R·I·T	Rochester Institute of Technology Golisano College of Computing and Information Sciences Department of Information Sciences and Technology

Lab 1 (8 points)
Inverted Index Construction and Query Processing

Overview

This lab consists of three major tasks:

- Processing a number of text documents to generate indexed terms.
- Building an inverted index for the terms and documents, and
- Formulate and process queries using the constructed inverted index.

Note: Make JavaDoc comments in your Java programs including Course #, Lab #, Your name, and main functional description of each method with @param & @return if applicable at the minimum.

Ref. http://www.oracle.com/technetwork/articles/java/index-137868.html

Resources

- You should have read Chapters 1 and 2 of Introduction to Information Retrieval.
- Carefully read the lecture examples of weeks 1 and 2 and understand the technical details.
- Go over the lecture notes of weeks 1 and 2.

Task 1: Inverted Index Construction (3 points)

In this task, you need to read in the text documents located in the lab1_data folder. There are five files in total in the folder. You need to create an inverted index by performing the following subtasks:

1. Assign a unique id to each text document, i.e., 1-5.

2. Read in the text in each document and perform tokenization. Treat punctuation (e.g., “. & % $ # ! /”), symbols (e.g., “+-*/”), and spaces as delimiters.

3. Adopt a proper data structure to store the given stop word list and use it to efficiently remove all the stop words in the documents.

4. Call Porter’s stemmer to perform stemming. (Covered in Week #3)

5. All the remaining tokens will be treated as terms in the dictionary. Documents that contain the term should appear in the postings list of the term.

Task 2: Query processing (5 points)

In this task, you need to implement search algorithms that use the constructed inverted index to perform the following queries.

1. Implement a search algorithm that can handle a query with a single keyword. Design two test cases and show the document names as the search result.

2. Implement a search algorithm that can handle a query with two keywords. Assume that query terms are connected using the AND operator. As an example, a query "information technology" means "information AND technology". Design two test cases and show the document names as the search result.

3. Implement a search algorithm that can handle a query with two keywords. Assume that query terms are connected using the OR operator. As an example, a query "information technology" means "information OR technology". Design two test cases and show the document names as the search result.

4. Implement a search algorithm that can handle a query with three or more keywords. Assume that query terms are connected using the AND operator. As an example, a query "Rochester Institute Technology" means "Rochester AND Institute AND Technology". Design two test cases and show the document names as the search result. The query should be optimized so that shorter postings lists will be processed first. It is also necessary to show the order in which these keywords are combined. Using the same example, if your algorithm process the keywords in the order of "Rochester", "Technology", and then "Institute", it should be shown as
    a. 1. Rochester
    b. 2. Technology
    c. 3. Institute

Submit your programs to a lab drop box in MyCourses before meeting with Instructor/TA.