DOKUZ EYLUL UNIVERSITY

ENGINEERING FACULTY

DEPARTMENT OF COMPUTER ENGINEERING

CME 2201 - Assignment 1

INVERTED INDEX BY USING HASH TABLES

by

2017510063 Arif Mertaslan

Lecturers

Prof.Dr. Zerrin Işık

Res.Asst. Altuğ Yiğit

Res.Asst. Ali Cüvitoğlu

Res.Asst. Feriştah Dalkılıç

IZMIR
15.11.2019

Progress Description

In this assignment, a record-level inverted index structure implemented by using our own hash table implementation in Java programming language. Index structure will be used to find all documents that contain a particular word (e.g., return all documents in which "computer" occurs).

Completed Tasks

Simple Summation Function and Polynomial Accumulation Function completed. Linear Probing and Double Hashing collusion handling techniques implemented and both implementation completed by 50% and 80% load factors.

Performance Monitoring

Load Factor	Hash Function	Collision Handling	Collision Count	Indexing Time (ms)	Avg. Search Time (ns)	Min. Search Time (ns)	Max. Search Time (ns)
α=50%	SSF	LP	560263816	93609	558053	800	4492500
		DH	206083578	48706	350410	1200	4818000
	PAF	LP	43831	7490	2929	600	139000
		DH	41319	7027	2752	600	112600
α=80%	SSF	LP	853821293	95513	564278	900	10613300
		DH	314425219	49849	299760	900	3380500
	PAF	LP	180566	7522	4022	600	796000
		DH	129061	7353	3065	700	111800

Table 1. *Performance matrix*

As in the table, the quickest indexing time is 4th case which is 50% load factor, Polynomial Accumulation Function (PAF) and Double Hashing collusion handling techniques. The number of collisions also has the lowest value in the same case. The slowest indexing time is 5th case which is 80% load factor, Simple Summation Function (SSF) and Linear Probing collusion handling techniques. The number of collisions also has the highest value in the same case.

At the same load factor and the same collusion handling techniques, SSF causes bigger collusion counter than the PAF. So, SSF function usage is slower than the PAF function (1st and 3rd cases in table).

At the same load factor and the same key function (SSF or PAF), The Double Hashing techniques work quicker than the Linear Probing techniques because of the value of the collusion count (3rd and 4th cases in table).

At the same key function (SSF or PAF) and the same collusion handling techniques, 50% load factor is faster than 80% load factor because of the collusion counter(1st and 5th cases in table).

At the same key function (SSF or PAF) and the same collusion handling techniques, 80% load factor causes more collusion counter than 50% load factor. So, 50% load factor is faster than the 80% load factor (4th and 8th cases in table).