P13

As you may know from the comic "Asterix and the Chieftain's Shield", Gergovia consists of one street, and every inhabitant of the city is a wine salesman. You wonder how this economy works? Simple enough: everyone buys wine from other inhabitants of the city. Every day each inhabitant decides how much wine he wants to buy or sell. Interestingly, demand and supply is always the same, so that each inhabitant gets what he wants.

There is one problem, however: Transporting wine from one house to another results in work. Since all wines are equally good, the inhabitants of Gergovia don't care which persons they are doing trade with, they are only interested in selling or buying a specific amount of wine. They are clever enough to figure out a way of trading so that the overall amount of work needed for transports is minimized.

In this problem you are asked to reconstruct the trading during one day in Gergovia. For simplicity we will assume that the houses are built along a straight line with equal distance between adjacent houses. Transporting one bottle of wine from one house to an adjacent house results in one unit of work.

Input

The input consists of several test cases. Each test case starts with the number of inhabitants n ($2 \leq n \leq 100000$). The following line contains n integers a_i ($-1000 \leq a_i \leq 1000$). If $a_i \geq 0$, it means that the inhabitant living in the i-th house wants to buy a_i bottles of wine, otherwise if $a_i < 0$, he wants to sell $-a_i$ bottles of wine. You may assume that the numbers a_i sum up to 0.

The last test case is followed by a line containing '0'.

Output

For each test case print the minimum amount of work units needed so that every inhabitant has his demand fulfilled. You may assume that this number fits into a signed 64-bit integer (in C/C++ you can use the data type "long long", in JAVA the data type "long").

Sample Input

```
5
5 -4 1 -3 1
6
-1000 -1000 -1000 1000 1000 1000
0
```

Sample Output

```
9
9000
```

P14

Flatland government is building a new highway that will be used to transport weapons from its main weapon plant to the frontline in order to support the undergoing military operation against its neighbor country Edgeland. Highway is a straight line and there are n construction teams working at some points on it.

During last days the threat of a nuclear attack from Edgeland has significantly increased. Therefore the construction office has decided to develop an evacuation plan for the construction teams in case of a nuclear attack. There are m shelters located near the constructed highway. This evacuation plan must assign each team to a shelter that it should use in case of an attack.

Each shelter entrance must be securely locked from the inside to prevent any damage to the shelter itself. So, for each shelter there must be some team that goes to this shelter in case of an attack. The office must also supply fuel to each team, so that it can drive to its assigned shelter in case of an attack. The amount of fuel that is needed is proportional to the distance from the team's location to the assigned shelter. To minimize evacuation costs, the office would like to create a plan that minimizes the total fuel needed.

Your task is to help them develop such a plan.

Input

The input file contains several test cases, each of them as described below.

The first line of the input file contains n — the number of construction teams ($1 \leq n \leq 4000$). The second line contains n integer numbers - the locations of the teams. Each team's location is a positive integer not exceeding 10^9, all team locations are different.

The third line of the input file contains m — the number of shelters ($1 \leq m \leq n$).

The fourth line contains m integer numbers — the locations of the shelters. Each shelter's location is a positive integer not exceeding 10^9, all shelter locations are different.

The amount of fuel that needs to be supplied to a team at location x that goes to a shelter at location y is equal to $|x - y|$.

Output

For each test case, the output must follow the description below.

The first line of the output file must contain z — the total amount of fuel needed. The second line must contain n integer numbers: for each team output the number of the shelter that it should be assigned to. Shelters are numbered from 1 to m in the order they are listed in the input file.

Sample Input

```
3
1 2 3
2
2 10
```

Sample Output

```
8
1 1 2
```