**Index building** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

In this assignment I built two versions of the index for a simple statistical retrieval system and also each version of the index shall be in uncompressed form and compressed form A copy of the publicly available Cranfield collection in Cranfield folder.

I built two versions of the index, namely: Index\_Version1.uncompress and Index\_Version2.uncompress of the index. Version 1 of your index considers the terms in the dictionary to be lemmas of words, whereas version 2 of your index considers that the terms of the dictionary are stems of the tokens.

I did not store stop-words in any version of your index. Before building the dictionaries of any of the two versions of index, as it is recommended to remove the stop words.

The terms in my 2nd version of the index should be stemmed with the Porter stemmer.

For every term that in each of the versions of the index, I stored: - Document frequency ( df ): The number of documents that the term occurs in; you should store the df in the dictionary of the index. - the posting file, which contains: - the list of documents containing the term, and along with the document id, also stored: - the Term frequency (tf): e.g. the number of times that the term occurs in each document.

For each document, I stored the frequency of the most frequent stem in that document (max\_tf), and the total number of word occurrences in the document (doclen). To be noted that the value of doclen includes the number of stop-words encountered in the respective documents.

I compressed both versions of your index, generating a Index\_Version1.compressed and Index\_Version2.compressed. To do so, I (a) compress the dictionary in each version of the index and compress the inverted lists before storing them. To compress the dictionaries, I used front coding as well as blocking with a block of size 8. To compress the posting files, I used the gamma code for the document-id compression and to use delta encoding for the frequency information.

Delta codes are similar to the gamma codes: they represent a gap by a pair: (length, offset). First the number is represented in binary code. The length of the binary representation is encoded in gamma code, prior to removing the leading 1-bit. After generating the code of the length only, the leading 1-bit is removed and represented in gamma code.

The program, and the following statistics: - the elapsed time ("wall-clock time") required to build any version of your index, - the size of the index Version 1 uncompressed (in bytes), - the size of the index Version 2 uncompressed (in bytes), - the size of the index Version 1 compressed (in bytes), - the size of the index Version 2 compressed (in bytes), - the number of inverted lists in each version of the index, and - the df, tf, and inverted list length (in bytes) for the terms: "Reynolds", "NASA", "Prandtl", "flow", "pressure", "boundary", "shock" (or stems that correspond to them) are assessed.