这个正整数 x 是好的，当且仅当它满足以下要求：

- 2, 3, 5, 7 都不是 x 的约数
- 对于 $i = 0, 1, \cdots, m-3$ ，都有 $d_{i+3} \neq 2$ 或 $d_{i+2} \neq 0$ 或 $d_{i+1} \neq 2$ 或 $d_i \neq 5$ 。

设函数 $f(x)$ 的定义域为正整数集。若 x 是好的，则 $f(x) = 2025^x$ ；否则， $f(x) = 0$ 。

给定正整数 L, R ($L \leq R$)，求 $\left(\sum_{i=L}^R f(i) \right) \bmod (10^9 + 7)$ 。

输入格式

一行两个整数 L, R ($1 \leq L \leq R \leq 10^{202512}$)。

输出格式

一行一个整数，表示答案。

样例 1

输入

```
20249 20257
```

输出

```
935315683
```

解释

只有 20249 是好的。 $2025^{20249} \bmod (10^9 + 7) = 935315683$ 。

样例 2

输入

```
1 20252025202520252025
```

输出

4427934

F. XCPC 重开器

时间限制： 0.5 s

内存限制： 256 MiB

题目描述

参加完 2077 年的 XCPC 后，你意识到自己无缘更进一步了。

你躺在床上，闭上眼，回想起你们曾经在赛场上挥洒汗水的场景。

眼泪还是流了出来。你不甘心，你觉得你们还可以做得更好。

你突然惊醒，发现自己回到了竞赛的前一天晚上。

之前经历的一切仿佛是一场梦，却又那么真实。

你意识到，这一次，你们还有机会。

你决定，这一次，让自己不留遗憾。

你充满了决心。

一场 XCPC 的时长是 300 分钟。

有 n 道题，数字编号依次是 $1, 2, \dots, n$ ，字母编号依次是 $A, B, \dots$ 。

已知数字编号为 i 的题做 t_i 分钟才能通过，且通过前会有 d_i 次错误的提交。

若数字编号为 i 的题拼尽全力也无法战胜，则 $t_i = 301$ 。

特别需要注意的是，在同一分钟内你们只能做同一道题。

最后，通过的题数多的队伍排名靠前。若通过的题数相等，则罚时少的队伍排名靠前。

设某支队伍在 $s_1, s_2, \dots, s_m$ 分钟时分别通过了数字编号为 $p_1, p_2, \dots, p_m$ 的题，则该队伍的罚时是

$$\sum_{i=1}^m (s_i + 20d_{p_i})$$

求最优策略——在使通过的题数最多的前提下，使罚时最少。

输入格式

第一行一个整数 T ($1 \leq T \leq 15000$)，表示该测试点有 T 组数据。

接下来，对于每组数据：

- 第一行一个整数 n ($1 \leq n \leq 26$)。
- 接下来 n 行中的第 i 行，有两个整数 t_i, d_i ($1 \leq t_i \leq 301$, $0 \leq d_i < t_i$)。

