School Team Contest #3 (Winter Computer School 2010/11)

A. Codecraft III

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

Today Vasya visited a widely known site and learned that the continuation of his favourite game Codecraft II will appear after exactly k months. He looked at the calendar and learned that at the moment is the month number s. Vasya immediately got interested in what month Codecraft III will appear. Help him understand that.

All the twelve months in Vasya's calendar are named using their usual English names: `January`, `February`, `March`, `April`, `May`, `June`, `July`, `August`, `September`, `October`, `November`, `December`.

Input

The first input line contains the name of the current month. It is guaranteed that it is a proper English name of one of twelve months. The first letter is uppercase, the rest are lowercase. The second line contains integer k $(0 \le k \le 100)$ — the number of months left till the appearance of Codecraft III.

Output

Print starting from an uppercase letter the name of the month in which the continuation of Codeforces II will appear. The printed name must be contained in the list `January`, `February`, `March`, `April`, `May`, `June`, `July`, `August`, `September`, `October`, `November`, `December`.

Sample test(s)

input
November 3
output
February

input
May 24
output
May

B. School

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

There are n students studying in the 6th grade, in group "B" of a berland secondary school. Every one of them has exactly one friend whom he calls when he has some news. Let us denote the friend of the person number i by $g(i)$. Note that the friendships are not mutual, i.e. $g(g(i))$ is not necessarily equal to i.

On day i the person numbered as a_i learns the news with the rating of b_i ($b_i \geq 1$). He phones the friend immediately and tells it. While he is doing it, the news becomes old and its rating falls a little and becomes equal to b_i - 1. The friend does the same thing — he also calls his friend and also tells the news. The friend of the friend gets the news already rated as b_i - 2. It all continues until the rating of the news reaches zero as nobody wants to tell the news with zero rating.

More formally, everybody acts like this: if a person x learns the news with a non-zero rating y, he calls his friend $g(i)$ and his friend learns the news with the rating of y - 1 and, if it is possible, continues the process.

Let us note that during a day one and the same person may call his friend and tell him one and the same news with different ratings. Thus, the news with the rating of b_i will lead to as much as b_i calls.

Your task is to count the values of res_i — how many students learned their first news on day i.

The values of b_i are known initially, whereas a_i is determined from the following formula:

$$a_i = ((v_i + res_{i-1} - 1) \mod n) + 1,$$

where mod stands for the operation of taking the excess from the cleavage, res_0 is considered equal to zero and v_i — some given integers.

Input

The first line contains two space-separated integers n and m ($2 \leq n, m \leq 10^5$) — the number of students and the number of days. The second line contains n space-separated integers $g(i)$ ($1 \leq g(i) \leq n, g(i) \neq i$) — the number of a friend of the i-th student. The third line contains m space-separated integers v_i ($1 \leq v_i \leq 10^7$). The fourth line contains m space-separated integers b_i ($1 \leq b_i \leq 10^7$).

Output

Print m lines containing one number each. The i-th line should contain res_i — for what number of students the first news they've learned over the m days in question, was the news number i. The number of the news is the number of the day on which it can be learned. The days are numbered starting from one in the order in which they are given in the input file. Don't output res_0.

Sample test(s)

input

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

output

```
1
1
1
0
```

input

```
8 6
7 6 4 2 3 5 5 7
10 4 3 8 9 1
1 1 1 2 2 2
```

output

```
1
1
1
2
1
1
```

C. Dancing Lessons

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

There are n people taking dancing lessons. Every person is characterized by his/her dancing skill a_i. At the beginning of the lesson they line up from left to right. While there is at least one couple of a boy and a girl in the line, the following process is repeated: the boy and girl who stand next to each other, having the minimal difference in dancing skills start to dance. If there are several such couples, the one first from the left starts to dance. After a couple leaves to dance, the line closes again, i.e. as a result the line is always continuous. The difference in dancing skills is understood as the absolute value of difference of a_i variable. Your task is to find out what pairs and in what order will start dancing.

Input

The first line contains an integer n ($1 \le n \le 2 \cdot 10^5$) — the number of people. The next line contains n symbols `B` or `G` without spaces. `B` stands for a boy, `G` stands for a girl. The third line contains n space-separated integers a_i ($1 \le a_i \le 10^7$) — the dancing skill. People are specified from left to right in the order in which they lined up.

Output

Print the resulting number of couples k. Then print k lines containing two numerals each — the numbers of people forming the couple. The people are numbered with integers from 1 to n from left to right. When a couple leaves to dance you shouldn't renumber the people. The numbers in one couple should be sorted in the increasing order. Print the couples in the order in which they leave to dance.

