Problem S3: Common Card Choice

Problem Description

There are N cards with positive integers written on them. Alice takes some of the cards, then Bob takes some of the remaining cards. Both need to take at least one card; Alice is not allowed to take all of the cards. Then Alice and Bob compute the sum of numbers on the cards that they have taken. If these sums have a common divisor greater than one, Alice and Bob get a bag of sweets; otherwise they don't. Given the cards on the table, determine if it is possible for Alice and Bob to get the sweets, and if so, how.

To make it even more challenging, in some cases, neither Alice nor Bob knows any of the values on the cards.

Note that an integer d is a *divisor* of an integer q if there is no remainder when q is divided by d. An integer z is a *common divisor* of integers x and y if z is a divisor of both x and y.

Input Specification

The first line of input contains N ($2 \leq N \leq 10^5$).

The second line contains N integers, c_i ($1 \leq c_i \leq 10^{12}$).

The following table shows how the 15 available marks are distributed:

Marks Awarded	**Bounds on N**	**Bounds on c_i**
2 marks	$2 \leq N \leq 3$	$1 \leq c_i \leq 100$
1 marks	$2 \leq N \leq 10$	$1 \leq c_i \leq 100$
1 marks	$2 \leq N \leq 10^5$	$1 \leq c_i \leq 2$
2 marks	$2 \leq N \leq 10^5$	$1 \leq c_i \leq 10^5$
2 marks	$2 \leq N \leq 10^5$	$1 \leq c_i \leq 10^{12}$
7 marks	$N = 10^5$	$c_i = -1$ and see Note below

Note: For the last subtask, you will be given N, but all N card values will be labelled with `-1`, indicating that those card values are unknown to you. In this last subtask, there will always be a selection of cards that Alice and Bob can pick to obtain a bag of sweets.

Output Specification

On the first line of output, print `YES` if it is possible to get the sweets, otherwise, print `NO`.

If the answer is `YES`, on the second line of output, produce A, the number of cards that Alice should take, followed by one space, followed by B, the number of cards that Bob should take. On the third line of output, print out A space-separated numbers corresponding to the indices of the cards that Alice should take. On the fourth line of output, print out B space-separated numbers corresponding to the indices of the cards that Bob should take. If there are several ways to select cards, print any of them.

If the input is for the last subtask, output an integer $K \leq 100$ on the first line, followed by K possible combinations of output as specified for all other subtasks. That is, each combination

La version française figure à la suite de la version anglaise.

will be the same sequence of three lines as stated above: one line containing A and B; one line containing A indices of cards; and one line containing B indices of cards. All guesses of indices of cards must be disjoint: that is, no card index can appear in more than one guess. Your output will be judged correct if any of the combinations would result in Alice and Bob obtaining a bag of sweets. Note that you will not receive any feedback between guesses.

Sample Input 1

```
7
19 7 11 31 99 13 17
```

Output for Sample Input 1

```
YES
3 3
2 3 7
4 6 1
```

Explanation of Output for Sample Input 1

The sum of Alice's cards is $7 + 11 + 17 = 35$ and the sum of Bob's cards is $31 + 13 + 19 = 63$, and both 35 and 63 are divisible by 7.

Sample Input 2

```
3
3 11 17
```

Output for Sample Input 2

```
NO
```

Explanation of Output for Sample Input 2

Notice that 3, 11, and 17 have no common divisors, and all possible sums of two of these numbers (14, 20, 28) do not have a common factor with the remaining third number.

La version française figure à la suite de la version anglaise.

Sample Input 3

```
100000
-1 -1 -1 ...
```

(The input has been truncated for space. There are a total of 10^5 `-1`s on the second line.)

Sample Output 3

```
2
1 2
1
3 5
3 2
100 101 102
201 200
```

Explanation of Output for Sample Input 3

There are a total of two guesses. The first guess consists of indices $\{1\}$ and $\{3, 5\}$. The second guess consists of indices $\{100, 101, 102\}$ and $\{201, 200\}$.

Note that all sets of indices are disjoint.

Although the output is formatted correctly, the output will receive **Wrong answer** because the two guesses are not enough for Alice and Bob to obtain a bag of sweets.

La version française figure à la suite de la version anglaise.

QOJ adaptation notes

Common Card Choice - CCC Senior 2026, Problem S3

The preceding pages are the original official statement, reproduced without changes. The official public test archive does not contain the actual card values used for its hidden-card subtask. This QOJ edition replaces that unavailable hidden instance with 30 locally reconstructed instances. These are replacement judge data, not recovered original contest data.

Input and output

Use standard input and standard output. For the first five subtasks, the original public inputs and the original YES / NO output specification are retained. Any valid pair of nonempty, disjoint selections whose sums have greatest common divisor greater than 1 is accepted. NO is accepted only when no such pair exists.

For every test of the last subtask, the input consists of N = 100000 followed by 100000 values equal to -1. Actual positive card values are held privately by the judge, independently of the submitted output, and are not provided in the input. Each actual value is between 1 and 10^12. Each instance satisfies the existence guarantee in the original statement.

For such an input, print K, where 1 <= K <= 100, followed by K guesses. Each guess consists of A B, then A card indices, then B card indices. Do not print YES or NO in this mode. Both selections must be nonempty. Every index must be in [1,N], and no index may occur twice, either within a guess or across different guesses. All guesses must obey these rules, even if an earlier guess succeeds. The output is accepted if at least one guess has a common divisor greater than 1 for its two hidden sums.

The third original sample illustrates the hidden-mode output format; it is not a correct-answer sample and is not used as a scored or required sample test here. Its stated verdict concerns the unavailable original hidden instance.

Scoring

The six subtask scores remain the official 2, 1, 1, 2, 2, and 7 points, totaling 15. Each subtask is all-or-nothing. The first five subtasks retain all 97 original public inputs. The final 7-point subtask requires passing all 30 reconstructed hidden instances, covering different odd/even distributions, extreme values, repeated values, prime-valued cards, and independently randomized arrays.