

# Problem A

## Contest Advancement

Time limit: 2 seconds

You are the RCD for your regional contest. It just ended and you now need to decide which teams qualify for the next level of competition. The teams are uniquely ranked (there are no ties). The next level has a limit on how many teams from any given school can participate. However, if that limit causes there to be not enough teams advancing, the remaining slots will be filled by teams from schools over the limit. If this happens, you will always give preference to higher-ranking teams, even if it means many teams from the same school advancing.

Output, in rank order, the teams that qualify for the next contest.

### Input

The first line of input contains three integers  $n$ ,  $k$  ( $1 \leq k < n \leq 10^5$ ) and  $c$  ( $1 \leq c \leq n$ ), where  $n$  is the number of teams in the competition,  $k$  is the number of teams that will advance, and  $c$  is the limit on the number of teams that may advance from any given school.

Each of the next  $n$  lines contains two integers  $t$  and  $s$  ( $1 \leq t, s \leq n$ ). Each line describes a team, with  $t$  being the team's unique ID, and  $s$  indicating the team's school. All team IDs will be distinct. The teams will be listed in rank order, with the highest ranking teams first.

### Output

Output  $k$  lines. On each line output a single integer, which is the ID of a team. List the IDs of the teams that qualified to advance in rank order.

#### Sample Input 1

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

#### Sample Output 1

```
3
1
4
9
2
6
8
```

This page is intentionally left blank.

# Problem B

## Digit Translation

Time limit: 1 second

You are given a string of lowercase letters. In one operation, if you can find a substring that is one of the written-out forms of one of the digits from zero to nine (“zero”, “one”, “two”, “three”, “four”, “five”, “six”, “seven”, “eight”, “nine”), you can replace that substring with the numeric digit.

Your goal is to find the shortest possible string you can end up with after applying zero or more of these operations, as well as how many distinct strings of that length there are.

### Input

The single line of input contains a string of lowercase letters with length at least one and at most  $10^6$ .

### Output

Output two separate lines.

On the first line output a single integer, which is the length of the shortest possible string.

On the second line output a single integer, which is the number of distinct strings of that length that can be obtained after applying zero or more of the specified operations, modulo 9302023.

#### Sample Input 1

icecreamcone

#### Sample Output 1

10  
1

#### Sample Input 2

onetwo

#### Sample Output 2

2  
1

#### Sample Input 3

twone

#### Sample Output 3

3  
2

This page is intentionally left blank.

# Problem C

## Don't Hunger Together

Time limit: 4 seconds

Ashley and Brandon are designing a survival video game called Don't Hunger Together. The game works as follows:

The game proceeds as a number of turns. Each turn consists of a day followed by a night. Several players have to survive for many turns in the wilderness. They collect food in the daytime and eat at night. During the daytime, they can scavenge up to a maximum amount of food, and the food found on that day must be eaten by some night in the future. Any leftover such food goes bad and is inedible. Each night, every player must eat a certain quantity of food, otherwise they will die of hunger. They win if every player is able to eat enough food on each of the nights.

Ashley and Brandon have designed a scenario and the last thing they need to do is pick the quantity of food that each player must eat every night. They wish to know the maximum possible value of this quantity, which must be positive. However, if the game is not winnable for any positive value, please let Ashley and Brandon know the scenario is impossible!

### Input

The first line of input contains two integers  $n$  ( $1 \leq n \leq 10^6$ ) and  $k$  ( $1 \leq k \leq 39$ ), where  $n$  is the number of turns in the game, and  $k$  is the number of players. The turns are numbered from 1 to  $n$ .

Each of the next  $n$  lines contains two integers  $q$  ( $0 \leq q \leq 10^9$ ) and  $f$  ( $i \leq f \leq n$ ), where  $q$  is the quantity of food that can be scavenged on that turn's day,  $f$  is the future turn's night after which the food goes bad, where  $i$  is the turn number. The turns are listed in order from 1 to  $n$ .

### Output

Output a single real number, which is the maximum positive quantity of food each player can eat on each night and still survive the scenario, or  $-1$  if the situation is not winnable for any positive value. Any answer within an absolute or relative error of  $10^{-9}$  will be accepted.

#### Sample Input 1

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

#### Sample Output 1

```
3.5
```

The 2023 ICPC North America Qualifier

**Sample Input 2**

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

**Sample Output 2**

```
1.5
```

**Sample Input 3**

```
1 17
0 1
```

**Sample Output 3**

```
-1
```

# Problem D

## Garden of Thorns

Time limit: 1 second

Eddy owns a rectangular garden and has noticed some trespassers stomping through his garden. There are some plants that he wants to protect. He hires an assistant, Zyra, to patrol and protect his garden.

