

password

仓鼠大学发生了一件大事，学校中凭空出现了一扇神秘的传送门，传说只要能进去这扇门，并通过里面的关卡，便能在ACM比赛中所向披靡。起床和水饺相约一起去闯关。

然而这扇门上需要密码，只有正确答出密码，才能进入。门上写了若干数字，每个数字都是不大于15的非负整数，密码就藏在这些数字里。解码的方式为，对于给出的数字，按照二进制展开后，第一位作为校验位，后三位为密码位，如果后三位中1的个数为奇数，那么校验位应该为1，如果后三位中1的个数为偶数，那么校验位应该为0，如果给出的数字符合这个条件，那么就算入密码，否则就不算入密码。

按顺序给出若干个数字后，将符合条件的数字的密码位转为十进制拼接起来，就可得到正确的密码。

例如有四个数字:15,8,9,3，数字15，二进制展开后为1111，后三位有3个1，是奇数，且第一位也为1，符合条件，那么111转为十进制是7，算做是密码的第一位。数字8展开后为1000，不满足条件。数字9为1001，数字3为0011，都满足条件，将001和011转为十进制是1和3，最后得到密码为713。

input

第一行给出一个数字N, ($1 \leq N \leq 100$)，下一行有N个整数，每个数字都是不大于15的非负整数。

output

输出一行，根据给出的数字得到的密码。

样例输入1

复制

```
4
15 8 9 3
```

样例输出1

复制

```
713
```

样例输入2

复制

```
6
12 14 2 6 5 0
```

样例输出2

复制

4650

watcher

起床和水饺成功进入了传送门，发现自己来到了一座山脚下，它们便顺着路一直向上，直到山顶，山顶一位看守人迎接了它们。看守人说，我知道你们上山来了，就在此等候你们，现在我需要你们回答一个问题，答对了便可继续前行，到山顶大殿里去。

这座山上的路和路口构成了一个满二叉树，路口为节点，路为边。山顶是根节点1，然后向下有两条路，分别通向路口2和路口3。对于路口*i*来说，向下通向路口*i**2和路口*i**2+1。这棵树的深度为*N*，所以山脚共有 $2^{(N-1)}$ 个入口。除了山顶以外，我在若干路口（包括山脚入口）放置了魔法阵，如果有人经过便会触发魔法阵，通知我有人上山了。放置魔法阵后，不管从山脚哪个入口上山一直向上的话，都能确保会触发且只会触发一个魔法阵。

现在我告诉你们树的深度和你们上山时触发魔法阵的路口序号，你们猜猜我最多和最少放置了多少魔法阵。

input

输入一行包含两个整数*N,M*，($2 \leq N \leq 30$, $2 \leq M < 2^N$)表示树的深度和触发魔法阵的路口序号。

output

输出一行包含两个整数，表示看守人最多和最少放置了多少魔法阵。

样例输入1

复制

4 2

样例输出1

复制

5 2

样例输入2

复制

4 15

样例输出2

复制

8 4

decompose

起床和水饺进入了大殿，一个老者正坐在中央，而他的上方悬浮着一个数字，随着老者双手的挥动，数字变成了若干个更小的数字。

老者说，你们知道质数吗，一个质数是不可由两个大于1的整数相乘得到的，所以质数又叫做prime，有起源的含义。而质数相乘则构成了其他千千万万无限的整数，若把一个整数分解为若干个质数，可以由这些质数相乘得到原来的整数，这个过程就叫做质因数分解。例如数字 $270=2*3*3*3*5$ ，则270可以分解为2,3,5。

