

月球异或

时间限制：10.0s 内存限制：512M

题目描述

在「月读」的世界中，有一种定义在自然数意义下的二元运算，我们称其为「月球异或」，记作 \circledast 。

「月球异或」相当于对两个数字三进制下的每一位，独立相加后模 3 得到运算结果在三进制下该位的值，比如 $8 = 2 \times 3^0 + 2 \times 3^1$, $16 = 1 \times 3^0 + 2 \times 3^1 + 1 \times 3^2$, 那么

$8 \circledast 16 = 0 \times 3^0 + 1 \times 3^1 + 1 \times 3^2 = 12$, 以下是更严谨的定义：

对于自然数 x, y , 可以唯一的表示为 $x = \sum_{0 \leq i} 3^i a_i$, $y = \sum_{0 \leq i} 3^i b_i$ 的形式 ($\forall_{0 \leq i} a_i, b_i \in \{0, 1, 2\}$)。

我们定义 $x \circledast y = \sum_{0 \leq i} 3^i ((a_i + b_i) \% 3)$ 。

容易证明该二元运算具有交换律和结合律。

有一个长度为 N 的自然数序列 v_1, v_2, \dots, v_N , 你可以做任意次（包含零次）如下操作：选择一个序号 i ($1 \leq i \leq N$), 然后令 $v_i \leftarrow v_i \circledast v_i$ 或者 $v_i \leftarrow 0$ 。

八千代每次会询问一个自然数 s , 你需要回答能否通过若干次操作使得 $v_1 \circledast v_2 \circledast \dots \circledast v_N = s$, **注意询问是独立的**, 你不会真的去做这些操作。

输入格式

第一行输入一个整数 T 表示测试数据组数。

对于每组数据：

第一行两个正整数 N, Q , 表示序列长度和询问次数。

接下来一行 N 个正整数 v_1, v_2, \dots, v_N 。

接下来 Q 个正整数 s_1, s_2, \dots, s_Q 表示每次询问的值。

令 N_{tot} 表示所有测试组 N 之和, Q_{tot} 表示所有测试组 Q 之和。

$1 \leq T \leq 100000, 1 \leq N_{tot}, Q_{tot} \leq 2 \times 10^6$ 。

对于每组数据：

$1 \leq N, Q \leq 10^6, 0 \leq v_i, s_i \leq 10^{18}$ 。

输出格式

对于每组数据：

输出 Q 行，对于每组询问，如果可行，输出单行字符串 "Yes"，否则输出单行字符串 "No"。

样例输入

```
3
1 4
1
0 1 2 3
3 5
0 5 10
4 5 6 7 8
3 6
8 16 32
12 17 57 41 4 44
```

样例输出

Yes

Yes

Yes

No

No

Yes

No

Yes

No

Yes

No

Yes

No

Yes

Yes