Zyra cannot be bothered to monitor his garden, so she plants a circle of thorns centered at a randomly chosen location within the boundaries of his garden. A plant is considered protected if it is strictly inside the circle of thorns - that is, the distance from the plant to the center of the circle of thorns is less than the circle's radius. The circle of thorns may extend outside of the boundary of the rectangular garden, though all plants will be inside or on the boundary of the garden.

Given the random nature of the placement of Zyra's circle of thorns, compute the expected value of the plants that will be protected. Note that Zyra's circle of thorns does not have to be centered at integer coordinates.

### Input

The first line of input contains four integers  $n$  ( $1 \leq n \leq 10$ ),  $r$  ( $1 \leq r \leq 2,000$ ),  $w$  and  $h$  ( $1 \leq w, h \leq 1,000$ ), where  $n$  is the number of plants in Eddy's garden,  $r$  is the radius of Zyra's circle of thorns,  $w$  is the width of Eddy's garden and  $h$  is the height of the garden.

Each of the next  $n$  lines contains three integers  $x$  ( $0 \leq x \leq w$ ),  $y$  ( $0 \leq y \leq h$ ) and  $v$  ( $1 \leq v \leq 1,000$ ), where  $(x, y)$  denotes the position of a plant from the lower left corner of Eddy's garden, and  $v$  is the value of that plant. No two plants will be at the same position.

### Output

Output a single real number, which is the expected value of plants which will be protected by Zyra's circle of thorns. Any answer within an absolute or relative error of  $10^{-6}$  will be accepted.

#### Sample Input 1

```
3 50 100 100
30 10 3
40 10 7
50 90 8
```

#### Sample Output 1

```
8.41906486932450803806204930879
```

The 2023 ICPC North America Qualifier

**Sample Input 2**

```
2 5 3 4
0 0 10
3 4 15
```

**Sample Output 2**

```
25.0
```



# Problem E

## ICPC Team Generation

Time limit: 1 second

Vi is a coach for her university's ICPC organization and is working on creating teams for their upcoming regional contest. They recently competed in the North America Qualifier and Vi is using the results as well as each person's preferences to create as many teams of three as possible to send to regionals.

More specifically,  $n$  people from Vi's university competed in the North America Qualifier (NAQ), and each person got a unique rank from 1 to  $n$ . The person at rank  $r$  has two parameters,  $a_r$  and  $b_r$ , where  $a_r \leq r \leq b_r$ , indicating that their two teammates must have a rank between  $a_r$  and  $b_r$ , inclusive. Teams must have exactly three people.

Due to the collaborative environment, Vi notes that for every pair of individuals at ranks  $i$  and  $j$ , if  $i < j$ , then  $a_i \leq a_j$  and  $b_i \leq b_j$ .

Compute the maximum number of teams that Vi can send to regionals

### Input

The first line of input contains a single integer  $n$  ( $3 \leq n \leq 50$ ), which is the number of competitors in the local contest.

Each of the next  $n$  lines contains two integers  $a_r$  and  $b_r$  ( $a_r \leq r \leq b_r$ ), where  $r$  is the competitor's rank. These are the limits of the ranks of the competitors that can be teamed with this competitor. The competitors are given in rank order, from 1 to  $n$ . If  $i < j$ , then  $a_i \leq a_j$  and  $b_i \leq b_j$ .

### Output

Output a single integer, which is the maximum number of teams Vi can send to the regional contest.

#### Sample Input 1

```
6
1 2
1 2
2 5
2 6
2 6
5 6
```

#### Sample Output 1

```
1
```

This page is intentionally left blank.

# Problem F

## Is Y a Vowel?

Time limit: 1 second

The Vowels are **a**, **e**, **i**, **o** and **u**, and possibly **y**. People disagree on whether **y** is a vowel or not. Unfortunately for you, you have been tasked with counting the number of vowels in a word. You'll have to count how many vowels there are assuming **y** is a vowel, and assuming **y** is not.

### Input

The single line of input contains a string of at least one and at most 50 lowercase letters.

### Output

Output two space-separated integers. The first is the number of vowels assuming **y** is not a vowel, the second is the number of vowels assuming **y** is a vowel.

#### Sample Input 1

asdfiy	2 3
--------	-----

#### Sample Output 1

This page is intentionally left blank.

# Problem G

## Lines Per Hour

Time limit: 1 second

You are the chief judge for the next ICPC regional contest, and you want to make sure that it is theoretically possible for a team to solve all problems in the contest, but only if they are performing at their peak skill. With your insider knowledge, you know that the best team has a fixed limit on the number of lines of code per hour that they can successfully write.

You have a number of problems to select among and you know that each problem has a number of lines of code that would be required to solve it. You remember that regional contests are five hours long. What is the maximum number of problems you can put in the problem set so that the best team has a chance of solving all of the problems in the contest?

### Input

The first line of input contains two integers  $n$  ( $1 \leq n \leq 50$ ) and  $lph$  ( $1 \leq lph \leq 5,000$ ), where  $n$  is the number of candidate problems to choose from, and  $lph$  is the lines of code per hour that the best team can successfully write.

Each of the next  $n$  lines contains a single integer  $loc$  ( $1 \leq loc \leq 1,000$ ), which is the number of lines of code required to solve that problem.

### Output

Output a single integer, which is the maximum number of problems that can be put in the problem set so that the best team has a chance to solve all of the problems.

#### Sample Input 1

```
10 100
30
300
1000
20
35
19
84
117
212
98
```

#### Sample Output 1

```
7
```

This page is intentionally left blank.

# Problem H

## Magnesium Supplementation

Time limit: 1 second

At a recent doctor's appointment, Brandon has been marked as being deficient in magnesium. His doctor has advised him to take exactly some number of picograms of magnesium a day. His doctor can prescribe pills that have any integer number of picograms of magnesium from one up to a maximum. Brandon doesn't like taking pills though, so he refuses any prescription that requires him to take too many pills in a day.

Your task is to find all prescriptions that Brandon could be given. Note that all pills prescribed to Brandon must have the same dosage.

### Input

The single line of input contains three integers  $n$ ,  $k$  and  $p$  ( $1 \leq n, k, p \leq 10^{11}$ ), where  $n$  is the number of picograms per day that Brandon must take,  $k$  is the maximum amount in picograms of magnesium in any of the pills, and  $p$  is the maximum number of pills per day that Brandon is willing to take.

### Output

On the first line, output a single integer, which is the number of prescriptions that Brandon could be given.

On each subsequent line, output a single integer, which is the number of picograms per pill in the prescription. Output the prescriptions in increasing order.

#### Sample Input 1

6 6 4

#### Sample Output 1

3  
2  
3  
6

#### Sample Input 2

6 2 3

#### Sample Output 2

1  
2

This page is intentionally left blank.



# Problem I

## Missing Number

Time limit: 2 seconds

Kirby has written down all the integers from  $a$  to  $b$  in order. He wrote them down as one big string, with no spaces between them and no leading zeroes, and he didn't tell you the values of  $a$  and  $b$ !

Because Kirby was hungry, he ate one of the numbers. Can you figure out what number he ate?

### Input

The first line contains a single integer  $t$  ( $1 \leq t \leq 10^5$ ), which is the number of test cases that follow.

Each of the next  $t$  lines contains a string of digits of length at least one and at most 488888. This string is what Kirby wrote. Note that  $1 \leq a < b \leq 99999$ . It is guaranteed that this string can be obtained by the process given above.

It is guaranteed that the sum of the lengths of all the strings does not exceed  $10^6$ .

### Output

Output  $2t$  lines, two for each test case.

On the first line, output a single integer, which is the number of numbers that Kirby could have possibly eaten.

On the second line, output the numbers that Kirby could have eaten, in ascending order, separated by at least one space.

#### Sample Input 1

```
1
891112
```

#### Sample Output 1

```
1
10
```

This page is intentionally left blank.

# Problem J

## Tip of Your Tongue

Time limit: 4 seconds

Has this scenario ever happened to you? You're having a lovely conversation with someone, begin to say a word you've used hundreds of times, and the word escapes your mind, leaving you sputtering the first syllable. Well worry no more, because the Institute for Cancelling Pauses in Conversation (ICPC) has developed software to curtail this issue.

If there's a word on the tip of your tongue, enter the first few characters of the word into the ICPC software, and it will tell you how many words in the dictionary have those characters as a prefix. The ICPC quickly discovered that many of their clients don't only have tip-of-your-tongue problems, but also base-of-your-tongue problems, where users only remember some suffix of a word.

