C. Looking for Order

time limit per test: 4 seconds
memory limit per test: 512 megabytes

Girl Lena likes it when everything is in order, and looks for order everywhere. Once she was getting ready for the University and noticed that the room was in a mess — all the objects from her handbag were thrown about the room. Of course, she wanted to put them back into her handbag. The problem is that the girl cannot carry more than two objects at a time, and cannot move the handbag. Also, if he has taken an object, she cannot put it anywhere except her handbag — her inherent sense of order does not let her do so.

You are given the coordinates of the handbag and the coordinates of the objects in some Cartesian coordinate system. It is known that the girl covers the distance between any two objects in the time equal to the squared length of the segment between the points of the objects. It is also known that initially the coordinates of the girl and the handbag are the same. You are asked to find such an order of actions, that the girl can put all the objects back into her handbag in a minimum time period.

Input

The first line of the input file contains the handbag's coordinates x_s, y_s. The second line contains number n ($1 \le n \le 24$) — the amount of objects the girl has. The following n lines contain the objects' coordinates. All the coordinates do not exceed 100 in absolute value. All the given positions are different. All the numbers are integer.

Output

In the first line output the only number — the minimum time the girl needs to put the objects into her handbag.

In the second line output the possible optimum way for Lena. Each object in the input is described by its index number (from 1 to n), the handbag's point is described by number 0. The path should start and end in the handbag's point. If there are several optimal paths, print any of them.

Examples

input Copy

```
0 0
2
1 1
-1 1
```

output Copy

```
8
0 1 2 0
```

input Copy

```
1 1
3
4 3
3 4
0 0
```

output Copy

```
32
0 1 2 0 3 0
```

Codeforces Beta Round 8
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 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

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++23 14.2 (64 bit, ms

Choose file: Choose File No file chosen

Submit

→ Last submissions

Submission	Time	Verdict
325039418	Jun/18/2025 17:14	Accepted

→ Problem tags

bitmasks dp *2000

No tag edit access

→ Contest materials

- Codeforces Beta Round #8
- Tutorial (en)

Codeforces (c) Copyright 2010-2025 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Jun/18/2025 21:14:42^{UTC+7} (k1).
Desktop version, switch to mobile version.
Privacy Policy | Terms and Conditions

Supported by