Sample test(s)

input

```
4
BGBG
4 2 4 3
```

output

```
2
3 4
1 2
```

input

```
4
BBGG
4 6 1 5
```

output

```
2
2 3
1 4
```

input

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

output

```
1
1 2
```

D. Event Dates

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

On a history lesson the teacher asked Vasya to name the dates when n famous events took place. He doesn't remembers the exact dates but he remembers a segment of days $[l_i, r_i]$ (inclusive) on which the event could have taken place. However Vasya also remembers that there was at most one event in one day. Help him choose such n dates of famous events that will fulfill both conditions. It is guaranteed that it is possible.

Input

The first line contains one integer n $(1 \le n \le 100)$ — the number of known events. Then follow n lines containing two integers l_i and r_i each $(1 \le l_i \le r_i \le 10^7)$ — the earliest acceptable date and the latest acceptable date of the i-th event.

Output

Print n numbers — the dates on which the events took place. If there are several solutions, print any of them. It is guaranteed that a solution exists.

Sample test(s)

input
3 1 2 2 3 3 4
output
1 2 3

input
2 1 3 1 3
output
1 2

E. Director

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

Vasya is a born Berland film director, he is currently working on a new blockbuster, "The Unexpected". Vasya knows from his own experience how important it is to choose the main characters' names and surnames wisely. He made up a list of n names and n surnames that he wants to use. Vasya haven't decided yet how to call characters, so he is free to match any name to any surname. Now he has to make the list of all the main characters in the following format: "$Name_1\ Surname_1,\ Name_2\ Surname_2,\ \ldots,\ Name_n\ Surname_n$", i.e. all the name-surname pairs should be separated by exactly one comma and exactly one space, and the name should be separated from the surname by exactly one space. First of all Vasya wants to maximize the number of the pairs, in which the name and the surname start from one letter. If there are several such variants, Vasya wants to get the lexicographically minimal one. Help him.

An answer will be verified a line in the format as is shown above, including the needed commas and spaces. It's the lexicographical minimality of such a line that needs to be ensured. The output line **shouldn't end with a space or with a comma**.

Input

The first input line contains number n ($1 \le n \le 100$) — the number of names and surnames. Then follow n lines — the list of names. Then follow n lines — the list of surnames. No two from those $2n$ strings match. Every name and surname is a non-empty string consisting of no more than 10 Latin letters. It is guaranteed that the first letter is uppercase and the rest are lowercase.

Output

The output data consist of a single line — the needed list. Note that **one should follow closely** the output data format!

Sample test(s)

input

```
4
Ann
Anna
Sabrina
John
Petrov
Ivanova
Stoltz
Abacaba
```

output

```
Ann Abacaba, Anna Ivanova, John Petrov, Sabrina Stoltz
```

input

```
4
Aa
Ab
Ac
Ba
Ad
Ae
Bb
Bc
```

output

```
Aa Ad, Ab Ae, Ac Bb, Ba Bc
```

F. Goats and Wolves

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

Once Vasya needed to transport m goats and m wolves from riverbank to the other as quickly as possible. The boat can hold n animals and Vasya, in addition, he is permitted to put less than n animals in the boat. If in one place (on one of the banks or in the boat) the wolves happen to strictly outnumber the goats, then the wolves eat the goats and Vasya gets upset. When Vasya swims on the boat from one shore to the other, he must take at least one animal to accompany him, otherwise he will get bored and he will, yet again, feel upset. When the boat reaches the bank, first all the animals get off simultaneously, and then the animals chosen by Vasya simultaneously get on the boat. That means that at the moment when the animals that have just arrived have already got off and the animals that are going to leave haven't yet got on, somebody might eat someone. Vasya needs to transport all the animals from one river bank to the other so that nobody eats anyone and Vasya doesn't get upset. What is the minimal number of times he will have to cross the river?

Input

The first line contains two space-separated numbers m and n ($1 \leq m, n \leq 10^5$) — the number of animals and the boat's capacity.

Output

If it is impossible to transport all the animals so that no one got upset, and all the goats survived, print -1. Otherwise print the single number — how many times Vasya will have to cross the river.

Sample test(s)

input
3 2
output
11

input
33 3
output
-1

Note

The first sample match to well-known problem for children.

G. Prime Problem

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

In Berland prime numbers are fashionable — the respectable citizens dwell only on the floors with numbers that are prime numbers. The numismatists value particularly high the coins with prime nominal values. All the prime days are announced holidays!

Yet even this is not enough to make the Berland people happy. On the main street of the capital stand n houses, numbered from 1 to n. The government decided to paint every house a color so that the sum of the numbers of the houses painted every color is a prime number.

