A. Five Dimensional Points

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

You are given set of n points in 5-dimensional space. The points are labeled from 1 to n. No two points coincide.

We will call point a *bad* if there are different points b and c, not equal to a, from the given set such that angle between vectors $\vec{ab}$ and $\vec{ac}$ is acute (i.e. strictly less than 90°). Otherwise, the point is called *good*.

The angle between vectors $\vec{x}$ and $\vec{y}$ in 5-dimensional space is defined as $\arccos(\frac{\vec{x} \cdot \vec{y}}{|\vec{x}||\vec{y}|})$, where $\vec{x} \cdot \vec{y} = x_1 y_1 + x_2 y_2 + x_3 y_3 + x_4 y_4 + x_5 y_5$ is the scalar product and $|\vec{x}| = \sqrt{\vec{x} \cdot \vec{x}}$ is length of $\vec{x}$.

Given the list of points, print the indices of the good points in ascending order.

Input

The first line of input contains a single integer n ($1 \le n \le 10^3$) — the number of points.

The next n lines of input contain five integers a_i, b_i, c_i, d_i, e_i ($|a_i|, |b_i|, |c_i|, |d_i|, |e_i| \le 10^3$) — the coordinates of the i-th point. All points are distinct.

Output

First, print a single integer k — the number of good points.

Then, print k integers, each on their own line — the indices of the good points in ascending order.

Examples

input

```
6
0 0 0 0 0
1 0 0 0 0
0 1 0 0 0
0 0 1 0 0
0 0 0 1 0
0 0 0 0 1
```

output

```
1
1
```

input

```
3
0 0 1 2 0
0 0 9 2 0
0 0 5 9 0
```

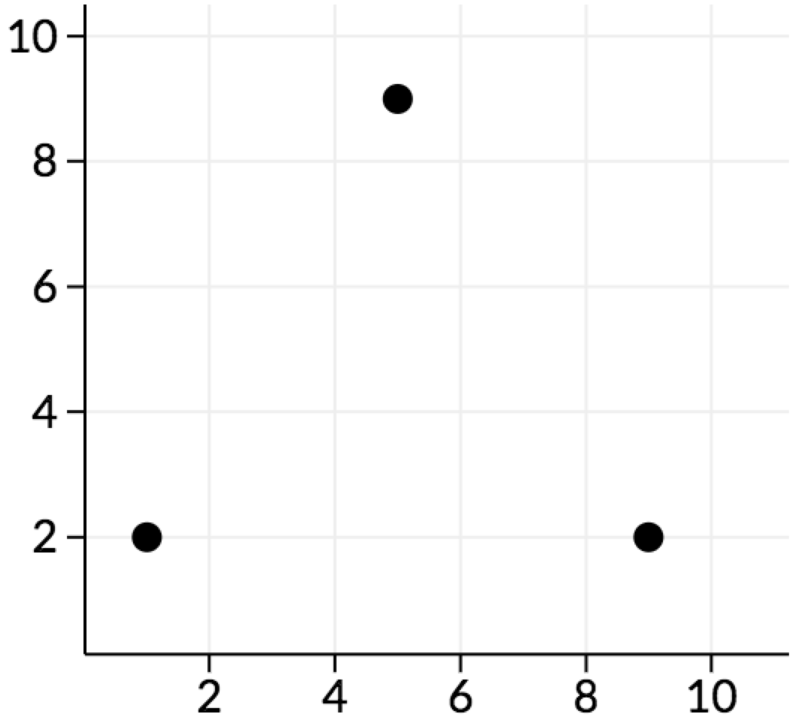
output

```
0
```

Note

In the first sample, the first point forms exactly a 90° angle with all other pairs of points, so it is good.

In the second sample, along the cd plane, we can see the points look as follows:



We can see that all angles here are acute, so no points are good.

B. Arpa and a list of numbers

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

Arpa has found a list containing n numbers. He calls a list bad if and only if it is not empty and gcd (see notes section for more information) of numbers in the list is 1.

Arpa can perform two types of operations:

- Choose a number and delete it with cost x.
- Choose a number and increase it by 1 with cost y.

Arpa can apply these operations to as many numbers as he wishes, and he is allowed to apply the second operation arbitrarily many times on the same number.

Help Arpa to find the minimum possible cost to make the list good.