保证 T 组数据中的 n 之和 $\sum n \leq 202500$ 。

输出格式

G. 孤独之歌

时间限制： 0.5 s
内存限制： 256 MiB

题目描述

拉普拉斯科算中心正在破解术士密码。

有三种操作。通过这三种操作，你需要打出给定的 01 串（由 0 和 1 构成的字符串）。

三种操作是 0、1 和 B：

- 0 表示在末尾添加一个 0。
- 1 表示在末尾添加一个 1。
- B 表示删除最后一个字符。特别地，若当前字符串为空，则什么也不做。

初始时，打字机上的字符串为空。

你需要进行恰好 n 次操作，打出 01 串 S 。

求有多少个操作方案。答案对 $10^9 + 7$ 取模。

输入格式

第一行一个整数 n ($1 \leq n \leq 5000$)。

第二行一个 01 串 S ($1 \leq |S| \leq n$)。

输出格式

一行一个整数，表示答案对 $10^9 + 7$ 取模后的结果。

样例 1

输入

3
1

输出

5

解释

有 5 个操作方案：10B、11B、0B1、1B1、BB1。

样例 2

输入

9
110

输出

509

H. 地球上最后的夜晚

时间限制： 1.0 s

内存限制： 256 MiB

题目描述

给定一张由 n 个节点和 m 条边构成的无向图。节点的编号是 $1, 2, \dots, n$ ，边的编号是 $1, 2, \dots, m$ 。

你需要选取若干条边。

用 01 序列 $\{a_i\}_{i=1}^m$ 来表示边的选取情况。若编号为 i 的边被选，则 $a_i = 1$ ；否则， $a_i = 0$ 。

你的选边方案必须满足：

- 保留被选的边而删除不选的边后，所有节点的度数都是奇数。
 - 这条要求也有可能无法被满足，此时无解。
- 在满足上条要求的前提下，使 a 的字典序最大。

对于任意两个 01 序列 $\{b_1\}_{i=1}^m$ 和 $\{b_2\}_{i=1}^m$ ，称 b_1 的字典序大于 b_2 ，当且仅当存在整数 $i \in [1, m]$ 满足以下两个条件：对于任意整数 $j \in [1, i)$ ，都有 $b_{1,j} = b_{2,j}$ ； $b_{1,i} = 1$ 且 $b_{2,i} = 0$ 。

显然，全体长度为 m 的 01 序列在字典序比较下构成严格全序集。

输入格式

第一行两个整数 n, m ($1 \leq n, m \leq 10^6$)。

接下来 m 行中的第 i 行，有两个整数 u_i, v_i ($1 \leq u_i, v_i \leq n$)，表示编号为 i 的边连接节点 u_i 和 v_i 。

保证图中无自环，即 $u_i \neq v_i$ 。

输出格式

若无解，输出 -1。

否则，输出一个长度为 m 的 01 序列（一行 m 个整数，中间用一个空格隔开），表示你的选边方案。

样例 1

输入

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

输出

```
1 0 1 0
```

解释

若不要使 a 的字典序最大，方案还可以是选取编号为 2 和 4 的边。

样例 2

输入

```
3 2
1 2
2 3
```

输出

```
-1
```

I. 心意不尽，前程已近

时间限制： 1.0 s

内存限制： 256 MiB

题目描述

给定一张由 n 个节点、 m_1 条未定向的边和 m_2 条已定向的边构成的图。节点的编号是 $1, 2, \dots, n$ 。

这张图满足：任取点集的一个子集，两端都在子集内的边（无论是否已定向）的条数不超过子集的大小。

求有多少个把未定向的边定向的方案，满足定向后图中不存在有向环。答案对 $10^9 + 7$ 取模。

输入格式

第一行三个整数 n, m_1, m_2 ($1 \leq n, m_1, m_2 \leq 2 \times 10^5$, $m_1 + m_2 \leq 2 \times 10^5$)。

接下来 m_1 行，每行两个整数 u_i, v_i ($1 \leq u_i, v_i \leq n$)，表示一条连接 u_i 和 v_i 的未定向的边。

接下来 m_2 行，每行两个整数 u'_i, v'_i ($1 \leq u'_i, v'_i \leq n$)，表示一条从 u'_i 连向 v'_i 的已定向的边。

保证图中无自环，即： $u_i \neq v_i$, $u'_i \neq v'_i$ 。

保证：在图中任取两个节点，它们之间至多只有一条边（无论是否已定向）。

输出格式

一行一个整数，表示答案对 $10^9 + 7$ 取模后的结果。

样例

输入

```
10 6 2
1 2
2 3
3 4
3 5
6 7
7 8
4 1
8 9
```

输出

```
56
```

J. 新鲜的日子和可爱的日子

时间限制： 0.5 s

内存限制： 256 MiB

题目描述

我总反复回忆过去的日子，怕稍不留意，那些鲜活的片段便悄然逝去。

新鲜的日子像新剥的橘瓣，落在舌尖，酸甜的汁水在齿间漫开，连呼吸都带着清爽的果香。阳光也落在掌心，软暖如一团绒，沿着指缝缓缓滚开。想把它攥紧，它已轻轻滑落，化作尘间微光。

