A. Choosing Teams

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

The Saratov State University Olympiad Programmers Training Center (SSU OPTC) has n students. For each student you know the number of times he/she has participated in the ACM ICPC world programming championship. According to the ACM ICPC rules, each person can participate in the world championship at most 5 times.

The head of the SSU OPTC is recently gathering teams to participate in the world championship. Each team must consist of exactly three people, at that, any person cannot be a member of two or more teams. What maximum number of teams can the head make if he wants each team to participate in the world championship with the same members at least k times?

Input

The first line contains two integers, n and k $(1 \le n \le 2000; 1 \le k \le 5)$. The next line contains n integers: $y_1, y_2, \dots, y_n$ $(0 \le y_i \le 5)$, where y_i shows the number of times the i-th person participated in the ACM ICPC world championship.

Output

Print a single number — the answer to the problem.

Examples

input

```
5 2
0 4 5 1 0
```

output

```
1
```

input

```
6 4
0 1 2 3 4 5
```

output

```
0
```

input

```
6 5
0 0 0 0 0 0
```

output

```
2
```

Note

In the first sample only one team could be made: the first, the fourth and the fifth participants.

In the second sample no teams could be created.

In the third sample two teams could be created. Any partition into two teams fits.

B. Football Kit

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

Consider a football tournament where n teams participate. Each team has two football kits: for home games, and for away games. The kit for home games of the i-th team has color x_i and the kit for away games of this team has color y_i $(x_i \neq y_i)$.

In the tournament, each team plays exactly one home game and exactly one away game with each other team ($n(n - 1)$ games in total). The team, that plays the home game, traditionally plays in its home kit. The team that plays an away game plays in its away kit. However, if two teams has the kits of the same color, they cannot be distinguished. In this case the away team plays in its home kit.

Calculate how many games in the described tournament each team plays in its home kit and how many games it plays in its away kit.

Input

The first line contains a single integer n $(2 \leq n \leq 10^5)$ — the number of teams. Next n lines contain the description of the teams. The i-th line contains two space-separated numbers x_i, y_i $(1 \leq x_i, y_i \leq 10^5;\ x_i \neq y_i)$ — the color numbers for the home and away kits of the i-th team.

Output

For each team, print on a single line two space-separated integers — the number of games this team is going to play in home and away kits, correspondingly. Print the answers for the teams in the order they appeared in the input.

Examples

input

```
2
1 2
2 1
```

output

```
2 0
2 0
```

input

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

output

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

C. Prime Swaps

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

You have an array $a[1], a[2], \ldots, a[n]$, containing distinct integers from 1 to n. Your task is to sort this array in increasing order with the following operation (you may need to apply it multiple times):

- choose two indexes, i and j ($1 \le i < j \le n$; $(j - i + 1)$ is a prime number);
- swap the elements on positions i and j; in other words, you are allowed to apply the following sequence of assignments: $tmp = a[i], a[i] = a[j], a[j] = tmp$ (tmp is a temporary variable).

You do not need to minimize the number of used operations. However, you need to make sure that there are at most $5n$ operations.

Input

The first line contains integer n $(1 \le n \le 10^5)$. The next line contains n distinct integers $a[1], a[2], \ldots, a[n]$ $(1 \le a[i] \le n)$.

Output

In the first line, print integer k $(0 \le k \le 5n)$ — the number of used operations. Next, print the operations. Each operation must be printed as "i j" ($1 \le i < j \le n$; $(j - i + 1)$ is a prime).

If there are multiple answers, you can print any of them.

Examples

input

```
3
3 2 1
```

output

```
1
1 3
```

input

```
2
1 2
```

output

```
0
```

input

```
4
4 2 3 1
```

output

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

D. Prefixes and Suffixes

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

You have a string $s = s_1s_2...s_{|s|}$, where $|s|$ is the length of string s, and s_i its i-th character.

Let's introduce several definitions:

- A substring $s[i..j]$ $(1 \le i \le j \le |s|)$ of string s is string $s_is_{i+1}...s_j$.
- The prefix of string s of length l $(1 \le l \le |s|)$ is string $s[1..l]$.
- The suffix of string s of length l $(1 \le l \le |s|)$ is string $s[|s| - l + 1..|s|]$.

Your task is, for any prefix of string s which matches a suffix of string s, print the number of times it occurs in string s as a substring.

Input

The single line contains a sequence of characters $s_1s_2...s_{|s|}$ $(1 \le |s| \le 10^5)$ — string s. The string only consists of uppercase English letters.

Output

In the first line, print integer k $(0 \le k \le |s|)$ — the number of prefixes that match a suffix of string s. Next print k lines, in each line print two integers l_i c_i. Numbers l_i c_i mean that the prefix of the length l_i matches the suffix of length l_i and occurs in string s as a substring c_i times. Print pairs l_i c_i **in the order of increasing** l_i.

Examples

input

```
ABACABA
```

output

```
3
1 4
3 2
7 1
```

input

```
AAA
```

output

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

E. Square Tiling

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

You have an $n \times m$ rectangle table, its cells are not initially painted. Your task is to paint all cells of the table. The resulting picture should be a tiling of the table with squares. More formally:

- each cell must be painted some color (the colors are marked by uppercase Latin letters);
- we will assume that two cells of the table are connected if they are of the same color and share a side; each connected region of the table must form a square.

Given n and m, find lexicographically minimum coloring of the table that meets the described properties.

Input

The first line contains two integers, n and m $(1 \leq n, m \leq 100)$.

Output

Print lexicographically minimum coloring of the table that meets the described conditions.

One coloring (let's call it X) is considered lexicographically less than the other one (let's call it Y), if:

- consider all the table cells from left to right and from top to bottom (first, the first cell in the first row, then the second cell in the first row and so on);
- let's find in this order the first cell that has distinct colors in two colorings;
- the letter that marks the color of the cell in X, goes alphabetically before the letter that marks the color of the cell in Y.

Examples

input

```
1 3
```

output

```
ABA
```

input

```
2 2
```

output

```
AA
AA
```

input

```
3 4
```

output

```
AAAB
AAAC
AAAB
```