Assignment 7

Key Words In Context – Maintenance of the System

We now maintain our Key Words In Context (KWIC) system.

a) Foreseen Changes

The following changes in the requirements occur over time:

1. The system should take the optional command line switch −**W** <regex> to specify a regular expression that defines what a word is. If no such switch is given, the default word regular expression is "`[^ \t]+`". (The output word breaks should continue to be single blanks.)

   See `man regex` about regular expressions.

2. The system should take the optional command line switch −**f** to specify that the lines should be folded to upper case before sorting. Otherwise, they should not be folded, as before.

3. The system should take the optional command line switch −**i** <file> to specify a file with a list of ignore words. The circular shifter excludes shifts which start with one of these words.

We assume that each of these changes occur at a different time, so please perform them separately. For each of these changes,

- identify the module or the modules affected,

- adjust the implementation to the new requirements,

- test it,

- hand in the file(s) of the changed module(s),

- write a brief explanation of what you actually changed in which files, maybe using a `diff` output. (This is just to make marking easier.)

In real life, of course, you should have updated the requirements document and the software internal implementation document first, before actually changing a line of code. (You may consider this assignment text as the updated requirements document.)

b)   Unforeseen Changes

Analyze the following idea for a change: the original system (without the extensions from part a)) should run within a fixed amount of RAM memory, independent of the size of the input file. As a trade-off, we are willing to accept a (much) longer execution time.

You don't need to implement this change. Instead, identify all modules which allocate variable-size memory depending on input, and describe whether it is possible to rewrite them. If yes, sketch a solution for how to do it.

Hints: think about a (less efficient) way to access a specific shifted line without keeping any auxiliary array of pointers to shifts. Think also about a (less efficient) way to access the lines in a sorted way without constructing any auxiliary array of pointers to the data to sort.

Describe whether you can do the improvements to the identified modules independently, i.e., whether different teams could work on this requirements change separately.

If you find that it is not easy to make such a change, analyze at which point in the development process which decision was made that prevents you from making the change easily.

Submit your solution by email (and, if possible, on paper) to `brederek@tzi.de` (as a PostScript or Pdf file).