可爱的日子像绵软的暖风，浸在午后，衣领上浅浅地沾有暖意。它掠过那柔和的背影，绕着打转，惊起花叶细响，然后又放慢了脚步，只肯贴着衣角徐徐地游。周遭的光斑悄悄晕开一圈软痕，空气里漾着温存。

我怎么舍得把这些悉数收进过去的抽屉，任由鲜活逐渐淡成旧影呢？我知道日子终要向前，但仍忍不住盼着，慢一点，再慢一点，让我多看一眼，每个日子的模样。

小 L 回忆了 n 个日子，并用一个大写英文字母来形容每个日子。

小 L 把字母依次写下来，得到字符串 $S_1S_2\cdots S_n$ （每个 S_i 都表示一个大写英文字母）。

称第 i 个日子是**新鲜的日子**，当且仅当 $S_i \neq S_{i-1}$ 和 $S_i \neq S_{i+1}$ 这两个条件中恰有一个成立。

称第 i 个日子是**可爱的日子**，当且仅当 $S_i \neq S_{i-1}$ 和 $S_i \neq S_{i+1}$ 这两个条件都不成立。

特别需要注意的是，在这里我们认为 $S_1 \neq S_0$ ， $S_n \neq S_{n+1}$ 。

小 L 数了数，发现有 m_1 个新鲜的日子和 m_2 个可爱的日子。

时光荏苒，小 L 忘记了 $S_1S_2\cdots S_n$ ，只对 n, m_1, m_2 还有模糊的印象。

给定 n, m_1, m_2 ，判定小 L 可能记对了，还是一定记错了。

若小 L 可能记对了，告诉他任意一个可能的 $S_1S_2\cdots S_n$ 。

输入格式

第一行一个整数 T ($T \geq 1$)，表示该测试点有 T 组数据。

接下来，对于每组数据，输入一行三个整数 n, m_1, m_2 ($n \geq 1$, $m_1, m_2 \geq 0$, $m_1 + m_2 \leq n$)。

保证 T 组数据中的 n 之和 $\sum \leq 10^6$ 。

输出格式

对于每组数据，输出一或两行：

第一行输出 **Possible** 或 **Impossible**，分别表示小 L 可能记对了和一定记错了。

若小 L 可能记对了，第二行输出一个长度为 n 、由大写英文字母构成的字符串 $S_1S_2\cdots S_n$ 。

这种情况下，你需要保证该字符串满足题目描述中的要求。若有多个解，只需输出任意一个。

若小 L 一定记错了，该组数据只输出一行。

样例 1

输入

```
1
10 6 3
```

一个满足要求的输出

```
Possible
ABBCCDDDD
```

解释

下面，用蓝色标出新鲜的日子，用绿色标出可爱的日子：

A B B C C C D D D D

样例 2

输入

```
8
1 0 0
5 2 3
6 0 0
6 4 1
8 6 2
4 2 0
1 1 0
4 0 3
```

一个满足要求的输出

```
Possible
A
Possible
AAAAA
Possible
ABABAB
Possible
AABBBA
Possible
AABBBBAA
Possible
YXXY
Impossible
Impossible
```