However it turned out that not all the citizens approve of this decision — many of them protest because they don't want many colored houses on the capital's main street. That's why it is decided to use the minimal possible number of colors. The houses don't have to be painted consecutively, but every one of n houses should be painted some color. The one-colored houses should not stand consecutively, any way of painting is acceptable.

There are no more than 5 hours left before the start of painting, help the government find the way when the sum of house numbers for every color is a prime number and the number of used colors is minimal.

Input

The single input line contains an integer n ($2 \le n \le 6000$) — the number of houses on the main streets of the capital.

Output

Print the sequence of n numbers, where the i-th number stands for the number of color for house number i. Number the colors consecutively starting from 1. Any painting order is allowed. If there are several solutions to that problem, print any of them. If there's no such way of painting print the single number `-1`.

Sample test(s)

input
8
output
1 2 2 1 1 1 1 2

H. Road Problem

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

The Berland capital (as you very well know) contains n junctions, some pairs of which are connected by two-way roads. Unfortunately, the number of traffic jams in the capital has increased dramatically, that's why it was decided to build several new roads. Every road should connect two junctions.

The city administration noticed that in the cities of all the developed countries between any two roads one can drive along at least two paths so that the paths don't share any roads (but they may share the same junction). The administration decided to add the minimal number of roads so that this rules was fulfilled in the Berland capital as well. In the city road network should exist no more than one road between every pair of junctions before or after the reform.

Input

The first input line contains a pair of integers n, m ($2 \le n \le 900$, $1 \le m \le 100000$), where n is the number of junctions and m is the number of roads. Each of the following m lines contains a description of a road that is given by the numbers of the connected junctions a_i, b_i ($1 \le a_i, b_i \le n$, $a_i \ne b_i$). The junctions are numbered from 1 to n. It is possible to reach any junction of the city from any other one moving along roads.

Output

On the first line print t — the number of added roads. Then on t lines print the descriptions of the added roads in the format of the input data. You can use any order of printing the roads themselves as well as the junctions linked by every road. If there are several solutions to that problem, print any of them.

If the capital doesn't need the reform, print the single number `0`.

If there's no solution, print the single number `-1`.

Sample test(s)

input
4 3 1 2 2 3 3 4
output
1 1 4

input
4 4 1 2 2 3 2 4 3 4
output
1 1 3

I. TCMCF+++

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

Vasya has gotten interested in programming contests in TCMCF+++ rules. On the contest n problems were suggested and every problem had a cost — a certain integral number of points (perhaps, negative or even equal to zero). According to TCMCF+++ rules, only accepted problems can earn points and the overall number of points of a contestant was equal to the product of the costs of all the problems he/she had completed. If a person didn't solve anything, then he/she didn't even appear in final standings and wasn't considered as participant. Vasya understood that to get the maximal number of points it is not always useful to solve all the problems. Unfortunately, he understood it only after the contest was finished. Now he asks you to help him: find out what problems he had to solve to earn the maximal number of points.

Input

The first line contains an integer n ($1 \le n \le 100$) — the number of the suggested problems. The next line contains n space-separated integers c_i ($-100 \le c_i \le 100$) — the cost of the i-th task. The tasks' costs may coincide.

Output

Print space-separated **the costs of the problems** that needed to be solved to get the maximal possible number of points. Do not forget, please, that it was necessary to solve at least one problem. If there are several solutions to that problem, print any of them.

Sample test(s)

input
5 1 2 -3 3 3
output
3 1 2 3

input
13 100 100 100 100 100 100 100 100 100 100 100 100 100
output
100 100 100 100 100 100 100 100 100 100 100 100 100

input
4 -2 -2 -2 -2
output
-2 -2 -2 -2

J. Planting Trees

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

Vasya is a *Greencode* wildlife preservation society proponent. One day he found an empty field nobody owned, divided it into $n \times m$ squares and decided to plant a forest there. Vasya will plant nm trees of all different heights from 1 to nm. For his forest to look more natural he wants any two trees growing in the side neighbouring squares to have the absolute value of difference in heights to be strictly more than 1. Help Vasya: make the plan of the forest planting for which this condition is fulfilled.

Input

The first line contains two space-separated integers n and m ($1 \le n, m \le 100$) — the number of rows and columns on Vasya's field

Output

If there's no solution, print `-1`. Otherwise, print n lines containing m numbers each — the trees' planting plan. In every square of the plan the height of a tree that should be planted on this square should be written. If there are several solutions to that problem, print any of them.

Sample test(s)

input
2 3
output
3 6 2 5 1 4

input
2 1
output
-1