Coding Arena

Problem : Valid Segments 1

Consider text comprised of sentences and sentences comprised of words. Words in a sentence will be space delimited. Given a text and **K** strings, task is to find out the number valid segments whose length lies in the range **[lb, ub]**.

A segment is said to be valid if

- It contains all the **K**-strings
- It should start and end with any one of the **K**-strings

Length of a segment is defined as number of words in that segment.

Input Format:

First line contains the text. Next line contains **K, lb, ub**. Next **K** lines consist of strings to be searched in the text.

Note:

1. Comparison should be case insensitive.
2. Comparison should be based only on words comprised of alphabets. Non-alphabet characters such as Full Stop ("."), Exclamation Marks ("!") etc. are called as Stop words. Stop words must be removed from sentences before comparison.
3. If more than one segment start with the same index (i.e. position of the word in the text), then count them as one. In other words, segments should begin from unique indexes.
4. If there exists two segments which starts with the same index and ends at different indexes, then the one which ends at the near index (from the starting) should given the priority as compared with other, and other should be discarded.

Output Format:

Print the number of valid segments.

Constraints:

1<= total number of words in the text <= 30,000

1<= length of a single string in the input / word in the text<= 500

1<=K<=total number of words

1<=lb<=ub<=total number of words

Sample Input and Output

SNo.	Input	Output
1	The European market crashes on Mondays. Crashes in the European market are quite common. 2 3 4 European Crashes	2

Explanation:

Valid segments are:

1. **European market crashes**
2. crashes on Mondays. Crashes in the European
3. **Crashes in the European**

So 1 and 3 are the valid segments whose length is in the range [3, 4].

Note:

Please do not use package and namespace in your code. For object oriented languages your code should be written in one class.

Note: