

D. Christmas Trees

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

There are n Christmas trees on an infinite number line. The i -th tree grows at the position x_i . All x_i are guaranteed to be distinct.

Each **integer** point can be either occupied by the Christmas tree, by the human or not occupied at all. Non-integer points cannot be occupied by anything.

There are m people who want to celebrate Christmas. Let y_1, y_2, \dots, y_m be the positions of people (note that all values $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m$ should be **distinct** and all y_j should be **integer**). You want to find such an arrangement of people that the value

$$\sum_{j=1}^m \min_{i=1}^n |x_i - y_j|$$

is the minimum possible (in other words, the sum of distances to the

nearest Christmas tree for all people should be minimized).

In other words, let d_j be the distance from the j -th human to the nearest Christmas tree (

$$d_j = \min_{i=1}^n |y_j - x_i|$$

Then you need to choose such positions y_1, y_2, \dots, y_m that $\sum_{j=1}^m d_j$ is

the minimum possible.

Input

The first line of the input contains two integers n and m ($1 \leq n, m \leq 2 \cdot 10^5$) — the number of Christmas trees and the number of people.

The second line of the input contains n integers x_1, x_2, \dots, x_n ($-10^9 \leq x_i \leq 10^9$), where x_i is the position of the i -th Christmas tree. It is guaranteed that all x_i are distinct.

Output

In the first line print one integer res — the minimum possible value of $\sum_{j=1}^m \min_{i=1}^n |x_i - y_j|$ (in other words, the sum of distances to the nearest Christmas tree for all people).

In the second line print m integers y_1, y_2, \dots, y_m ($-2 \cdot 10^9 \leq y_j \leq 2 \cdot 10^9$), where y_j is the position of the j -th human. All y_j should be distinct and all values $x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m$ should be **distinct**.

If there are multiple answers, print any of them.

Examples

input	Copy
2 6 1 5	
output	Copy

Codeforces Round #611 (Div. 3)

Finished

→ 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 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](#)

→ Problem tags

[graphs](#) [greedy](#) [shortest paths](#) [*1800](#)

No tag edit access

→ Contest materials

- Announcement (en) [×](#)
- Tutorial [×](#)

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

input[Copy](#)

```
3 5
0 3 1
```

output[Copy](#)

```
7
5 -2 4 -1 2
```

[Codeforces](#) (c) Copyright 2010-2020 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Jul/21/2020 22:47:45^{UTC-4} (i1).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#)

Supported by



ITMO UNIVERSITY