K. 无聊的猪宝宝

时间限制： 1.0 s

内存限制： 256 MiB

题目描述

无聊的猪宝宝有一个长度为 n 的整数序列 $a_1, a_2, \dots, a_n$ 。

它想将其划分成若干段，设为 $[l_1, r_1], [l_2, r_2], \dots, [l_m, r_m]$ (m 自行决定)，满足：

- $l_1 = 1, r_m = n$ 。对于任意整数 $i \in [1, m]$ ， $l_i \leq r_i$ ，若 $i \neq m$ 则 $r_i + 1 = l_{i+1}$ 。
- 以下两个条件中至少有一个成立：
 - 条件 A：对于任意整数 $i \in [1, m]$ ，都有 $a_{l_i} \leq a_{l_i+1} \leq \dots \leq a_{r_i}$ 。
 - 条件 B：对于任意整数 $i \in [1, m]$ ，都有 $a_{l_i} \geq a_{l_i+1} \geq \dots \geq a_{r_i}$ 。

求满足上述要求的划分方案的个数。答案对 $10^9 + 7$ 取模。

输入格式

第一行一个整数 n ($1 \leq n \leq 10^5$)，表示序列的长度。

第二行 n 个整数 $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^9$)。

输出格式

一行一个整数，表示答案对 $10^9 + 7$ 取模后的结果。

样例 1

输入

```
6
1 2 3 2 2 1
```

输出

```
14
```

解释

可以用两个集合的容斥原理来计算这个样例。

把 $a_1, a_2, \dots, a_6$ 划分成 m 段，相当于从 5 个邻项之间的间隙中选 $m - 1$ 个插入隔板。

要使条件 A 成立，必须且只需在 a_3 与 a_4 之间、 a_5 与 a_6 之间插入隔板，可得方案数为 $2^3 = 8$ 。

要使条件 B 成立，必须且只需在 a_1 与 a_2 之间、 a_2 与 a_3 之间插入隔板，可得方案数为 $2^3 = 8$ 。

要使条件 A 和 B 同时成立，同理可得方案数为 2。

所以，答案是 $8 + 8 - 2 = 14$ 。

样例 2

输入

31
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

输出

73741817

解释

显然，答案是 $2^{30} = 1073741824$ ，对 $10^9 + 7$ 取模后是 73741817。

样例 3

输入

12
1 1 2 3 2 2 1 999999999 1000000000 2023 2024 2025

输出

284

I. 量子计算程序设计

时间限制： 1.0 s

内存限制： 256 MiB

题目描述

本题没有输入，你的代码只需输出解决方案。

背景

一个量子比特 $|\psi\rangle$ 的状态可用 $\alpha|0\rangle + \beta|1\rangle$ 描述。其中， α, β 是复数，且满足 $|\alpha|^2 + |\beta|^2 = 1$ 。

在这里，我们只把量子比特看作形式化的组合对象，而不考虑它的物理意义。

若对 $|\psi\rangle$ 进行测量：

- 有 $|\alpha|^2$ 的概率测得 0，且 $|\psi\rangle$ 将坍缩为 $|0\rangle$ （即 $1 \cdot |0\rangle + 0 \cdot |1\rangle$ ）。
- 有 $|\beta|^2$ 的概率测得 1，且 $|\psi\rangle$ 将坍缩为 $|1\rangle$ （即 $0 \cdot |0\rangle + 1 \cdot |1\rangle$ ）。

对于多个量子比特，例如 $|\psi_1\rangle = \alpha|0\rangle + \beta|1\rangle$ 和 $|\psi_2\rangle = \gamma|0\rangle + \delta|1\rangle$ ，其复合系统的状态是

$$\begin{aligned} |\psi_1\rangle|\psi_2\rangle &= (\alpha|0\rangle + \beta|1\rangle)(\gamma|0\rangle + \delta|1\rangle) \\ &= \alpha\gamma|0\rangle|0\rangle + \alpha\delta|0\rangle|1\rangle + \beta\gamma|1\rangle|0\rangle + \beta\delta|1\rangle|1\rangle \end{aligned}$$

易知 $|\alpha\gamma|^2 + |\alpha\delta|^2 + |\beta\gamma|^2 + |\beta\delta|^2 = 1$ 。

