Dear Students,

Phase 1 of your project sets up the necessary data structures for future experimentation. Your code must compile on the CommandLine by using oracle jdk, javac compiler using the command: javac \*.java.  This implies that ALL of your java files are located in the same place. So, when submitting, DO NOT submit files in different directories (which code editors tend to do). Likewise COMMENT out the "package" command from the first line of each .java file submitted. (Many editors insert such a package command. You must comment out because CommandLine compilation tends to ignore it and this could cause errors otherwise.)

For consistency across your projects, call the primary (main) class of your project as SearchEngine and the name of its file, SearchEngine.java This file will remain so named for Phase 2 and 3 as well. The CommandLine to run your program will be: (note: CommandLine flags can be listed in any order; your code should be able to handle that)

java **SearchEngine** -CorpusDir **PathOfDir** -InvertedIndex **NameOfIIndexFile** -StopList **NameOfStopListFile** -Queries **QueryFile** -Results **ResultsFile**

where **PathOfDir** is the user’s choice of where the Corpus is installed and where the InvertedIndex can be outputted to; **NameOfIIndexFile** is the name of a text file where the InvertedIndex can be outputted to; **NameOfStopListFile** is the name of a text file containing the stoplist; **QueryFile** will contain the queries for the inverted index and the outputted results to these queries will be written to the **ResultsFile**.

How to submit these files will be further discussed next. Since the files may be large, for this phase ONLY, you can email from your nonCuny email (that you have told me about) to the Class Gmail. Because the outputs of some of the steps are large, it will be required to email the project in four emails, now described.

**1)** You should have already downloaded all the necessary html (data) files for your corpuss stored in directory **PathOfDir**. Again, you were to choose 10 neutral (non-controversial queries) and for each of these, download the first 20 noncontroversial (webpage) results returned by the search engine that are in *html* format. (This was all explained in the email that had the attached document sent 2-3 weeks ago.) Anyway, now also prepare a Word document listing the actual 10 queries and then attach to one email this word document and a compressed file containing all the html results from these queries that you downloaded. The Subject Line of this first email should be:

**Subject: IR Corpus Mar-5-2020**

2) In a Second Email, you will email the stoplist you are using, **NameOfStopListFile**, and also attach a compressed version of the inverted index that your program should output to a text file **NameOfIIndexFile**. (You can manually compress it after your project outputted it.) Your system will print out the inverted index into a text file (described soon). You will compress that file and attach as well to the second email. The subject line for this email should be as follows:

**Subject: IR StopList And Index Mar-5-2020**

**3)** Then, In a Third Email for the actual project code submission, you should email all .java files (properly commented) attached to one Qmail to Class Gmail. You should also include a Readme.txt (and/or UserManual) file which explains how to compile your code from the CommandLine using the javac compiler (again all java files should be assumed to be in the same directory) and run your code using the java runtime command (explained above at beginning of this document.) The subject line for this email should be as follows:

**Subject: IR Phase1 Code Mar-5-2020**

4) A fourth email involves searches on the *inverted index* (as described in the original email requirements documents.) On the CommandLine, you will specify a text file **QueryFile** that holds queries that will be conducted on the Inverted Index. The information of all those queries will as well be outputted to a text file **ResultsFile**. There will be two types of queries: (the keywords to submit queries will be standardized here)

a) Does a specific word appear in any document? The output would be which documents do they appear in. This query would appear in **QueryFile** simply as:

**Query <Term>** where <Term> without the “< >” is the word you are seeking. You should search the inverted index of your Corpus and output the query itself (i.e. Query <Term>) following by the results to text file **ResultsFile**.

b) How many times (frequency) does a word appear in a given document? This query would appear in **QueryFile** simply as:

**Frequency <Term>** where <Term> is a in a) and output the query itself following by the output to text file **ResultsFile**. Here too you should use the information provided by the InvertedIndex.

The code must satisfy all five requirements listed in the originally emailed document (3 main and 2 ancillary):

Main#1: is the corpus above (compressed version with a document listing the original queries used to obtain the corpus documents, emailed in the first email for this project.)

Main#2: is identifying a stoplist (this file is to be submitted in the second email for this project.)

Main#3: is building an inverted index of the corpus (a compressed outputted version also attached to the second email). Your code should print out the inverted index into a text file. The file name should be "passed" on the command line as a parameter to a flag.

Ancillary#1: Command Line Parsing. Specified at beginning of this document.

Ancillary#2: Searching capabilities on the inverted index (see original email file for what information can be expected.)

Sincerely, Professor