Task	BIJELE	CRNE	PRVA	TURBO	KEMIJA	PRAVOKUTNI
Input	standard input (keyboard)					
Output	standard output (screen)					
Memory limit (heap+stack)	32 MB					
Time limit (per test)	1 second					4 seconds
Number of tests	5	5	8	10	10	10
Points per test	4	4	5	7	7	8
Total points	**20**	**20**	**40**	**70**	**70**	**80**
	300					

Note: The evaluation system has two Intel Pentium 4 3.0 GHz processors and is running the Linux operating system. The following compile options are used for different languages:

- C: –O2 –s –static –std=c99 –lm
- C++: –O2 –s –static –lm
- Pascal: –O1 –XS

1. BIJELE

Mirko has found an old chessboard and a set of pieces in his attic. Unfortunately, the set contains only white pieces, and apparently an incorrect number of them. A set of pieces **should** contain:

- One king
- One queen
- Two rooks
- Two bishops
- Two knights
- Eight pawns

Mirko would like to know how many pieces of each type he should **add or remove** to make a valid set.

Input

The input consists of 6 integers on a single line, each between 0 and 10 (inclusive). The numbers are, in order, the numbers of kings, queens, rooks, bishops, knights and pawns in the set Mirko found.

Output

Output should consist of 6 integers on a single line; the number of pieces of each type Mirko should add or remove. If a number is positive, Mirko needs to add that many pieces. If a number is negative, Mirko needs to remove pieces.

Sample test data

input	**input**
`0 1 2 2 2 7`	`2 1 2 1 2 1`
output	**output**
`1 0 0 0 0 1`	`-1 0 0 1 0 7`

Thrilled about his new valid set of pieces, Mirko rushed over to Slavko's, to find that Slavko too found a set of chess pieces in his attic. Slavko's set, miraculously, contains only black pieces. But since neither of them can play chess, they settled on smashing one another senseless with their chessboards.

While Slavko is warming up with a series of stretches, Mirko decided to sabotage Slavko's chessboard. An expert in carving wood, he decided to cut Slavko's chessboard so that it shatters into **as many pieces as possible** when Slavko attempts to hit Mirko.

Mirko can only make **horizontal and vertical cuts** (parallel to the sides to the board), edge to edge, and has time to make **at most N cuts**.

Input

The first line of input contains an integer N ($1 \leq N \leq 100$), the number of cuts Mirko can make.

Output

Output the largest number of pieces Slavko's chessboard can crash into.

Sample test data

input	input
1	3
output	output
2	6

Little Ivica solves crossword puzzles every day. In case you haven't seen one, a crossword puzzle starts on a grid of R×C squares, each of which is either empty or blocked. The player's task is to write words in consecutive empty squares vertically (top down) or horizontally (left to right).

Ivica's sister has a strange habit of looking at crosswords Ivica has finished solving, and finding the **lexicographically smallest word** in it. She only considers words at least 2 characters long.

Write a program that, given a crossword puzzle, finds that word.

Input

The first line contains two integers R and C ($2 \leq R, C \leq 20$), the number of rows and columns in the crossword.

Each of the following R lines contains a string of C characters. Each of those characters is either a lowercase letter of the English alphabet, or the character '#' representing a blocked square.

The input will be such that a solution will always exist.

Output

Output the lexicographically smallest word in the crossword.

Sample test data

input

```
4 4
luka
o#a#
kula
i#a#
```

output

```
kala
```

input

```
4 4
luka
o#a#
kula
i#as
```

output

```
as
```

input

```
4 5
adaca
da##b
abb#b
abbac
```

output

```
abb
```

Frane has been given the task of sorting an array of numbers. The array consists of N integers, each between 1 and N (inclusive), with each of those appearing exactly once in the array. Frane has come up with the following sorting algorithm which operates in N phases, and named it turbosort:

- In the first phase, the number 1 is moved to position 1 by repeatedly swapping consecutive elements.
- In the second phase, the number N is moved to position N in the same manner.
- In the third phase, the number 2 is moved to position 2.
- In the fourth phase, the number N−1 is moved to position N−1.
- And so on.

In other words, when the number of the phase is odd, Frane will choose the smallest number not yet chosen, and move it to its final position. In even phases he chooses the largest number not yet chosen.

Write a program which, given the initial array, output the **number of swaps in each phase** of the algorithm.

Input

The first line contains an integer N ($1 \leq N \leq 100000$), the number of elements in the array.

Each of the following N lines contains an integer between 1 and N (inclusive), the array to be sorted. The array will contain no duplicates.

Output

For each of the N phases, output the number of swaps on a single line.

Scoring

In test cases worth 70% of points, N will be less than 100.

Sample test data

input

```
3
2
1
3
```

output

```
1
0
0
```

input

```
5
5
4
3
2
1
```

output

```
4
3
2
1
0
```

input

```
7
5
4
3
7
1
2
6
```

output

```
4
2
3
0
2
1
0
```

Instead of paying attention in chemistry class, Luka is killing time playing with numbers again. This time, he wrote N **positive integers** so that they form a ring (circle). After that he formed a new ring by **adding to each number its two neighbours**.

The teacher noticed this and took away the first piece of paper, with the original ring. This did not trouble Luka much because he knows he can use the other ring to reconstruct the original.

Write a program that solves Luka's problem.

Input

The first line contains the integer N ($3 \leq N \leq 10000$), the number of integers in the rings.

Each of the following N lines contains an integer less than 10^9 (one billion). These numbers, in order, form the **second** ring.

The input will be such that a solution, although not necessarily unique, will always exist.

Output

Output the original ring on N lines. The numbers must be positive.

Rotating the ring is not allowed. For example, the sum of the first three output numbers must be equal to the second number in the input ring.

Note: The solution need not be unique.

Scoring

In test cases worth 70% of points, N will be less than 100.

Sample test data

input

```
3
5
5
5
```

output

```
2
1
2
```

input

```
4
20
15
17
14
```

output

```
5
8
2
7
```

input

```
5
7
8
9
10
11
```

output

```
4
1
3
5
2
```

N points are placed in the coordinate plane.

Write a program which calculates in how many ways a **right triangle** can be formed by three of the given points. A right triangle is one in which one of the angles is 90 degrees.

Input

The first line of input contains an integer N ($3 \leq N \leq 1500$), the number of points.

Each of the following N lines contains the coordinates of one point, two integers separated by a space. The coordinates will be between -10^9 and 10^9.

No two points will be located at the same coordinates.

Output

Output the number of right triangles.

Sample test data

input

```
3
4 2
2 1
1 3
```

output

```
1
```

input

```
4
5 0
2 6
8 6
5 7
```

output

```
0
```

input

```
5
-1 1
-1 0
0 0
1 0
1 1
```

output

```
7
```