E. Bear and Bad Powers of 42

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Limak, a bear, isn't good at handling queries. So, he asks you to do it.

We say that powers of 42 (numbers $1, 42, 1764, ...$) are *bad*. Other numbers are *good*.

You are given a sequence of n good integers $t_1, t_2, ..., t_n$. Your task is to handle q queries of three types:

1. `1 i` — print t_i in a separate line.
2. `2 a b x` — for $i \in [a, b]$ set t_i to x. It's guaranteed that x is a good number.
3. `3 a b x` — for $i \in [a, b]$ increase t_i by x. After this repeat the process while at least one t_i is bad.

You can note that after each query all t_i are good.

Input

The first line of the input contains two integers n and q $(1 \le n, q \le 100\,000)$ — the size of Limak's sequence and the number of queries, respectively.

The second line of the input contains n integers $t_1, t_2, ..., t_n$ $(2 \le t_i \le 10^9)$ — initial elements of Limak's sequence. All t_i are good.

Then, q lines follow. The i-th of them describes the i-th query. The first number in the line is an integer $type_i$ $(1 \le type_i \le 3)$ — the type of the query. There is at least one query of the first type, so the output won't be empty.

In queries of the second and the third type there is $1 \le a \le b \le n$.

In queries of the second type an integer x $(2 \le x \le 10^9)$ is guaranteed to be good.

In queries of the third type an integer x $(1 \le x \le 10^9)$ may be bad.

Output

For each query of the first type, print the answer in a separate line.

Example

input

```
6 12
40 1700 7 1672 4 1722
3 2 4 42
1 2
1 3
3 2 6 50
1 2
1 4
1 6
2 3 4 41
3 1 5 1
1 1
1 3
1 5
```

output

```
1742
49
1842
1814
1822
43
44
```

Codeforces Round #356 (Div. 1)

Finished

Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Submit?

Language: GNU G++ 5.1.0

Choose file: 选择文件 未选择任何文件

Be careful: there is 50 points penalty for submission which fails the pretests or resubmission (except failure on the first test, denial of judgement or similar verdicts). "Passed pretests" submission verdict doesn't guarantee that the solution is absolutely correct and it will pass system tests.

Submit

→ Problem tags

data structures

No tag edit access

→ Contest materials

- Announcement
- Tutorial

```
107
```

Note

After a query `3 2 4 42` the sequence is $40, 1742, 49, 1714, 4, 1722$.

After a query `3 2 6 50` the sequence is $40, 1842, 149, 1814, 104, 1822$.

After a query `2 3 4 41` the sequence is $40, 1842, 41, 41, 104, 1822$.

After a query `3 1 5 1` the sequence is $43, 1845, 44, 44, 107, 1822$.