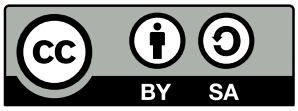
ICPC SouthWestern Europe Regional Practice Contest 2024

Lisbon, Lyon, Pisa, Nov 30th, 2024

Judges and Problem Setters

- Guillaume Aubian (Université Paris-Cité)
- Alessandro Bortolin (University of Milan)
- Thomas Deniau
- Raphaël Marinier (Google), *chief judge*
- Henrique Navas (Técnico, ULisbon)
- Jaime Ramos (Técnico, ULisbon)
- Kevin Pucci (UPorto)
- Cheng Zhong (Google), *deputy chief judge*

This problem set consists of 4 problems, on 8 pages.



A: Find the Tomb

You are part of an archeological expedition trying to locate a hidden chamber inside the Egyptian Great Pyramid, possibly containing the tomb of Pharaoh Khufu.

After long investigations, you have found a wall behind which a hidden chamber lies. To find the exact location of the cavity, you have at your disposition a tomography scanner, which, when you drill a hole through the wall to insert the scanner, can tell you if the cavity lies between the position of the scanner and the end of the wall.

You know the position of the chamber is at an integer position between 1 and 1000 (inclusive) on the X axis. If you drill a hole at position i, the scanner tells you if the cavity lies in $[i, 1000]$ or not. Of course, if you drill at the correct position, you immediately find the hidden chamber.

Input and Output

This is an interactive problem. As such, no initial input is provided in this problem, and you get to ask a question first.

Your program should output guesses for the correct position, in the form of an integer i between 1 and 1000 (inclusive) on a line on its own. After making each guess, you need to make sure to flush the standard output.

After each guess, there will be a response to be read from the standard input. This response is a line with one of the following three words:

- `hollow` if the chamber lies in $(i, 1000]$;
- `solid` if chamber does not lie in $[i, 1000]$
- `found` if you have found the chamber.

After having guessed the right answer your program should exit. If you guess incorrectly 10 times, you won't get any more chances, as you are drilling into the pyramid too much, and your program (and your research expedition) will be terminated.

Sample Interaction

In this section, to clarify what the input and output should be, "< " is printed before what your program can read on standard input and "> " is printed before what your program should output. Do not include these characters in your real input/output.

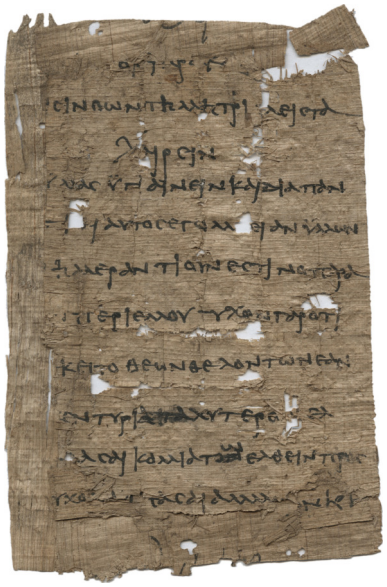
An interactive Python script where you can enter commands and see how the judge would respond is also available on the Web version of this problem.

```
> 41
< hollow
> 43
```

```
< solid
> 42
< found
```

Image credits: Detail of the north side of the Cheops pyramid, by Olaf Tausch, CC BY 3.0, via Wikimedia Commons

B: Etymology



In the remains of a distant civilization, an ancient manuscript was discovered containing N words $W_1, W_2, ..., W_N$. It was observed that some words may share common prefixes, thus prompting some linguistic studies. Your task is to find a prefix P whose length is exactly L, such that at least M out of the N words are prefixed with P. Notably, a word is also considered a prefix of itself.

If there are multiple valid prefixes, output the smallest one in lexicographical order. If no such prefix exists, output a single integer 0.

Input

The first line contains three integers N, M, L, respectively the number of words, minimum number of words with prefix P, and the length of P. The N words $W_1, W_2, ..., W_N$ then follow, each on its own line.

Output

The output should contain the lexicographically smallest prefix P you can find, or a single integer 0 if no such prefix exists.

Limits

- $1 \leqslant M \leqslant N \leqslant 50\,000$;
- $1 \leqslant L \leqslant 20$;
- $1 \leqslant |W_i| \leqslant 20$ for all $i \leqslant N$;
- These N words are distinct and each consists of only lowercase English letters.

