Task 5: Explosives

You are an explosive handler in charge of transporting explosives. There are n factories and n mines located on a straight line. The factories produce explosives and the mines require explosives.

Each factory produces 1 unit of explosives, and each mine requires 1 unit of explosives. The i-th factory ($1 \leq i \leq n$) is located at position $a[i]$ and the j-th mine ($1 \leq j \leq n$) is located at position $b[j]$. The factories and mines all have different locations (i.e. $a[1 \dots n], b[1 \dots n]$ are all distinct).

As the explosive handler, you drive a truck which can carry at most c units of explosives. You start at location 0 and your truck contains no explosives initially. Your aim is to transport all the explosives from factories to mines.

More specifically, you can execute the following operations:

1. `pickup(x)`: Drive from your current location to the factory located at x, and pickup 1 unit of explosive from this factory. You are only allowed to do this if
   - $x = a[i]$ (for some $1 \leq i \leq n$)
   - Your truck contains at most $c - 1$ units of explosives.

   This operation increases the amount of explosives on your truck by 1.
2. `offload(x)`: Drive from your current location to the mine located at x, and offload 1 unit of explosive to this mine. You are only allowed to do this if
   - $x = b[j]$ (for some $1 \leq j \leq n$)
   - Your truck contains at least 1 unit of explosives.

   This operation decreases the amount of explosives on your truck by 1.

You should pickup exactly 1 unit of explosive at every factory, and offload exactly 1 unit of explosive at every mine.

Whenever there are explosives on the truck, you need to pay a safety officer to watch the explosives for you.

In particular, if you drive from x to y carrying at least 1 unit of explosives, then you need to pay the officer a cost of $|x - y|$, which does not depend on the amount of explosives on your truck. Note that you do not need to pay this cost when there are no explosives on your truck.

Construct a sequence of operations that minimizes this cost.

Input format

Your program must read from standard input.

The first line of input contains two integers, n and c.

The second line will contain n integers $a[1 \ldots n]$.

The third line will contain n integers $b[1 \ldots n]$.

Output format

Your program must print to standard output.

The first line should contain the minimum cost required. The second line should contain $2n$ integers, indicating the positions of all the factories and mines you visit, in order.

For example, if your operations are `pickup(3), offload(5), pickup(6), offload(2)` in that order, the second line should contain `3 5 6 2`.

Subtasks

For all testcases, the input will satisfy the following bounds:

- $1 \leq n \leq 1000$
- $1 \leq c \leq 1000$
- $1 \leq a[i], b[i] \leq 10000$ (for all $1 \leq i \leq n$)
- $a[1 \ldots n], b[1 \ldots n]$ are all distinct (i.e. all factories and mines have distinct locations).

Your program will be tested on input instances that satisfy the following restrictions:

Subtask	Marks	Additional Constraints
0	0	Sample Testcases
1	16	$c = 1$
2	22	$a[i] \leq 5000, b[i] > 5000$ (for all $1 \leq i \leq n$)
3	62	No additional restrictions

For each subtask, you will get 50% of points if you output the correct minimum cost for all test cases in the subtask

Sample Testcase 1

Input	Output

`3 2`
`12 14 4`
`9 5 8`
	`7`
	`4 5 14 12 9 8`

Sample Testcase 1 Explanation

There are 3 factories, located at positions 12, 14 and 4. There are 3 mines, located at positions 9, 5 and 8.

A possible sequence of operations is `pickup(4), offload(5), pickup(14), pickup(12), offload(9), offload(8)`. The cost of these operations can be calculated as follows:

- Initially, there are no explosives on the truck. You do not need to pay to move from 0 to 4.
- Once we pickup at position 4, there is now 1 unit of explosive on your truck. The operation `offload(5)` costs $|5 - 4| = 1$. The truck is now empty (i.e. contains no explosives).
- `pickup(14)` does not cost you anything as the truck is now empty.
- `pickup(12)` costs $|14 - 12| = 2$ as the truck contains 1 unit of explosives.
- `offload(9)` costs $|12 - 9| = 3$
- `offload(8)` costs $|9 - 8| = 1$
- The cost you pay is therefore $1 + 2 + 3 + 1 = 7$.

If your output had just been:

```
7
```

Then you would get 50% of the points. Note that you are **not allowed** to print anything after the cost and it has to be correct for all test cases in the subtask.