Pratap Luitel
CS 50, Indexer

Introduction

Indexer is second phase module for the Tiny Search Engine. The purpose of the indexer is to output a file with a list of all the words with their frequencies in various document files saved by the crawler. This is achieved by iterating through each word in all the files while keeping track of the frequency and document name.

The indexer can be called by two different argument choices.
Normal mode: ./indexer directory filename
Testing mode: ./indexer directory filename1 filename2 filename3

The speciality about the testing mode is that it parses and recreates the data structure created from the inverted index in the normal mode.

Data Structure

Hashtable stores WordNode.

WordNode: ptr to another word node, word, ptr to a documentNode

Wordnode has a link to DocumentNode.

DocumentNode: ptr to another document node, document id, frequency

Data Flow

Check input arguments
Initialize data structures
Filenames from the valid directory are saved using GetFilenamesInDir()
Loop through each file, load the content to buffer.
Get the words from buffer, normalize.
Make WordNode and DocumentNode corresponding to each word.
Write a file with inverted index at the end

Test

Thorough testing conditions are implemented in BATS.sh. Some of them are as follows.

Test Conditions

Input Argument Size: 2 (smaller)
Input Argument Size: 7 (bigger)
Incorrect directory path
Incorrect filename
Argument 3 and 4 not same
Argument 4 and 5 same

The result of the BATS.sh is saved in a log file named after the date stamp.