Assignment 1:

The Cranfield collection is a standard IR text collection(included in this directory)., consisting of 1400 documents from the aerodynamics field.

1. Write a program that preprocesses the collection. This preprocessing stage should specifically include:
   a. Function that eliminates SGML tags
   b. Function that tokenizes the text. In doing this, pay particular attention to characters that need special handling, as discussed in the text (. , - etc.). For this task, please use _your own_ implementation of a tokenizer.

2. Determine the frequency of occurence for all the words in this collection.
Answer the following questions:
   a. What is the vocabulary size? (i.e. number of unique terms)
   b. What are the top 10 words in the ranking? (i.e. the words with the highest frequencies)
   c. From these top 10 words, which are "meaningful" (i.e. they are not stopwords), and which ones you would eliminate as "stopwords".
   d. What is the minimum number of unique words accounting for half of the total number of words in the collection?

Example: if the total number of words in the collection is 100,and we have the following word-frequency pairs: the - 30 of – 10 a - 10 clear - 8 cut - 7 etc. the answer to this question will be 3 (3 unique words account for half of the total 100 words)

3. Integrate the Porter stemmer and a stopword eliminator into your code. Answer again questions a-d from the previous point.
Check the Link below for various implementations of the Porter stemmer and the lists of stopwords).
Porter: http://tartarus.org/~martin/PorterStemmer/
Stopwords: http://www.ranks.nl/stopwords

4. Pick two subsets of this dataset, and determine the size of the vocabulary and the size of the subset you selected. Use this information to derive the K and beta parameters required by the application of the Heaps law. Use these values to predict what would be the vocabulary size if the corpus were to increase to 500,000 words. How about 2,000,000 words?

Note: It is highly recommended that your code is as modularized as possible; many of the functions that you implement during this assignment will be needed in future assignments.

Full Cranfield dataset:
http://ir.dcs.gla.ac.uk/resources/test_collections/cran/cran.tar.gz