Input

First line contains three integers n, x and y ($1 \le n \le 5 \cdot 10^5$, $1 \le x, y \le 10^9$) — the number of elements in the list and the integers x and y.

Second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^6$) — the elements of the list.

Output

Print a single integer: the minimum possible cost to make the list good.

Examples

input

```
4 23 17
1 17 17 16
```

output

```
40
```

input

```
10 6 2
100 49 71 73 66 96 8 60 41 63
```

output

```
10
```

Note

In example, number 1 must be deleted (with cost 23) and number 16 must increased by 1 (with cost 17).

A gcd (greatest common divisor) of a set of numbers is the maximum integer that divides all integers in the set. Read more about gcd here.

C. Arpa and a game with Mojtaba

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Mojtaba and Arpa are playing a game. They have a list of n numbers in the game.

In a player's turn, he chooses a number p^k (where p is a prime number and k is a positive integer) such that p^k divides at least one number in the list. For each number in the list divisible by p^k, call it x, the player will delete x and add $\frac{x}{p^k}$ to the list. The player who can not make a valid choice of p and k loses.

Mojtaba starts the game and the players alternatively make moves. Determine which one of players will be the winner if both players play optimally.

Input

The first line contains a single integer n $(1 \le n \le 100)$ — the number of elements in the list.

The second line contains n integers $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le 10^9)$ — the elements of the list.

Output

If Mojtaba wins, print "`Mojtaba`", otherwise print "`Arpa`" (without quotes).

You can print each letter in any case (upper or lower).

Examples

input
```
4
1 1 1 1
```
output
```
Arpa
```

input
```
4
1 1 17 17
```
output
```
Mojtaba
```

input
```
4
1 1 17 289
```
output
```
Arpa
```

input
```
5
1 2 3 4 5
```
output
```
Arpa
```

Note

In the first sample test, Mojtaba can't move.

In the second sample test, Mojtaba chooses $p = 17$ and $k = 1$, then the list changes to $[1, 1, 1, 1]$.

In the third sample test, if Mojtaba chooses $p = 17$ and $k = 1$, then Arpa chooses $p = 17$ and $k = 1$ and wins, if Mojtaba chooses $p = 17$ and $k = 2$, then Arpa chooses $p = 17$ and $k = 1$ and wins.

D. Tournament Construction

time limit per test: 2 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Ivan is reading a book about tournaments. He knows that a tournament is an oriented graph with exactly one oriented edge between each pair of vertices. The score of a vertex is the number of edges going outside this vertex.

Yesterday Ivan learned Landau's criterion: there is tournament with scores $d_1 \le d_2 \le \ldots \le d_n$ if and only if $d_1 + \cdots + d_k \geqslant \frac{k(k-1)}{2}$ for all $1 \le k < n$ and $d_1 + d_2 + \cdots + d_n = \frac{n(n-1)}{2}$.

Now, Ivan wanna solve following problem: given a **set** of numbers $S = \{a_1, a_2, \ldots, a_m\}$, is there a tournament with given set of scores? I.e. is there tournament with sequence of scores $d_1, d_2, \ldots, d_n$ such that if we remove duplicates in scores, we obtain the required set $\{a_1, a_2, \ldots, a_m\}$?

Find a tournament with **minimum** possible number of vertices.

Input

The first line contains a single integer m $(1 \le m \le 31)$.

The next line contains m distinct integers $a_1, a_2, \ldots, a_m$ $(0 \le a_i \le 30)$ — elements of the set S. It is guaranteed that all elements of the set are distinct.

Output

If there are no such tournaments, print string "=(" (without quotes).

Otherwise, print an integer n — the number of vertices in the tournament.

Then print n lines with n characters — matrix of the tournament. The j-th element in the i-th row should be 1 if the edge between the i-th and the j-th vertices is oriented towards the j-th vertex, and 0 otherwise. The main diagonal should contain only zeros.

Examples

input

```
2
1 2
```

output

```
4
0011
1001
0100
0010
```

input

```
2
0 3
```

output

```
6
000111
100011
110001
011001
001101
000000
```

E. Random Elections

time limit per test: 2 seconds
memory limit per test: 1024 megabytes
input: standard input
output: standard output

The presidential election is coming in Bearland next year! Everybody is so excited about this!

So far, there are three candidates, Alice, Bob, and Charlie.