Sample Input 1

```
6 2 3
practice
prefer
protect
predict
progress
prepare
```

Sample Output 1

```
pre
```

Sample Explanation 1

Note that "pre" is the prefix of 3 words, "pro" is the prefix of 2 words, so both of them are valid. Since "pre" is lexicographically smaller, we output it.

Sample Input 2

```
6 2 4
practice
prefer
protect
predict
progress
prepare
```

Sample Output 2

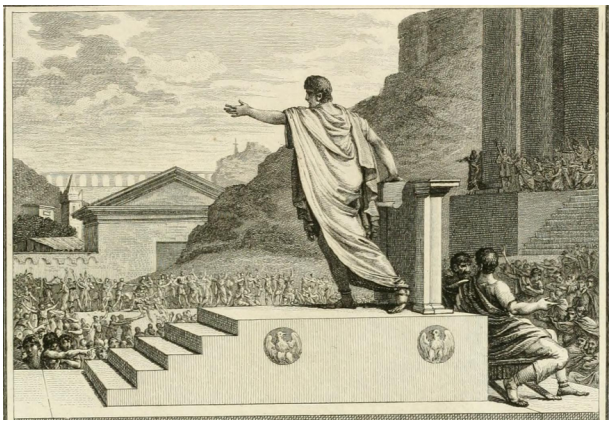
```
0
```

Sample Explanation 2

The difference from the first sample is that $L = 4$. Since there is no valid prefix of length 4, we output a single integer 0.

Image credits: Greek papyrus, UBC Library Digitization Centre, no restrictions, via Wikimedia Commons

C: Election



In the dawn of a venerable republic, the people are electing a consul through the Centuriate Assembly. There are two candidates for consul. Each centuria has a certain voting weight and will fully cast its vote for one of the candidates. The candidate who receives the largest sum of voting weights will be elected as consul. In the event of a tie, it is agreed that the first candidate will be elected.

As an augur, you are interested in the probability of the first candidate being elected. By looking at the flight of ravens, you have discerned the probability of each centuria voting for the first candidate. You can therefore now calculate the probability that the first candidate will ultimately be elected as consul.

Input

The first line contains an integer N, represents the number of centuriæ (the plural of centuria). The second line contains N integers; the i^{th} one w_i is the voting weight of the i^{th} centuria. The third line contains N real numbers; the i^{th} one p_i is the probability of the i^{th} centuria voting for the first candidate.

Output

A single real number, the probability of the first candidate being elected as consul. Answers within 10^{-4} of the correct answer will be accepted.

Limits

- $1 \leqslant N \leqslant 1000$;
- $1 \leqslant w_i \leqslant 100$ for all $i \leqslant N$;
- $1 \leqslant w_1 + w_2 + ... + w_N \leqslant 1000$;
- $0.0 \leqslant p_i \leqslant 1.0$ for all $i \leqslant N$.

Sample Input

```
3
1 1 2
0.5 0.6 0.7
```

Sample Output

```
0.790000
```

Sample Explanation

Out of a total voting weight of $1 + 1 + 2 = 4$, the first candidate needs to secure at least a voting weight of 2 to be elected.

If the third centuria votes for the first candidate, then regardless of how the first two centuriæ vote, the first candidate will surely be elected. Otherwise, the first candidate can only be elected if the votes of both of the first two centuriæ are secured.

Therefore, the probability of the first candidate being elected is $0.7 + (1 - 0.7) \times 0.5 \times 0.6 = 0.79$.

Image credits: Gracchus presiding over the Plebeian Council, Silvestre David Mirys, Public domain, via Wikimedia Commons

D: Help The Organizers

This year's SWERC is spread across three sites: Lisbon, Lyon and Pisa. The system that judges the solutions from the contestants is shared among the three sites. The organizers are trying to determine the number of judge hosts needed so that the contestants get judging verdicts timely. A good rule of thumb is that a single host can handle at most 20 teams. Can you help the organizer determine the number of hosts that should be provisioned according to this rule of thumb?

Input

The first line contains three space-separated integers t_1, t_2, t_3, the number of teams in the sites Lisbon, Lyon and Pisa, respectively.

Output

A single integer: the number of hosts that should be provisioned according to the rule of thumb.

Limits

- $1 \leqslant t_i \leqslant 100$ for $i = 1, 2, 3$.

Sample Input

```
34 23 51
```

Sample Output

```
6
```

Photo credits: Los Alamos National Laboratory, Attribution, via Wikimedia Commons