

庭中有奇树 / Tree

时间限制：3.0s 内存限制：512M

时间限制：3.0 秒 空间限制：512 MiB

题目描述

mej 先生的庭院里有一棵奇树。

这棵树上每个点都有一个正整数编号，且以编号为 1 的点为根，所有正整数都作为点的编号在这棵树上出现过。同时，编号为 i 的点恰好有 i 个子节点，并且这 i 个子节点的编号是连续的：记 $mn(i)$ 为 i 号节点的子节点中编号最小的、 $mx(i)$ 为最大的，则 $mx(i) - mn(i) = i - 1$ 且 $mn(i) \dots mx(i)$ 中所有数都恰好出现一次。不仅如此，对于任意的 $1 \leq i < j$ ，都有 $mx(i) < mn(j)$ 。

可以发现以上性质唯一确定了这棵奇树。例如，1 号节点的子节点有 2，2 号节点的子节点有 3, 4，3 号节点的子节点有 5, 6, 7，等等。不过因为 mej 先生不喜欢无限大的树，所以他只保留了这棵树上编号在 1 到 n 之间的节点。

mej 先生可以从这棵奇树上获取法力。具体来说，树上每个节点都有一个魔力值，初始时所有节点的魔力值都为 0；他可以选择一个节点 x 开始获取法力，此时他将会获得 x 的子树中（包括 x ）所有节点魔力值的异或和的法力。同时他还会对这棵树进行维护，每次他会选择一个节点 x 和一个值 c ，然后施加法术，将 x 的子树中所有节点的魔力值按位或上 c 。

现在，mej 先生一共进行了 q 次操作，每次操作可能是维护奇树或者获取法力。他想知道，他每次获取法力时获取的法力值是多少。

输入格式

从标准输入读入数据。

第一行输入两个正整数 n, q ($1 \leq n \leq 10^{18}, 1 \leq q \leq 10^6$)，表示树的大小和操作次数。

接下来 q 行，每行先输入一个正整数 op ($op \in \{1, 2\}$)，表示操作类型。若 $op = 1$ ，则再输入两个整数 x, c ($1 \leq x \leq n, 1 \leq c < 2^{60}$)，表示将以 x 为根的子树中所有节点的魔力值按位或 c ；若 $op = 2$ ，则再输入一个正整数 x ，表示查询以 x 为根的子树中所有节点魔力值的异或和。

输出格式

输出到标准输出。

对于每次查询，输出一个整数，表示所求的异或和。

样例

样例 1 输入

```
11 6
1 3 931
1 4 209
1 2 28
2 1
1 8 287
2 4
```

样例 1 输出

```
193
479
```

样例 1 解释

初始时所有节点的魔力值都为 0。

第一次操作，将以 3 为根的子树中所有节点的魔力值按位或 931，所有节点的魔力值序列变为 0, 0, 931, 0, 931, 931, 931, 0, 0, 0, 0；

第二次操作，将以 4 为根的子树中所有节点的魔力值按位或 209，所有节点的魔力值序列变为 0, 0, 931, 209, 931, 931, 931, 209, 209, 209, 209；

第三次操作，将以 2 为根的子树中所有节点的魔力值按位或 28，所有节点的魔力值序列变为 0, 28, 959, 221, 959, 959, 959, 221, 221, 221, 221；

第四次操作，查询以 1 为根的子树中所有节点的魔力值的异或和，即 $0 \oplus 28 \oplus 959 \oplus 959 \oplus 959 \oplus 959 \oplus 221 \oplus 221 \oplus 221 \oplus 221 = 193$ ；

第五次操作，将以 8 为根的子树中所有节点的魔力值按位或 287，所有节点的魔力值序列变为 0, 28, 959, 221, 959, 959, 959, 479, 221, 221, 221；

第六次操作，查询以 4 为根的子树中所有节点的魔力值的异或和，即 $221 \oplus 479 \oplus 221 \oplus 221 = 479$ 。