There are n citizens in Bearland. The election result will determine the life of all citizens of Bearland for many years. Because of this great responsibility, each of n citizens will choose one of six orders of preference between Alice, Bob and Charlie uniformly at random, independently from other voters.

The government of Bearland has devised a function to help determine the outcome of the election given the voters preferences. More specifically, the function is $f(x_1, x_2, \ldots, x_n) : \{0, 1\}^n \to \{0, 1\}$ (takes n boolean numbers and returns a boolean number). The function also obeys the following property: $f(1 - x_1, 1 - x_2, \ldots, 1 - x_n) = 1 - f(x_1, x_2, \ldots, x_n)$.

Three rounds will be run between each pair of candidates: Alice and Bob, Bob and Charlie, Charlie and Alice. In each round, x_i will be equal to 1, if i-th citizen prefers the first candidate to second in this round, and 0 otherwise. After this, $y = f(x_1, x_2, \ldots, x_n)$ will be calculated. If $y = 1$, the first candidate will be declared as winner in this round. If $y = 0$, the second will be the winner, respectively.

Define the probability that there is a candidate who won two rounds as p. $p \cdot 6^n$ is always an integer. Print the value of this integer modulo $10^9 + 7 = 1\ 000\ 000\ 007$.

Input

The first line contains one integer n ($1 \le n \le 20$).

The next line contains a string of length 2^n of zeros and ones, representing function f. Let $b_k(x)$ the k-th bit in binary representation of x, i-th (0-based) digit of this string shows the return value of $f(b_1(i), b_2(i), \ldots, b_n(i))$.

It is guaranteed that $f(1 - x_1, 1 - x_2, \ldots, 1 - x_n) = 1 - f(x_1, x_2, \ldots, x_n)$ for any values of $x_1, x_2, ldots, x_n$.

Output

Output one integer — answer to the problem.

Examples

input
```
3
01010101
```
output
```
216
```

input
```
3
01101001
```
output
```
168
```

Note

In first sample, result is always fully determined by the first voter. In other words, $f(x_1, x_2, x_3) = x_1$. Thus, any no matter what happens, there will be a candidate who won two rounds (more specifically, the candidate who is at the top of voter 1's preference list), so $p = 1$, and we print $1 \cdot 6^3 = 216$.

F. Rainbow Balls

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

You have a bag of balls of n different colors. You have a_i balls of the i-th color.

While there are at least two different colored balls in the bag, perform the following steps:

- Take out two random balls without replacement one by one. These balls might be the same color.
- Color the second ball to the color of the first ball. You are not allowed to switch the order of the balls in this step.
- Place both balls back in the bag.
- All these actions take exactly one second.

Let $M = 10^9 + 7$. It can be proven that the expected amount of time needed before you stop can be represented as a rational number $\frac{P}{Q}$, where P and Q are coprime integers and where Q is not divisible by M. Return the value $P \cdot Q^{-1} \mod M$.

Input

The first line of input will contain a single integer n ($1 \le n \le 2\,500$) — the number of colors.

The next line of input will contain n space separated integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^5$) — the number of balls of each color.

Output

Print a single integer, the answer to the problem.

Examples

input

```
2
1 1
```

output

```
1
```

input

```
3
1 2 3
```

output

```
750000026
```

Note

In the first sample, no matter what happens, the balls will become the same color after one step.

For the second sample, we have 6 balls. Let's label the balls from 1 to 6, and without loss of generality, let's say balls 1,2,3 are initially color 1, balls 4,5 are color 2, and ball 6 are color 3.

Here is an example of how these steps can go:

- We choose ball 5 and ball 6. Ball 6 then becomes color 2.
- We choose ball 4 and ball 5. Ball 5 remains the same color (color 2).
- We choose ball 1 and ball 5. Ball 5 becomes color 1.
- We choose ball 6 and ball 5. Ball 5 becomes color 2.
- We choose ball 3 and ball 4. Ball 4 becomes color 1.
- We choose ball 4 and ball 6. Ball 6 becomes color 1.
- We choose ball 2 and ball 5. Ball 5 becomes color 1.

At this point, the game ends since all the balls are the same color. This particular sequence took 7 seconds.
It can be shown that the answer to this case is $\frac{83}{4}$.

Codeforces (c) Copyright 2010-2017 Mike Mirzayanov
The only programming contests Web 2.0 platform