LAB 2

Based on lecture 4

PREPROCESSING

- You need to have Perl or Python on your machine (you still can use something else).
- Download the following file: link, which contains:
  - Collection 1: 5 sample 1-line-documents used in lecture 5.
  - Collection 2: 1000 sample news articles.
  Note1: there are two versions of the files, XML and TXT. You are free to use the one you want for the lab. However, it worth noting that XML format is the more standard one in TREC IR test collections.
  Note 2: for the news articles collection, please include the headline of the article to the index.
  Note 3: these are very small collections just for experimentation
- Be sure that you have your preprocessing module ready (revise: lab 1), then apply it to the collections.
  if you didn't have it done, then at least get the tokeniser and casefolding ready for this lab
- Implement positional inverted index. You need to save the following information in terms inverted lists:
  - term (pre-processed) and its document frequency (optional)
  - list of documents where this term occured
  - for each document, list of positions where the term occured within the document
- Print output in a text file for visualisation. Example output here (*examples didn't apply stopping or stemming*)
  in practice (for assignment), you might save in a more effecient format (e.g. binary files).
- Please check the output inverted index when enable/disable the following:
  - stopping
  - stemming

RUNNING SEARCH ON INDEX

- Build a module that allows the following:
  - load index into memory
  - apply Boolean search (AND, OR, and NOT)
  - apply phrase search
  - apply proximity search
- Run the following queries in the following file, and report the list of retrieved documents for each query. Discuss your results with your colleagues and try to find where is the problem if there are difference in answers.
- Think of the following:
  - If the index is very large, what would be the best way to save the index, and how it should be loaded into memory for search later?
  - What if the number of document is very large, and terms appears in documents 100000001 to 10010000. Is there a more effecient way to save document numbers? (read on Delta Encoding) .