

Matching Book Names and Descriptions



Our bot has crawled several product pages from the popular Indian e-commerce website, [Flipkart.com](https://www.flipkart.com). All of these pages are specifically about the most popular books being sold on Flipkart (at the time of the crawl).

Each page contained information for exactly one book. We noted down exactly two fields from each of these pages:

1. The name of the book.
2. Description fragment: The first few sentences of the description of the book, as displayed on the page. In some cases, this string or text field might be terminated prematurely (i.e., not exactly at a word or a sentence boundary).

Each of these text blocks is split into two parts of roughly equal length.

Set A contains the **names of all the books**. **Set B** contains the **description fragments** for all the books.

Both the Sets A and B are shuffled up, and the ordering of elements is lost.

Your task is to identify, for each name (a) in **Set A**, which is the correct corresponding text fragment (b) in **Set B**, such that, b was the descriptive fragment for the book named a .

Hints: Getting started - Think about using TF-IDF Scores (or a modification of it)

For those getting started with this fascinating domain of text classification, here's a wonderful Youtube video of Professor Christopher Manning from Stanford, explaining the TF-IDF, which you could consider using as a starting point.

Input Format

An Integer N on the first line. This is followed by $2N+1$ lines.

Text fragments (numbered 1 to N) from Set A, each on a new line (so a total of N lines).

A separator with five asterisk marks "*" which indicates the end of Set A and the start of Set B.

Text fragments (numbered 1 to N) from Set B, each on a new line (so a total of N lines).

Output Format

N lines, each containing one integer.

The i -th line should contain an integer j such that the j -th element of **Set A** and the i -th element of **Set B** are a pair, i.e., both originally came from the same listing page on Flipkart.

Constraints

1 <= N <= 1000

No text fragment will have more than 10000 characters.

Sample Input

```
5
How to Be a Domestic Goddess : Baking and the Art of Comfort Cooking (Paperback)
Embedded / Real-Time Systems 1st Edition (Paperback)
The Merchant of Venice (Paperback)
Lose a Kilo a Week (Paperback)
Few Things Left Unsaid (Paperback)
*****
Today the embedded systems are ubiquitous in occurrence, most significant in function and project an absolutely promising picture of developments in the near future.
The Merchant Of Venice is one of Shakespeare's best known plays.
How To Be A Domestic Goddess: Baking and the Art of Comfort Cooking is a bestselling cookbook by the famous chef Nigella Lawson who aims to introduce the art of baking through text with an emphasis.
Lose A Kilo A Week is a detailed diet and weight loss plan, and also shows how to maintain the ideal weight after reaching it.
Few Things Left Unsaid is a story of love, romance, and heartbreak.
```

Sample Output

```
2
3
1
4
5
```

Explanation

Explaining the Input

The first line indicates that the test case contains the names and descriptions of five popular books listed on Flipkart.

The next five lines are the names of the books (i.e, **Set A**). After that, we have a separator. That is followed by five lines, each containing description fragments from **Set B**.

Explaining how we arrived at the Output

The first description, is visibly most closely related to the second book (Embedded / Real-Time Systems 1st Edition (Paperback)).

The second description, is clearly about the Merchant of Venice - which is the third book name in Set-A. The third description is about Baking - and so, it corresponds to the first of the book names, in Set-A. Similarly, the fourth and fifth descriptions match best with the fourth and fifth book names (i.e, it so happens that they are already in order).

So, the expected output is 2, 3, 1, 4, 5 respectively.

Scoring

The weight for a test case will be proportional to the number of tests (book names) it contains. Two sample tests are available and visible on **Compile & Test**. A training driven approach or solution is not expected in this challenge, which is why no comprehensive training data has been provided.

Score = $M * (C)/N$ Where M is the Maximum Score for the test case.

C = Number of correct answers in your output.

N = Total number of book names in the test set (which were divided into Set A and Set B respectively).

Note: Submissions will be disqualified if it is evident that the code has been written in such a way that the sample test case answers are hard-coded, or similar approaches, where the answer is not computed, but arrived at by trying to ensure the code matches the sample answers.

Timelimits

Timelimits can be seen [here](#).

Libraries

Libraries available in our Machine Learning/Real Data challenges will be enabled for this contest and are listed [here](#). Please note, that occasionally, a few functions or modules might not work in the constraints of our infrastructure. For instance, some modules try to run multiple threads (and fail). So please try importing the library and functions and cross checking if they work in our online editor in case you plan to develop a solution locally, and then upload to our site.