不同的量子比特之间并非总是可分离的。考虑一般的双量子比特系统的状态

$$|\psi_1\rangle|\psi_2\rangle = \alpha_{00}|0\rangle|0\rangle + \alpha_{01}|0\rangle|1\rangle + \alpha_{10}|1\rangle|0\rangle + \alpha_{11}|1\rangle|1\rangle$$

其中， $|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$ 。若上式无法被分解为 $(\alpha|0\rangle + \beta|1\rangle)(\gamma|0\rangle + \delta|1\rangle)$ ，则称 $|\psi_1\rangle$ 和 $|\psi_2\rangle$ 处于纠缠态。纠缠态形成后，即使两个量子比特在空间上远离，它们之间的关联也仍然存在。

若对该系统中的 $|\psi_1\rangle$ 进行测量：

- 有 $|\alpha_{00}|^2 + |\alpha_{01}|^2$ 的概率测得 0，且系统将坍缩为 $\frac{|0\rangle(\alpha_{00}|0\rangle + \alpha_{01}|1\rangle)}{\sqrt{|\alpha_{00}|^2 + |\alpha_{01}|^2}}$ 。
- 有 $|\alpha_{10}|^2 + |\alpha_{11}|^2$ 的概率测得 1，且系统将坍缩为 $\frac{|1\rangle(\alpha_{10}|0\rangle + \alpha_{11}|1\rangle)}{\sqrt{|\alpha_{10}|^2 + |\alpha_{11}|^2}}$ 。

对该系统中的 $|\psi_2\rangle$ 进行测量同理。 $|\psi_1\rangle$ 和 $|\psi_2\rangle$ 的地位平等，只不过规定了顺序。

量子门是操作量子比特的基本线路。在本题中，可能用到且只能应用以下四种量子门：

H 门：设目标比特为 $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ ，令 $|\psi\rangle \leftarrow \alpha\frac{|0\rangle + |1\rangle}{\sqrt{2}} + \beta\frac{|0\rangle - |1\rangle}{\sqrt{2}}$ 。

X 门：设目标比特为 $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ ，令 $|\psi\rangle \leftarrow \alpha|1\rangle + \beta|0\rangle$ 。

Z 门：设目标比特为 $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ ，令 $|\psi\rangle \leftarrow \alpha|0\rangle - \beta|1\rangle$ 。

CNOT 门:

- 设控制比特为 $|\psi_1\rangle$, 目标比特为 $|\psi_2\rangle$,
- 操作前 $|\psi_1\rangle|\psi_2\rangle = \alpha_{00}|0\rangle|0\rangle + \alpha_{01}|0\rangle|1\rangle + \alpha_{10}|1\rangle|0\rangle + \alpha_{11}|1\rangle|1\rangle$,
- 令 $|\psi_1\rangle|\psi_2\rangle \leftarrow \alpha_{00}|0\rangle|0\rangle + \alpha_{01}|0\rangle|1\rangle + \alpha_{10}|1\rangle|1\rangle + \alpha_{11}|1\rangle|0\rangle$ 。

注:

应用量子门对所涉及的量子比特与其它量子比特之间的纠缠情况不作要求。

例如, 假设 $|\psi_1\rangle|\psi_2\rangle|\psi_3\rangle = |0\rangle|0\rangle|1\rangle + |1\rangle|1\rangle|0\rangle$ 。

若以 $|\psi_3\rangle$ 为控制比特, $|\psi_1\rangle$ 为目标比特, 则结果是 $|\psi_1\rangle|\psi_2\rangle|\psi_3\rangle = |1\rangle|0\rangle|1\rangle + |1\rangle|1\rangle|0\rangle$ 。

设定

小 L 学习量子计算的初步知识后, 暗想: 何意味?