我在这里不断地进行质因数分解，希望有朝一日能找到所有整数的本源。虽然我有无限的时间，但整数也是无限的，我不知何时能够完成。我先给你们N个整数，你们来帮我分解，我好偷个懒休息一下。等你们完成任务后，我就给你们打开去后山的门。

input

输入第一行有一个整数N,($1 \leq N \leq 10000$),接下来N行，每行一个整数a,
($2 \leq a \leq 100000000$)，表示要分解的整数。

output

对于每个要分解的整数，输出一行，包含质因数分解后产生的若干质数，从小到大排好，整数间以空格分隔。

样例输入

[复制](#)

```
3
8
23
54
```

样例输出

[复制](#)

```
2
23
2 3
```

cableway

来到了后山，索道管理员领着起床和水饺来到了索道站。它们面对着的是一座座连绵起伏，各不相同的山峰。

管理员说，你们下一步将通过这些索道，到最远的那座山上去，但是现在还有个问题没解决。这里共有 N 座山峰，序号为 $(1-N)$ ，无限的时间以来这里修了 M 条索道，每条索道都连接着 u,v 两座山峰。两座山峰是连通的当有索道直接连接两座山峰，或者可以经由其他索道间接连接。如1连接2，2连接3，那么 $(1,2),(2,3),(1,3)$ 都是连通的。

我告诉你山峰的个数，和已经修建好的索道，你帮我计算一下最少还需要修建多少条索道，才能让所有两座山峰之间都连通。

input

第一行两个整数 N,M ， $(1 \leq N \leq 10000, 1 \leq M \leq 100000)$ ，表示山峰个数和修建索道个数。接下来 M 行，每行两个整数 u,v ， $(1 \leq u,v \leq N, u \neq v)$ 表示索道连接的两座山峰序号。

output

输出一行一个整数，表示最少还需要修建索道的个数。

样例输入

[复制](#)

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

样例输出

[复制](#)

```
2
```

pancake

终于，起床和水饺到达了最后的那座山峰，经过了长途跋涉，它们的肚子早就饿的咕咕叫了，正好碰到了一家煎饼店。

煎饼店老板会做圆形的煎饼，但是每次做出来的煎饼都大小各异，做好的煎饼都放在一个盘子里，后来的煎饼会放在之前的煎饼上面。起床和水饺每次都会拿走最上面的煎饼去吃，但是这可不是白吃的，老板要求它们在拿之前先说出目前盘子里最大的煎饼的大小。

首先输入整数 N ，表示接下来操作的次数。然后有两种操作：

(1) 1 R : (R 为整数)

老板将一个半径为 R 的煎饼放在盘子最上方。

(2) 2

起床和水饺要答出目前盘子里最大煎饼的半径大小，然后拿走最上面的煎饼。

input

输入第一行一个整数 N ，($1 \leq N \leq 1000000$)表示操作的次数。接下来 N 行，每行一个操作。
($1 \leq R \leq 100000000$)。保证输入不会在盘子为空时去拿煎饼。数据量较大，请尽量使用更有效率的输入输出方式。

ouput

在每个(2)操作之前，输出一行，表示当前盘子里最大煎饼的半径大小。

样例输入

[复制](#)

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

样例输出

[复制](#)

3
3
1

exfibonacci

吃饱肚子后，起床水饺沿着道路继续前行，直到来到一座殿堂前。又一位老者站在殿堂的大门前，说道，你们好，我是斐波那契，这次由我来把守知识殿堂的大门。

相信你们一定听说过斐波那契数列吧，我自己也因为我的这个发明而名声远扬。今天我在这里无聊的时候，想到了一个新的数列，我把它叫做增强斐波那契数列。它的第一项 $F(1)=1$ ，第二项 $F(2)=1$ ，之后的每一项，都为之前所有项的和， $F(n)=F(1)+\dots+F(n-1)$ ， $(n \geq 3)$ 。

你们就来回答以下它的第 n 项是多少吧，答对了我就让你们进去，答案可能过大，所以只要输出 $F(n)\%1000000007$ 的结果即可。

input

输入第一行包含一个整数 T ， $(1 \leq T \leq 100000)$ ，表示询问的个数。接下来 T 行，每行一个整数 n ， $(1 \leq n \leq 1000000000000)$ 表示询问增强斐波那契数列的第 n 项。

