WOBURN CHALLENGE

2015-16 On-Site Finals

Friday, May 6th, 2016

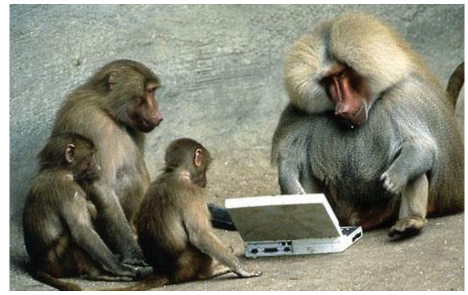
Junior Division Problems

Automated grading is available for these problems at:
wcipeg.com

For more problems from past contests, visit:
woburnchallenge.com

Problem J1: Vertical Posting

6 Points / Time Limit: 2.00s / Memory Limit: 16M



After 14 years of living in fearful hiding, the Head-Monkey and her band of primates have finally perfected their battle strategy to regain control of their native home of Scarberia. At last, they're ready to launch a ruthless series of attacks against the wicked cows who, under the leadership of Bo Vine, unjustly exiled the monkeys all those years ago!

Now, the cows had really become quite content with their Scarberian lives, free to eat their feed on farmer Daniel MacDonald's farm in peace. However, they weren't about to be taken by surprise by this turn of events. As everyone knows, all cows possess a geomagnetic sixth sense, which historically guided their ancestors during migrations. One day, while Bo Vine was out chewing his cud, his particularly potent sixth sense picked up on something rather peculiar. Detecting a disturbance in the electromagnetic field surrounding the Sacred Farm, Bo Vine swiftly realized that the monkeys' long overdue counterattack was imminent! As fast as he could, he rushed back to the barn, where the lethargic farmer had been nibbling a stalk of wheat. Standing before him, Bo Vine let out a thunderous bellow which could be heard throughout the entire countryside – "Dang, Daniel! The monkeys are back at it again with the bright plans!"

Just what is the monkeys' bright plan? Well for starters, they know that the imminent war will be fought in 2016. Preparing to be belligerents of the 21st century, they concluded that the only true way of declaring war is by trash-talking their enemies on the Internet. A recent trend on many online forums is known as *vertical posting*, where the poster writes a message vertically and horizontally, joined by the first letter. For example, the word `BRAVO` can be vertically-posted as follows:

```
B R A V O
R
A
V
O
```

Note that, to account for the difference in dimensions of most computer fonts, a single space must be added between adjacent horizontal letters.

This method of spamming is a very effective in provoking rage amongst members of a community. As such, the monkeys plan on vertically posting nasty messages over the cows' social networks! The problem is, it takes lots of time and effort to manually create such hateful (yet beautiful) works of art in a text editor. To speed up this process, the Head-Monkey has designated you to write a program to automatically generate the vertical posting from any desired message.

Input Format

The first and only line of input consists of a single string of at most 20 uppercase/lowercase letters from A to Z.

Output Format

Output the resulting vertical posting, preserving the case of every letter in the input. Your output must match the format of the sample output exactly!

Sample Input

```
getREKTcows
```

Sample Output

```
g e t R E K T c o w s
e
t
R
E
K
T
c
o
w
s
```

Problem J2: The Oxford Comma

10 Points / Time Limit: 2.00s / Memory Limit: 16M

Knowing that the monkeys are back to take over Scarberia, Bo Vine and Old MacDonald determined that it was time to plan their defense strategy. Bo, a master tactician, knows that the best defense is a good offense. He has thus contracted the services of his good friends, the oxen, who are some of the most hardened mercenaries the world has ever seen. Bo Vine instructed the oxen to report back at once with a battle plan, consisting of a sequence of sentences. Each sentence consists of a sequence of words terminated by a single period ("."). A word is a string of non-whitespace characters (lowercase/uppercase letters from A to Z, as well as digits from 0 to 9). Words are case sensitive, so for example, "ox" and "OX" are not considered the same word. Adjacent words in any given sentence are separated by a single space. Every word in a sentence except for the last is optionally followed by a comma, which comes before the adjoining space before the next word. Every sentence ends with a period, and every sentence except for the last one is followed by a single space to separate it from the following sentence.

After slaving away through the night, the oxen were able to produce a thick stack of papers containing their battle plan, which they laid out in front of the cow lord. However, upon reading it over, Bo Vine was less than impressed by its grammar. "You call yourselves oxen? No ox of mine neglects to follow *The Oxford Style Guide*. Look at all of these missing Oxford commas! Just imagine if our enemies got a hold of this. They'd mock us and question your status as true oxen!"

In English language punctuation, the Oxford comma refers to the comma placed immediately before the coordinating conjunction (usually the words *and* or *or*) in a list. For example, a list of cattle breeds may be written "Jersey, Holstein and Angus" (without the Oxford comma) or "Jersey, Holstein, and Angus" (with the Oxford comma). While controversial, there are many benefits of the Oxford comma, such as resolving ambiguity. For example, Bo Vine may write his dedication of the battle plan as "*To my parents, Bessie and Buttercup*" or as "*To my parents, Bessie, and Buttercup*". The former phrase implies that he considers Bessie and Buttercup to be parental figures, while the latter phrase implies that Bessie and Buttercup are simply two other individuals to whom he's choosing to dedicate the work. For the purposes of this problem, we will always choose to follow the latter interpretation.

A list is a substring of a sentence with three or more entries in one of the following two forms:

[entry], …, [entry](,) and [entry]

or

[entry], …, [entry](,) or [entry]

where each [entry] is a single word and (,) means that the second-to-last entry on the list *may or may not be immediately followed by the Oxford comma*. Every other entry in the list must be directly followed by a comma, except for the last entry, which may be directly followed by a comma, period, or space. For each sentence, it is guaranteed that either there is exactly one instance of the word *and* (but no instances of *or*), exactly one instance of the word *or* (but no instances of *and*), or no instances of either word. Your task is: given the battle plan produced by Bo Vine's ox mercenaries, insert the Oxford comma into each list which is missing one. Everything else should be left the same.

Input Format

The first line of input consists of a single integer representing the number of words in the battle plan.

The second line of input contains the battle plan itself, in the format described above. The battle plan will be at most 1000 characters in length.

Output Format

Output a single line, the same battle plan but with an Oxford comma added to every list which doesn't already have one, in accordance with the specifications above.

Sample Input 1

```
24
Before we engage in battle, each cow soldier may choose to have spam, ham or eggs.
Good luck and that will be all.
```

Sample Output 1

```
Before we engage in battle, each cow soldier may choose to have spam, ham, or eggs.
Good luck and that will be all.
```

Sample Input 2

```
22
To my parents, Bessie And Buttercup. They taught me that the breakfast of a healthy
cow consists of coffee, grass and toast.
```

Sample Output 2

```
To my parents, Bessie And Buttercup. They taught me that the breakfast of a healthy
cow consists of coffee, grass, and toast.
```

Problem J3: FuzzBiz

18 Points / Time Limit: 3.00s / Memory Limit: 32M

To win the upcoming battle against the cows, the Head-Monkey will need a great deal of money to purchase supplies and ammunition. Alas, the monkeys have been in hiding for so many years that they were never able to find jobs and make money. To make up for this financial deficit, the Head-Monkey has decided to run a business on the side. Unfortunately, the monkeys do not possess many belongings that are in demand by the market... except for one – monkey fur is one of the most luxurious and prized materials in the fashion industry! The smooth, silky locks of any monkey's fur are in high demand by the world's finest coat designers and couturiers. In a stroke of genius, the Head-Monkey ordered all of her subordinates to shave off the fuzz on their bodies and donate it to the business. Before long, sales seemed to be booming – and yet, there were still some problems.

Monkeys aren't particularly renowned for their arithmetic. As sales grew, accounting errors were slipping through the cracks, resulting in unnecessary losses to their income. To practice their division skills, the Head-Monkey proposed that her accountants play the classic game of *FizzBuzz*. In FizzBuzz, players sit around in a circle and take turns counting incrementally, replacing any number divisible by three with the word "fizz", and any number divisible by five with the word "buzz". Numbers divisible by both three and five are replaced with "fizzbuzz". The game ends once a player misspeaks or once they hit N ($1 \le N \le 10^9$). For example, a perfect game of FizzBuzz where $N = 16$ would sound like this:

1, 2, fizz, 4, buzz, fizz, 7, 8, fizz, buzz, 11, fizz, 13, 14, fizzbuzz, 16

Monkeys aren't particularly renowned for their articulateness. When the game gets long, no monkey is a fan of saying "…, five thousand seven hundred and ninety two, fizz, five thousand seven hundred and ninety four, buzz, …". Thus, the Head-Monkey invented her own variation of the game – FizzBuzzOok – which is streamlined for primates. The rules of FizzBuzzOok are almost identical to FizzBuzz, except that whenever a number is supposed to be said in FizzBuzz, the player should instead say "ook". As such, a perfect game of FizzBuzzOok where $N = 16$ would sound like this:

ook, ook, fizz, ook, buzz, fizz, ook, ook, fizz, buzz, ook, fizz, ook, ook, fizzbuzz, ook

Monkeys aren't particularly renowned for their memory. Sometimes, games of FizzBuzzOok would have to be paused as the monkeys went on banana breaks. When they returned, they would quickly lose track of where they had stopped! However, they would be able to remember what their desired goal of N was, along with what may have been said in the past M ($1 \le M \le 10^5$; $M \le N$) turns. For example, the monkeys may remember that they were going for a game of $N = 16$ turns and vaguely recall the last $M = 3$ turns had been "ook, fizz, ook". In this case, there are two possible positions at which they could have paused – after either turn 4 or 13.

Given a sequence of words representing the past M turns, your task is to determine the number of possible positions at which the accountants could have paused, within a perfect game of N turns. Their recollection of what was said in the past M turns could very well be faulty, in which case the given sequence may not be a contiguous subsequence of the N turns of the game. In that case, you should report 0 as the number of possible positions.

In test cases worth 6/18 of the points, $N \le 1000$ and $M \le 1000$.

Input Format

The first line of input consists of two space-separated integers N and M.
M lines follow, with the i-th of these lines consisting of a single word that is either "ook", "fizz", "buzz", or "fizzbuzz" (for i = 1..M).

Output Format

Output a single integer – the number of possible positions that the game could have been paused at, or 0 if the given sequence of turns will never occur in a game of FizzBuzzOok up to N.

Sample Input 1

```
16 3
ook
fizz
ook
```

Sample Output 1

```
2
```

Sample Input 2

```
16 3
ook
buzz
fizzbuzz
```

Sample Output 2

```
0
```

Problem J4: Hydration

28 Points / Time Limit: 4.00s / Memory Limit: 64M

The monkeys have made a booming business out of selling their fur, and their newfound cash flow has been directed towards purchasing all kinds of dangerous weapons to be used against the cows. While the cows have sensed the general fact that the monkeys are about to strike, they are puzzled by how their enemies could be acquiring the necessary funds. Watching shipments of nondescript boxes move in and out of the monkeys' lair has put the entirety of Old MacDonald's farm on edge. To address the frustrating turn of events, Bo Vine has decided to send an elite group of N ($1 \le N \le 10^6$) cow spies on an important mission. This group of highly trained agents are to infiltrate the Sacred Tree, gathering intelligence about the monkeys' source of income and other vile plans! There's just one problem... he's noticed that the cows are all rather thirsty. Needing to ask for water during this assignment could very well blow their cover, so Bo Vine needs to ensure that they all get a drink before heading out! Fortunately, there are M ($1 \le M \le 10^6$) water troughs available in the barn.

The i-th cow is C_i ($1 \le C_i \le 10^9$) centimetres tall, while the i-th trough is T_i ($1 \le T_i \le 10^9$) centimetres high. Each cow is willing to drink from a trough if it's no taller than them, and no more than K ($0 \le K \le 10^9$) centimetres shorter than them – in other words, cow i can drink from trough j if $C_i - K \le T_j \le C_i$. Each cow will need to drink from a trough of their choice for exactly 1 minute. The cows certainly like their privacy, so during each minute, each trough can be used by at most one cow. With little time left until the start of their mission, Bo Vine needs your help to determine the minimum amount of time required for all of them to get a drink, if it's even possible!

In test cases worth 14/28 of the points, $N \le 100$ and $M \le 100$.

Input Format

The first line of input consists of three space-separated integers N, M, and K.
N lines follow, with the i-th of these lines consisting of a single integer C_i (for $i = 1..N$).
M lines follow, with the i-th of these lines consisting of a single integer T_i (for $i = 1..M$).

Output Format

Output a single integer – the minimum number of minutes required for all of the cows to drink from a trough, or −1 if it's impossible.

Sample Input 1

```
4 4 5
13
24
10
13
20
8
14
22
```

Sample Output 1

```
3
```

Sample Explanation 1

The following is one optimal sequence of events:

- 1st minute: The 4th cow drinks from trough 2 and the 2nd cow drinks from trough 4.
- 2nd minute: The 3rd cow drinks from trough 2.
- 3rd minute: The 1st cow drinks from trough 2.

Sample Input 2

```
2 1 0
1
1000000000
99999999
```

Sample Output 2

```
-1
```

Problem J5: Driving Range

38 Points / Time Limit: 3.00s / Memory Limit: 32M

With intel from the Sacred Tree stolen by hydrated bovine spies, times are getting ever more desperate for the monkeys. The Head-Monkey has decided it's time to send out her own elite special agent – Agent Tiny. Tiny has been ordered to journey to Old MacDonald's farm and perform a precision strike against the cows' military headquarters. For this extremely dangerous mission, Tiny is being outfitted with a sophisticated piece of monkey technology – a fully functional car made of branches and leaves, and powered by banana juice. This car will offer him more than enough speed and maneuverability to infiltrate the barn. Also, it can shoot missiles from its front headlights.

In an effort to ensure that Tiny actually accomplishes something useful with his car in the field, the Head-Monkey has instructed him to take it for a spin on the monkeys' specialized driving course. The course will take place in a very large room, which can be represented as a 2D grid of cells, with some targets on the walls. The rows are numbered in increasing order from North to South (starting from 1), and similarly the columns are numbered in increasing order from West to East. Tiny will be required to drive around and fire missiles at all of the targets.

Tiny will start in the Northwestern-most cell (in the first row and first column), with his car facing South. Each second, he may either drive to an adjacent cell, or fire a missile in the direction that his car is currently facing. If he chooses to drive, he may do so in one of the four cardinal directions (North, South, East, or West) as long as he stays within the room (he can't drive North from row 1 or West from column 1), and his car will be left facing the direction in which he just drove.

The Head-Monkey will add N ($1 \le N \le 10^5$) targets to the course, one after another, and make Tiny complete the current course after each one is added. Each time, Tiny will have to start in the Northwestern-most cell and hit all of the targets which have been added so far with missiles, in any order. That is, he will have to complete the course N separate times, and on his i-th run, he'll be required to hit targets 1..i. The i-th target's position is described by a character D_i (either "`R`" or "`C`") and an integer P_i ($1 \le P_i \le 10^9$). If D_i = "`R`", then the i-th target is at the far East end of row P_i, meaning that it can be hit by firing a missile Eastward from any cell on row P_i. Otherwise if D_i = "`C`", then the i-th target is instead at the far South end of column P_i. No two targets are at the same location.

To help convince the Head-Monkey of Tiny's driving skills, can you help Tiny determine how quickly the course can be completed each of the N times?

In test cases worth 8/38 of the points, $N \le 10$ and $P_i \le 10$.
In test cases worth another 14/38 of the points, $N \le 40$ and $P_i \le 40$.

Input Format

The first line of input consists of a single integer N.
The next N lines each consist of a character and integer D_i and P_i, separated by a space, for i = 1..N.

Output Format

Output N lines, one integer per line. The i-th line of output (for i = 1..N) should consist of the minimum number of seconds required to complete the course for the i-th time.

Sample Input 1

```
3
R 3
C 2
C 3
```

Sample Output 1

```
4
6
8
```

Sample Explanation 1

The third and final time that Tiny completes the course, one optimal sequence of actions that he can take is as follows:

- Drive East
- Drive South
- Fire a missile (hitting target 2)
- Drive East
- Drive South
- Fire a missile (hitting target 3)
- Drive East
- Fire a missile (hitting target 1)

Sample Input 2

```
2
R 1
R 10
```

Sample Output 2

```
2
13
```