CMSC 476/676 Information Retrieval
Homework 1 - inidividually or in pairs

Phase 1 due 2/19/2021 @ 11:59 pm. Extra credit if you turn it in by 2/12/2021.

The objective of this assignment is to compare two approaches to tokenize and downcase all words in a collection of HTML documents. You may choose any of the following approaches: flex, javacc, other publicly available tokenizer, or custom code in C, C++, Perl, Python, PHP, or Java. Each program should read a directory name for the input documents from the command line and a directory name for the output documents from the command line. The program should produce three things:

a directory of all tokenized documents (one output file per input file)
a file of all tokens and their frequencies sorted by token
a file of all tokens and their frequencies sorted by frequency

You may use the UNIX sort facility to sort the output files. However, there must be a single command line call to your function, e.g.,
tokenize input-dir output-dir

Program Testing
The set of files to be preprocessed is available in this compressed tarfile or this files directory. For initial testing, copy a few of these files into your home directory for processing. For final testing, use the full path to these files as the input and your own path for the output to conserve disk space. There is about 12 megs of data, and managing data within your quota is your business. You are free to store the files on your own machine.

Program Documentation.
After your internal documentation (comments) are complete, write a report that provides a short executive summary of your programs. In particular, discuss how you handled punctuation and numbers, and describe how you calculated the frequency of each word. Identify some HTML constructs or words which are incorrectly tokenized (if any) and discuss why your program does not handle them properly. Also, discuss the efficiency of your frequency program in terms of order of magnitude and timings (cpu time, elapsed time). Include a small graph or table of time versus number of documents processed. The entire document should be no more than four pages in length. You should discuss the differences between the results and the efficiency between the two versions of the solution.

We will primarily be grading from the report, so make sure it clearly describes what you did and your program's output and efficiency. You may work alone, but do get the results (timings and tokens for the complete set of test files) from a classmate so you can compare your results with theirs. **You must compare the two approaches, even if they were developed independently.** Also, discuss which tokenizer produces the better output, and why.

You may work with your partner on implementing these programs, or you can implement this on your own. You will need to share results with your partner so that you can have comparisons to make in your report.

Hand In
Hardcopy of your code (including shell scripts), the report, and the first 50 and last 50 lines of the two frequency files.

Late Policy
10% deduction per 24 hours. Assignments turned in during or after the deadline will result in a 10% deduction.