



Lab 3

Data structures (Btree and simple search engine)

Elsayed Akram Elsayed 16

Hassan Abdelhameed Hassan 25

Seif Ehab Mostafa 33

Abdelrahman Adel Abdelfattah 37

Simple search engine

Simple search engine in this lab is an application of BTree and its use in indexing. BTree is used to store every word of a doc. For every document, there is an ID and a paragraph. There is a BTree corresponding to each ID. For every file, there are several IDs.

That's why there are two lists parallel to each other :

1.paths: where the path of each file is stored.

2.files:where for each path in paths list there is a corresponding hashmap(indexes) at the same index of the path.

The hashmap indexes maps each ID to a BTree containing the words in the document having this ID.

Methods :

1.Index webpage : the file is read using a method called ReadFile, then we do some operations through which we have a list of strings having the words of each document.This list of strings is inserted into BTree then this BTree is added to the hashmap with its corresponding ID then we do the same through each document, then we add this hashmap to files list and at the same index the path of file is added to paths list.

2.Index webDirectory: same as Index webPage but the only difference is there are more files ,so we just loop these files.

3.Delete webPage: removes the file with the given path where it removes the path and the hashmap of this file from paths and files lists.

4.SearchByWordWithRanking : loops each file and loops through every document in this file then performs a traditional BTree search to the words of this document.

5.SearchByMultipleWordWithRanking : divides the string into multiple words then performs SearchByWordWithRanking method for each word then adds each word's search results to a list which is to be returned.

Time and Space complexity:

1.Space complexity: $O(n)$

2.Time complexity:

1.BTree:

Search,delete,insert: $O(\log n)$

2.Simple search engine:

- Index webPage and WebDirectory: $O(n \log n)$.n:total number of inserted words
- Delete webPage: $O(n)$ because it searches for the path in a list
- SearchByWordWithRanking and SearchByMultipleWordWithRanking: $O(n \log n)$.n: total number of documents (BTrees) searched