

Name: Muzaffer Sevil

Number: 2019510069

Firstly, in this homework, we are expected to implement a record-level inverted index structure using our own hash table implementation in Java program. Our index structure is able to find all documents that contain a particular words.

In my program, hash table is type of ArrayList and takes Nodes <Key, Value>, key for words and value for files and Value also takes Nodes <Key, Value>, key for files and value for number of words in this file. My program have 4 class which are HashTable, Node, FolderOperations and Main and 1 interface which is HashInterface.

HashTable; the class is for build the hash table and it's main functions which are put(Key k, Value), get(Key k), remove(Key k) and resize that resizes according to load factor. This class also contains Hash Functions; Simple Summation Function(SSF) and Polynomial Accumulation Function(PAF). These functions are using for indexing. In this class, also there are two function to handle collision; Linear Probing(LP) and Double Hashing (DH).

Node; the class takes Key and Value to place the data to hash table.

FolderOperations; this class contains two function which are named Read_file that reads content of the files and Read_lines that reads the file line by line. Also try-catch is used in it.

Main; the most important class for the all other operations. In this class, the bbc files are read, splitted and placed to the ArrayList in nodes according to DELIMITERS.txt file. While reading the stopwords which are in the stopwords.txt file are ignored. During the adding checked whether word or file has already been added. Here also there is searching the words that we want to see which files and how many contains this word. Also input is taken here.

Finally, when the coding has finished I tried all possible combinations according to load_factor (%50, %80), SSF, PAF, LP, DH and I see the performance of my program.

Load Factor	Hash Function	Collision Handling	Collision Count	Indexing Time	Avg. Search Time	Min. Search Time	Max. Search Time
$\alpha=50\%$	SSF	LP	2452081239	27184 ms	17	16	19
		DH	436547	38923 ms	16.2	16	17
	PAF	LP	41545	3886 ms	17	16	18
		DH	30317	3906 ms	22	21	24
$\alpha=80\%$	SSF	LP	270896612	28780 ms	15	14	16
		DH	453184	37051 ms	21	18	22
	PAF	LP	118963	3866 ms	17	15	20
		DH	40493	3965 ms	16	15	18