Week 11 Programming Assignment 1

Due on 2020-04-15, 23:59 IST
In processing documents, we often build something called an index, or an inverted index. This is simply a set of pairs (WORD, OCCURRENCES) where WORD is a word occurring in the document, and OCCURRENCES is a sequence of integers giving the positions (in increasing order) at which the word occurs. For example, for a document consisting of the single sentence "ask not what your country can do for you ask what you can do for your country", the index will be

- ask 0 9
- can 5 12
- country 4 16
- do 6 13
- for 7 14
- not 1
- what 2 10
- you 8 11
- your 3 15

You are to write a program that reads in a document and prints the index, in lexicographic order. Your answer for the document above should be as given above, you will notice that the words in it are in lexicographical sorted order. You should handle two more complications. Each word may have a single terminating comma or semicolon or full stop. These should be removed when making the index. Also, capitalisation of letters should be disregarded. Thus the index should not change if the input above were given as "Ask not what your COUNTRY can do for you, ask what you can do for your country."

Finally note that the document is given as a sequence of white space separated words, with the single character word '*' being the last word. The word '*' should not be included in the index.

### Sample Test Cases

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Test Case 1</strong></td>
<td><strong>Output</strong></td>
</tr>
<tr>
<td>abc abc abc abc zxy, zxy, zxy, *</td>
<td>abc 0</td>
</tr>
<tr>
<td></td>
<td>1 2 3</td>
</tr>
<tr>
<td></td>
<td>zxy 4</td>
</tr>
<tr>
<td></td>
<td>5 6</td>
</tr>
<tr>
<td><strong>Test Case 2</strong></td>
<td><strong>Output</strong></td>
</tr>
<tr>
<td>The quick Brown Fox; the Quick brown fox. t</td>
<td>brown 2 6 1</td>
</tr>
<tr>
<td>He qUick bRown fOX, *</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>fox 3</td>
</tr>
<tr>
<td></td>
<td>7 11</td>
</tr>
<tr>
<td></td>
<td>quick 1 5 9</td>
</tr>
<tr>
<td></td>
<td>the 0</td>
</tr>
<tr>
<td></td>
<td>4 8</td>
</tr>
</tbody>
</table>
ask not what your country can do for you as what you can do for your country *

The due date for submitting this assignment has passed.
As per our records you have not submitted this assignment.
Sample solutions (Provided by instructor)

```cpp
#include <iostream>
#define repeat(x) for(int _iterator_i = 0, _iterator_limit = x; _iterator_i < _iterator_limit; _iterator_i++)
#define main_program int main()
#include <cmath>
using namespace std;
#include <iostream>
#include <string>
#include <iterator>
#include <map>
#include <vector>
using namespace std;

string clean(string nextword){
    char last = nextword[nextword.size()-1];
    if(last == ',' || last == '.' || last == ';')
        nextword = nextword.substr(0,nextword.size()-1);
    //cout << nextword << endl;
    for(int i=0; i<nextword.size(); i++)
        if('A' <= nextword[i] && nextword[i] <= 'Z')
            nextword[i] = nextword[i] + 'a' - 'A';
    return nextword;
}

int main(){
    map<string,vector<int>> index;
    for(int i=0; ; i++){
        string nextword; cin >> nextword;
        if(nextword == "*") break;
        nextword = clean(nextword);
        index[nextword].push_back(i);
    }
    for(auto e : index){
        cout << e.first;
        for(auto w : e.second) cout << ' ' << w;
        cout << endl;
    }
}```