A. Mister B and Boring Game

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Sometimes Mister B has free evenings when he doesn't know what to do. Fortunately, Mister B found a new game, where the player can play against aliens.

All characters in this game are lowercase English letters. There are two players: Mister B and his competitor.

Initially the players have a string s consisting of the first a English letters in alphabetical order (for example, if $a = 5$, then s equals to "abcde").

The players take turns appending letters to string s. Mister B moves first.

Mister B must append exactly b letters on each his move. He can arbitrary choose these letters. His opponent adds exactly a letters on each move.

Mister B quickly understood that his opponent was just a computer that used a simple algorithm. The computer on each turn considers the suffix of string s of length a and generates a string t of length a such that all letters in the string t are distinct and don't appear in the considered suffix. From multiple variants of t lexicographically minimal is chosen (if $a = 4$ and the suffix is "bfdd", the computer chooses string t equal to "aceg"). After that the chosen string t is appended to the end of s.

Mister B soon found the game boring and came up with the following question: what can be the minimum possible number of different letters in string s on the segment between positions l and r, inclusive. Letters of string s are numerated starting from 1.

Input

First and only line contains four space-separated integers: a, b, l and r ($1 \le a, b \le 12$, $1 \le l \le r \le 10^9$) — the numbers of letters each player appends and the bounds of the segment.

Output

Print one integer — the minimum possible number of different letters in the segment from position l to position r, inclusive, in string s.

Examples

input
```
1 1 1 8
```
output
```
2
```

input
```
4 2 2 6
```
output
```
3
```

input
```
3 7 4 6
```
output
```
1
```

Note

In the first sample test one of optimal strategies generate string s = "abababab...", that's why answer is 2.

In the second sample test string s = "abcdbcaefg..." can be obtained, chosen segment will look like "bcdbc", that's why answer is 3.

In the third sample test string s = "abczzzacad..." can be obtained, chosen, segment will look like "zzz", that's why answer is 1.

B. Mister B and PR Shifts

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Some time ago Mister B detected a strange signal from the space, which he started to study.

After some transformation the signal turned out to be a permutation p of length n or its cyclic shift. For the further investigation Mister B need some basis, that's why he decided to choose cyclic shift of this permutation which has the minimum possible deviation.

Let's define the deviation of a permutation p as $\sum_{i=1}^{i=n} |p[i] - i|$.

Find a cyclic shift of permutation p with minimum possible deviation. If there are multiple solutions, print any of them.

Let's denote id k $(0 \le k < n)$ of a cyclic shift of permutation p as the number of right shifts needed to reach this shift, for example:

- $k = 0$: shift $p_1, p_2, \dots p_n$,
- $k = 1$: shift $p_n, p_1, \dots p_{n-1}$,
- ...,
- $k = n - 1$: shift $p_2, p_3, \dots p_n, p_1$.

Input

First line contains single integer n $(2 \le n \le 10^6)$ — the length of the permutation.

The second line contains n space-separated integers $p_1, p_2, \dots, p_n$ $(1 \le p_i \le n)$ — the elements of the permutation. It is guaranteed that all elements are distinct.

Output

Print two integers: the minimum deviation of cyclic shifts of permutation p and the id of such shift. If there are multiple solutions, print any of them.

Examples

input

```
3
1 2 3
```

output

```
0 0
```

input

```
3
2 3 1
```

output

```
0 1
```

input

```
3
3 2 1
```

output

```
2 1
```

Note

In the first sample test the given permutation p is the identity permutation, that's why its deviation equals to 0, the shift id equals to 0 as well.

In the second sample test the deviation of p equals to 4, the deviation of the 1-st cyclic shift $(1, 2, 3)$ equals to 0, the deviation of the 2-nd cyclic shift $(3, 1, 2)$ equals to 4, the optimal is the 1-st cyclic shift.

In the third sample test the deviation of p equals to 4, the deviation of the 1-st cyclic shift $(1, 3, 2)$ equals to 2, the deviation of the 2-nd cyclic shift $(2, 1, 3)$ also equals to 2, so the optimal are both 1-st and 2-nd cyclic shifts.

C. Mister B and Beacons on Field

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Mister B has a house in the middle of a giant plain field, which attracted aliens life. For convenience, aliens specified the Cartesian coordinate system on the field in such a way that Mister B's house has coordinates $(0, 0)$. After that they sent three beacons to the field, but something went wrong. One beacon was completely destroyed, while the other two landed in positions with coordinates $(m, 0)$ and $(0, n)$, respectively, but shut down.

Mister B was interested in this devices, so he decided to take them home. He came to the first beacon, placed at $(m, 0)$, lifted it up and carried the beacon home choosing the shortest path. After that he came to the other beacon, placed at $(0, n)$, and also carried it home choosing the shortest path. When first beacon was lifted up, the navigation system of the beacons was activated.

Partially destroyed navigation system started to work in following way.

At time moments when both survived beacons are at points with integer coordinates the system tries to find a location for the third beacon. It succeeds if and only if there is a point with integer coordinates such that the area of the triangle formed by the two survived beacons and this point is equal to s. In this case the system sends a packet of information with beacon positions to aliens, otherwise it doesn't.

Compute how many packets of information system sent while Mister B was moving the beacons.

Input

The first line contains one integer t $(1 \le t \le 1000)$ — the number of test cases. The next $3 \cdot t$ lines describe t test cases.

Every test case is described in three lines as follows. **Note that each parameter is given as a product of three factors.**

The first line of a test case contains three space-separated integers: n_1, n_2, n_3 $(1 \le n_i \le 10^6)$ such that $n = n_1 \cdot n_2 \cdot n_3$.

The second line contains three space-separated integers: m_1, m_2, m_3 $(1 \le m_i \le 10^6)$ such that $m = m_1 \cdot m_2 \cdot m_3$.

The third line contains three space-separated integers: s_1, s_2, s_3 $(1 \le s_i \le 10^6)$ such that $s = s_1 \cdot s_2 \cdot s_3$.

Note that for hacks only tests with $t = 1$ allowed.

Output

Print t integers one per line — the answers for each test.

Example

input

```
3
2 1 1
2 1 1
1 1 3
1 5 1
2 2 1
1 1 2
10 6 18
2 103 2
13 1 13
```

output

```
4
7
171
```

Note

First test case contains the following beacon positions: $(2, 0)$ and $(0, 2)$, $s = 3$. The following packets could be sent: $((2, 0), (0, 2), (-1, 0))$, $((1, 0), (0, 2), (4, 0))$, $((0, 0), (0, 2), (3, 1))$, $((0, 0), (0, 1), (-6, 0))$, where (b_1, b_2, p) has next description: b_1 — first beacon position, b_2 — second beacon position, p — some generated point.

Second test case contains the following beacon initial positions: $(4, 0)$ and $(0, 5)$, $s = 2$. The following packets could be sent: $((4, 0), (0, 5), (0, 4))$, $((3, 0), (0, 5), (2, 3))$, $((2, 0), (0, 5), (2, 2))$, $((1, 0), (0, 5), (1, 4))$, $((0, 0), (0, 4), (0, -1))$, $((0, 0), (0, 2), (2, 0))$, $((0, 0), (0, 1), (4, 0))$.

D. Mister B and Astronomers

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

After studying the beacons Mister B decided to visit alien's planet, because he learned that they live in a system of flickering star Moon. Moreover, Mister B learned that the star shines once in exactly T seconds. The problem is that the star is yet to be discovered by scientists.

There are n astronomers numerated from 1 to n trying to detect the star. They try to detect the star by sending requests to record the sky for 1 second.

The astronomers send requests **in cycle**: the i-th astronomer sends a request exactly a_i second after the $(i - 1)$-th (i.e. if the previous request was sent at moment t, then the next request is sent at moment $t + a_i$); the 1-st astronomer sends requests a_1 seconds later than the n-th. The first astronomer sends his first request at moment 0.

Mister B doesn't know the first moment the star is going to shine, but it's obvious that all moments at which the star will shine are determined by the time of its shine moment in the interval $[0, T)$. Moreover, this interval can be split into T parts of 1 second length each of form $[t, t + 1)$, where $t = 0, 1, 2, ..., (T - 1)$.

Mister B wants to know how lucky each astronomer can be in discovering the star first.

For each astronomer compute how many segments of form $[t, t + 1)$ ($t = 0, 1, 2, ..., (T - 1)$) there are in the interval $[0, T)$ so that this astronomer is the first to discover the star if the first shine of the star happens in this time interval.

Input

The first line contains two integers T and n ($1 \le T \le 10^9$, $2 \le n \le 2 \cdot 10^5$).

The second line contains n integers $a_1, a_2, ..., a_n$ ($1 \le a_i \le 10^9$).

Output

Print n integers: for each astronomer print the number of time segments describer earlier.

Examples

input

```
4 2
2 3
```

output

```
3 1
```

input

```
5 4
1 1 1 1
```

output

```
2 1 1 1
```

Note

In the first sample test the first astronomer will send requests at moments $t_1 = 0, 5, 10, ...$, the second — at moments $t_2 = 3, 8, 13, ...$. That's why interval $[0, 1)$ the first astronomer will discover first at moment $t_1 = 0$, $[1, 2)$ — the first astronomer at moment $t_1 = 5$, $[2, 3)$ — the first astronomer at moment $t_1 = 10$, and $[3, 4)$ — the second astronomer at moment $t_2 = 3$.

In the second sample test interval $[0, 1)$ — the first astronomer will discover first, $[1, 2)$ — the second astronomer, $[2, 3)$ — the third astronomer, $[3, 4)$ — the fourth astronomer, $[4, 5)$ — the first astronomer.

E. Mister B and Flight to the Moon

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In order to fly to the Moon Mister B just needs to solve the following problem.

There is a complete indirected graph with n vertices. You need to cover it with several simple cycles of length 3 and 4 so that each edge is in exactly 2 cycles.

We are sure that Mister B will solve the problem soon and will fly to the Moon. Will you?

Input

The only line contains single integer n ($3 \le n \le 300$).

Output

If there is no answer, print `-1`.

Otherwise, in the first line print k ($1 \le k \le n^2$) — the number of cycles in your solution.

In each of the next k lines print description of one cycle in the following format: first print integer m ($3 \le m \le 4$) — the length of the cycle, then print m integers $v_1, v_2, \ldots, v_m$ ($1 \le v_i \le n$) — the vertices in the cycle in the traverse order. Each edge should be in exactly two cycles.

Examples

input

```
3
```

output

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

input

```
5
```

output

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