output

对于每个询问，输出一行包含一个整数，表示 $F(n)\%1000000007$ 的结果。

样例输入

[复制](#)

```
3
1
3
10000
```

样例输出

[复制](#)

```
1
2
476402953
```


knowledge

成功答对了问题后，斐波那契领着它们进入了知识殿堂，这里有无穷无尽的知识，排成一排悬浮在空中，发出金色的光辉。

斐波那契说，这里都是无限的时间以来，数不清的人努力思考的结晶，你们今天有幸来到这里，有机会拿走一部分，让自己变得更加渊博。但是你们能否能成功吸收这些知识也要看自身的能力，目前这里有 N 个知识按顺序排成一排，每个知识对你们来说有个数值 $a[i]$ ， $a[i]$ 可正可负，也可能为0，你们最多可以拿走其中连续的 K 个知识，它们的数值相加起来就是你们可得到的知识数量。这对起床和水饺来说也是一个考验，它们想知道最多可以得到的知识数量为多少。

input

输入第一行有两个整数 N, K ，($1 \leq N \leq 10000$, $1 \leq K \leq 1000$)。接下来一行包含空格分隔的 N 个整数，第 i 个整数为 $a[i]$ ，($-10000 \leq a[i] \leq 10000$)表示从左到右第 i 个知识的数值。

output

输出一行包含一个整数，表示最多可以得到的知识数量。

样例输入1

[复制](#)

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

样例输出1

[复制](#)

```
5
```

样例输入2

[复制](#)

```
5 5
2 0 2 -2 -2
```

样例输出2

[复制](#)

```
4
```

escape

拿到知识后，殿堂突然震动起来，斐波那契说，你们拿走了知识，这个世界很快就要关闭了，你们快点到后面的传送门去，就可以离开这里回到你们的世界。

起床和水饺跑到了殿堂后面，发现一个巨大的深渊和排成一行在深渊上悬空的石块，最后一个石块上就是传送门。总共有 N 个石块，起床和水饺一开始站在第一个石块上，传送门位于第 N 个石块上。它们每次只能向前跳跃，最少跳至下一个石块，且最多能跳至向前第 K 个石块上。还有 M 个石块太小是无法落脚的，它们不能跳到这些石块上。

与此同时，这个世界的时空开始混乱，无限多的平行世界汇聚在了这里，无限的起床和水饺都在跳跃前往传送门。所以它们需要知道总共有多少种不同的跳法，可以从第一个石块跳到第 N 个石块上，(两个跳法是不同的，当它们的跳跃次数不同或者其中任意一步跳跃距离不同)。不同的跳法数量可能过大，所以只需求数量对 1000000007 取模后的结果。

input

输入第一行包含三个整数 N, K, M ，($2 \leq N \leq 10000$, $1 \leq K \leq 1000$, $0 \leq M \leq 100$)含义如题目所说。接下来一行有 M 个整数，表示不能落脚的石块，保证第一个和第 N 个石块一定可以落脚。

output

输出一行包含一个整数，表示不同跳法的个数对 1000000007 取模后的结果。

样例输入

[复制](#)

```
5 2 2
2 4
```

样例输出

[复制](#)

```
1
```

database

起床被拉去写一个简单的数据库程序。数据库可以看作是一个字符串列表，里面按顺序存放着多个字符串。有以下几种对列表的操作方式：

(1)insert P S: (P为整数, S为字符串)

将列表第P个位置及以后的字符串后移一位，然后将字符串S放到P位置。

(2)delete P: (P为整数)

删除列表第P个位置的字符串，然后将P位置之后的字符串前移一位。

(3)sort A B: (A和B为整数)

将A位置到B位置（包括A和B位置）的字符串按照字典序排序，排序时大小写敏感，且大写字母的字典序比小写字母的更小。

(4)find S: (S为字符串)