To be as useful as possible, the ICPC software needs to support the following queries:

1. **AND**  $p$   $s$ . Count the number of words in the dictionary that have  $p$  as a prefix and  $s$  as a suffix.
2. **OR**  $p$   $s$ . Count the number of words in the dictionary that have  $p$  as a prefix or  $s$  as a suffix.
3. **XOR**  $p$   $s$ . Count the number of words in the dictionary that have  $p$  as a prefix or  $s$  as a suffix, but not both.

Prefixes and suffixes of a word may be the whole word, and can overlap within the word. Due to a quirk in the ICPC software,  $p$  and  $s$  must be the same length. You must help the software developers answer all the incoming queries.

## Input

The first line of input contains two integers  $n$  and  $q$  ( $1 \leq n, q \leq 2 \cdot 10^5$ ), where  $n$  is the number of words in the dictionary and  $q$  is the number of queries.

Each of the next  $n$  lines contains a single string  $w$ , which is a word in the dictionary. All characters in dictionary words and queries are lowercase Latin characters ('a'–'z'). All words in the dictionary will be distinct.

Each of the next  $q$  lines contains three strings  $o$ ,  $p$ , and  $s$ , where  $o$  is one of the operations 'AND', 'OR', or 'XOR', and  $p$  and  $s$  are strings of one or more lowercase Latin characters ('a'–'z') with  $|p| = |s|$ .

The total number of characters in the dictionary and across all queries is at most  $10^6$ ; this does

not include query operations 'AND', 'OR', or 'XOR', whitespace, or newline characters. Said another way, there are at most  $10^6$  lowercase letters in the input.

## Output

Output  $q$  lines, one per query, in the order the queries were given. On each line output a single integer, which is the number of words in the dictionary that match that particular query.

### Sample Input 1

```
4 4
cat
catcat
octal
occidental
AND cat cat
OR oc at
AND ca at
XOR oc al
```

### Sample Output 1

```
2
4
2
0
```

# Problem K

## Very Important Edge

Time limit: 3 seconds

You are given a simple connected graph where each edge is assigned a non-negative weight. Recall that a minimum spanning tree of a graph is a connected, acyclic subset of the edges of the graph with minimum total weight. Find an edge which maximizes the minimum spanning tree weight of a given graph if that edge is deleted. It is guaranteed that the input graph remains connected after deleting any one edge.

### Input

The first line of input contains two integers  $n$  ( $3 \leq n \leq 10^5$ ) and  $m$  ( $3 \leq m \leq 10^6$ ), where  $n$  is the number of vertices and  $m$  is the number of edges in the input graph. The vertices are numbered from 1 to  $n$ .

Each of the next  $m$  lines contains three integers  $a, b$  ( $1 \leq a < b \leq n$ ) and  $w$  ( $1 \leq w \leq 10^6$ ). This denotes an edge between vertices  $a$  and  $b$  with weight  $w$ .

### Output

Output a single integer, which is the minimum spanning tree weight of the input graph after the right edge is deleted.

#### Sample Input 1

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

#### Sample Output 1

```
4
```

#### Sample Input 2

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

#### Sample Output 2

```
10
```

The 2023 ICPC North America Qualifier

**Sample Input 3**

```
5 7
2 5 8
1 3 19
4 5 9
1 5 15
1 2 14
3 4 16
2 4 15
```

**Sample Output 3**

```
54
```

# Problem L

## Water Journal

Time limit: 1 second

Every day, Eugene logs how many cups of water he drank. He forgot to log one entry in his journal though, but he remembers the minimum number and maximum number of cups of water he drank every day since he started logging his journal, and that there was at least one day where he drank the minimum and one day where he drank the maximum. Compute all possible amounts of cups of water he could have imbibed on the day he forgot to log.

### Input

The first line of input contains three integers  $n$  ( $2 \leq n \leq 50$ ),  $a$  and  $b$  ( $1 \leq a \leq b \leq 50$ ), where  $n$  is the number of days in Eugene's log,  $a$  is the minimum number of cups of water he drank on any day, and  $b$  is the associated maximum.

Each of the next  $n - 1$  lines contains a single integers  $w$  ( $a \leq w \leq b$ ). These are the entries in Eugene's log for every day except the day he forgot to log an entry.

### Output

Output in increasing order, all of the possible values for the missing entry separated by whitespace, or simply output -1 if Eugene's memory is faulty and it isn't possible.

#### Sample Input 1

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

#### Sample Output 1

```
1
2
3
4
5
```

#### Sample Input 2

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

#### Sample Output 2

```
5
```

This page is intentionally left blank.