于是, 他设计了 $\mathcal{H}\mathcal{Y}\mathcal{W}$ 语言——一种极其简单的量子计算程序设计语言。

一个 $\mathcal{H}\mathcal{Y}\mathcal{W}$ 程序由若干条语句构成, 每条语句占据一行。解释器将从前到后依次执行语句。

语句分为测量语句和操作语句。

测量语句的格式是 `measure  $x$` , 表示对 $|\psi_x\rangle$ 进行测量, 结果为 0 或 1。

操作语句分两种:

格式	含义
<code><statement></code>	执行 <code><statement></code>
<code>if <condition> then <statement></code>	若 <code><condition></code> 为真, 执行 <code><statement></code>

`<statement>` 分四种:

格式	含义
<code>H x</code>	以 $ \psi_x\rangle$ 为目标比特, 应用 H 门
<code>X x</code>	以 $ \psi_x\rangle$ 为目标比特, 应用 X 门
<code>Z x</code>	以 $ \psi_x\rangle$ 为目标比特, 应用 Z 门
<code>CNOT x y</code>	以 $ \psi_x\rangle$ 为控制比特, $ \psi_y\rangle$ 为目标比特, 应用 CNOT 门

`<condition>` 的格式是 `m $x$ = $y$` , 表示「第 x 次测量 (measure) 的结果是 y 」。

例如, 初始时 $|\psi_1\rangle|\psi_2\rangle = |0\rangle|0\rangle$, 要使 $|\psi_1\rangle|\psi_2\rangle = \frac{|0\rangle|0\rangle + |1\rangle|1\rangle}{\sqrt{2}}$, $\mathcal{H}\mathcal{Y}\mathcal{W}$ 程序可以是

```
H 1
CNOT 1 2
```

执行每条语句后的情况如下:

1. $|\psi_1\rangle|\psi_2\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}|0\rangle$ 。
2. $|\psi_1\rangle|\psi_2\rangle = \frac{|0\rangle|0\rangle + |1\rangle|1\rangle}{\sqrt{2}}$ 。

此后，要使 $|\psi_1\rangle|\psi_2\rangle$ 变回 $|0\rangle|0\rangle$ ， $\mathcal{H}\mathcal{Y}\mathcal{W}$ 程序可以是

```
measure 1
if m1=1 then X 1
if m1=1 then X 2
```

执行每条语句后的情况如下：

1. 可能是 $|\psi_1\rangle|\psi_2\rangle = |0\rangle|0\rangle$ （测得 0），也可能是 $|\psi_1\rangle|\psi_2\rangle = |1\rangle|1\rangle$ （测得 1）。
2. 可能是 $|\psi_1\rangle|\psi_2\rangle = |0\rangle|0\rangle$ ，也可能是 $|\psi_1\rangle|\psi_2\rangle = |0\rangle|1\rangle$ 。
3. $|\psi_1\rangle|\psi_2\rangle = |0\rangle|0\rangle$ 。

任务

现在，考虑这样一个问题：

初始时， $|\psi_1\rangle = \alpha|0\rangle + \beta|1\rangle$ ， $|\psi_2\rangle = |0\rangle$ ， $|\psi_3\rangle = |0\rangle$ 。

我们的目标是使： $|\psi_1\rangle = |0\rangle$ ， $|\psi_2\rangle = |0\rangle$ ， $|\psi_3\rangle = \alpha|0\rangle + \beta|1\rangle$ 。

你需要设计一个不超过 16 条语句的 $\mathcal{H}\mathcal{Y}\mathcal{W}$ 程序，实现这一目标。

输入格式

本题没有输入。

输出格式

输出你的解决方案。不要输出多余的内容。

样例

假设你的 $\mathcal{H}\mathcal{Y}\mathcal{W}$ 程序是

```
H 1
CNOT 1 2
```

你可以提交以下 C++ 代码：

```
#include <bits/stdc++.h>
int main() {
    printf(
        "H 1\n"
        "CNOT 1 2"
    );
    return 0;
```

}

当然，你也会获得一发罚时。