

In NLP positional indexes is most commonly employed technique. In positional index, each term in the vocabulary, postings are stored in the form docID: (position1, position2, ...), as shown below Figure 2.1, where each position is a token index in the document. Each posting will also usually record the term frequency.

to, 993427:

```
{ 1, 6: {7, 18, 33, 72, 86, 231};  
  2, 5: {1, 17, 74, 222, 255};  
  4, 5: {8, 16, 190, 429, 433};  
  5, 2: {363, 367};  
  7, 3: {13, 23, 191}; ... }
```

be, 178239:

```
{ 1, 2: {17, 25};  
  4, 5: {17, 191, 291, 430, 434};  
  5, 3: {14, 19, 101}; ... }
```

► **Figure 2.1** Positional index example. The word to has a document frequency 993,477, and occurs 6 times in document 1 at positions 7, 18, 33, etc.

Write a program to build a positional index in linear time.

Once positional index is built, If a word is given, write a program to print all the documents in descending order of frequency of that word in those documents.

You were given 9 files(in Files folder) where each file contains single line input contains of lower case words separated with spaces. You need to build a positional index for all the words as shown in the above example.

In the testcases you were given one word where you need to print the positional index sorted on basis of word frequency in each document. If frequencies of two files are equal print the file with smaller Document id(Name of the file without file extension) first.

**Sample input:**

king

**Sample output:**

king:46

6, 18, [36, 72, 103, 132, 155, 200, 488, 529, 593, 722, 744, 753, 785, 827, 890, 1177, 1324, 1329]

7, 12, [8, 15, 25, 82, 124, 168, 219, 226, 240, 772, 947, 2016]

3, 8, [33, 41, 47, 75, 98, 240, 244, 262]

8, 8, [13, 54, 1736, 2210, 2262, 2318, 2626, 2655]

**Output format:**

First line of the output contains word and total frequency of that word separated with colon

From second line output contains three fields separated with comma space. The three fields are as follows

Document id(Name of the file without file extension)

Frequency of the word present in that file

List containing positions of the word present in that file

**Explanation:**

Word king appears 46 times in all the files

In file 6 word king appears 18 times at positions 36, 72, 103, 132, 155, 200, 488, 529, 593, 722, 744, 753, 785, 827, 890, 1177, 1324, 1329

In file 7 word king appears 12 times at positions 8, 15, 25, 82, 124, 168, 219, 226, 240, 772, 947, 2016

In file 3 word king appears 8 times at positions 33, 41, 47, 75, 98, 240, 244, 262

In file 8 word king appears 8 times at positions 13, 54, 1736, 2210, 2262, 2318, 2626, 2655

**Note: The output will be in descending order of the frequencies of the word in individual file. If frequencies of two files are equal print the file with smaller Document id(Name of the file without file extension) first.**

So in the above output files 3 and 8 has word king 8 times but 3 is less than 8 so 3 print first.

**Note: Positions of the words in file will start from 0.**

**Note: If the given word is not present in any of the files print “word is not present in any file”**