SE375 - Laboratory Assignment 04

Part 1: Word Count with Thread Pools and Thread-safe Collections

You are now very familiar with the Word Count application. In your previous laboratory assignment you have used threads with a shared data structure. Your task for this assignment is to use a thread-safe collection that will be shared between threads that operate on files. Instead of creating a new thread for each file, you will also have to assign one from a thread pool of size 10. You should use the Java ExecutorService to do so.

You should follow these steps in your Java applications: (1) For each file passed as a parameter, you will assign a thread from the thread pool to parse the file and a shared thread-safe data structure to save the results. (2) Each file reads the file assigned to it, and uses the shared thread-safe data structure to add or update the words. (3) Once all threads are complete, ask the user for a word they are looking for, and will print the number of appearances of that word.

Try your application with 4, 8, and 12 different files. You can make copies of existing files by renaming them.

Part 2: Inverted Index with Threads and Shared Data Structure

Repeat the same approach for the Inverted Index. This time, you are asked to find the files the word appears in. Use one thread per file from the thread pool with a shared thread-safe data structure passed as argument to each thread. Once the threads are complete, let the user query the data structure.

Try your application with 4, 8, and 12 different files, as you did with the Word Count application.