查找列表中包含S的字符串，并按照列表顺序输出，查找时大小写不敏感。

一开始列表为空，如果当前列表中有N个字符串，那么他们的位置为从1到N，保证insert操作中 $1 \leq P \leq N+1$ ，delete操作中 $1 \leq P \leq N$ ，sort操作中 $1 \leq A \leq B \leq N$ 。题目中的字符串长度都不超过20，且只包含大小写英文字母。

input

输入第一行一个整数T, ($1 \leq T \leq 1000$)表示接下来的操作个数。接下来T行，每行为一个操作。

output

处理每个操作，对于find操作，按列表顺序输出查找到的字符串，每个占一行。

样例输入

[复制](#)

```
10
insert 1 helloa
insert 1 hellob
find hello
delete 1
find hello
insert 2 acm
insert 3 ACM
find acm
sort 1 3
find acm
```

样例输出

[复制](#)

```
hellob
helloa
helloa
acm
ACM
ACM
acm
```

cave

某天起床和水饺一起爬山，爬到一半时，没注意脚下，起床和水饺突然都掉进了一个山洞里。它们俩掉到了不同的位置，且水饺不幸摔断了腿，起床还可以走动，起床需要找到路去往水饺那里，然后带着水饺从出口离开山洞。

整个山洞处于一个三维坐标系上，X轴和Y轴垂直，且分布在水平面上，Z轴和X轴Y轴都垂直，且正方向向上。山洞是一个边长为N的立方体大空间，则可以按坐标系划分为 $N*N*N$ 个空间，每个空间大小都为 $1*1*1$ ，可以用 (x,y,z) 表示其所处位置。每个空间可以是空的，也可以是岩石，(起床和水饺需要位于空的空间里，且空间下方相邻的那个空间需要是岩石，通俗的讲，就是它们需要站在岩石上，不能悬空，且不能处于有岩石的地方)，以上是仓鼠所处的位置条件。

起床自己移动时，每次移动可以选择向X轴和Y轴上的四个方向中的一个移动一格，且每次最多可以向上移动一格，向下移动两格，且保持它的位置条件。通俗的讲，就是当它水平移动碰到岩石时，要向上爬到岩石上，且每次最高爬一格，如果超过一格，或者当前上方也为岩石（无法穿过岩石），视为不能向这个方向移动；当脚下为空时，会下落到下方第一个岩石上，且最大下落不超过两格，否则会摔伤，如果超过两格，或者脚下没有岩石了，则视为不能向这个方向移动。

当起床带上水饺时，移动方式不变，但是不能向上移动了，且每次最多向下移动一格，并保持位置条件。通俗的讲，就是它无法爬到更高的岩石上，且下落不超过一格，否则会摔伤，如果不满足这些要求，就视为这个方向不能移动。

输入会给出洞穴大小N，以及洞穴中岩石数目M，和这M块岩石每个所处的空间位置 (x,y,z) ，岩石可以违反重力悬空。然后给出起床和水饺一开始的空间位置 (x_0,y_0,z_0) 和 (x_1,y_1,z_1) ，保证它们一开始都处在合适的位置，满足位置条件。然后是出口数目E，以及每个出口的空间位置 (x,y,z) ，当起床带着水饺移动到任意一个出口位置时，即可离开山洞。

起床在移动中不能移动至山洞外($1 \leq x,y,z \leq N$)，也不能上下斜着穿过岩石，可以经过出口位置而不离开山洞，保证起床可以移动到达水饺的位置，且可以带着水饺移动到出口位置。输出起床从开始到带着水饺到达出口位置最少需要移动多少次。

input

输入第一行有两个整数N,M, ($1 \leq N \leq 100$, $1 \leq M \leq 1000000$)含义如题目所说。接下来M行，每行三个整数 (x,y,z) ，描述每个岩石所处的位置。接下来一行，有六个整数 (x_0,y_0,z_0) , (x_1,y_1,z_1) ，代表起床和水饺一开始的位置。接下来一行一个整数E, ($1 \leq E \leq 10$),表示出口数量，再接下来E行，每行三个整数 (x,y,z) ，描述每个出口位置。保证以上给出的所有位置中没有两个是相同的，且($1 \leq x \leq N$, $1 \leq y \leq N$, $1 \leq z \leq N$)。

output

输出一行一个整数，表示起床要救出水饺的最少移动次数。

样例输入

复制

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

样例输出

复制

```
4
```