COSC 2406 Database Systems Assignment #1 – S3163776 – jlanders – semester 1 2009

5.1 Equality searching with a heap file

		wheap		
Run #		100	1000	10000
	1	119680	126820	142410
	2	118170	130150	142110
	3	121770	128510	142060
	4	124600	129920	145050
	5	117680	129970	154430
	6	125530	128040	141500
	7	123260	123080	154900
	8	120070	133150	139890
	9	123510	131350	144390
	10	124600	133980	140650
	11	126160	131560	141590
	12	128120	133450	142470
	13	127720	133650	142610
	14	125280	128370	142450
	15	129700	133650	141980
	16	126310	129120	141740
	17	126880	129920	142460
	18	123890	130120	142680
	19	123500	131540	139450
	20	124500	132450	140810
mean		124046.5	130440	143281.5
standard deviation		3249.02	2720.4	4100.17

5.2 Equality searching with a hashed file

	whash		
Run #	100	1000	10000
1	40	470	4990
2	60	490	4930
3	70	490	4810
4	40	450	4820
5	60	430	4930
6	50	470	4880
7	40	450	4870
8	50	500	4890
9	50	440	4880
10	50	450	4860
11	20	460	4920
12	50	450	4840
13	60	430	4890
14	50	470	4820
15	40	480	4820
16	40	460	4910
17	50	490	5080
18	60	460	4880
19	60	440	4860
20	60	480	4930
mean	50	463	4890.5
standard devi	11.24	20.8	64.19

COSC 2406 Database Systems Assignment #1 – S3163776 – jlanders – semester 1 2009

5.3 Comparison of approaches

Discussion of heap and hash file implementations and their respective advantages and disadvantages:

We can see from the results that equality searching a hash file is much quicker than a heap file by quite a large factor.

This is because the hash file is not only unsorted but may require us to traverse (in this case) up to 400,000 records on each search whereas the heap file may search up to 125, 1250, 12500 records before a match is found depending on the page size and occupancy level.

The heap file is faster because by placing records in a "bucket" we limit how many records we need to search through. The heap file does this by hashing the unique key of the record and placing it into a free space in the bucket the hash maps to.

The hash file is smaller because it only writes the records that exist – occupancy is 100% and it does not write any additional data. Whereas the heap file is larger because it has 80% occupancy per bucket and it has to write headers to the start of each bucket to indicate how many records are in that bucket and also to the start of the file to indicate how many buckets exist.

Page sizes
The trends do change for different page sizes:
With hashing the search time increases by a factor of 10 as the number of records in a page(remember bucket is larger) increase by that same factor.

The average search is lower for larger pages when equality searching a heap file. This could be because we are reading blocks in order and the OS/HDD is pre-caching the next blocks.

Are these the results you would have expected to see based on your understanding of file organisation?
The results are what I expected to see as per my explanation of the file structures and access methods given above.

COSC 2406 Database Systems Assignment #1 – S3163776 – jlanders – semester 1 2009

5.4 Theory

1)

If we carried out range searches we would expect the results to change.

The heap file would be more preferred because it would be faster. We would just seek through the file and see if each record falls into the range.

The lecture notes state the cost of this search would be BD.

The hash file would be slower than the heap file because it's not practical to use the hashing function to do range searches. Therefore we would need to scan the file. Hash files are larger, in this case by slightly over 25% (occupancy is 80% + some counter overheads) leading to more disk reads.

The lecture notes state for a scan that cost of a search would be 1.25BD.

B = data blocks or pages
D = average time to transfer a disk block

2)

Simple assumptions in this assignment with regards to file organisation are that the buckets are always big enough to hold all records mapping to a particular hash.

In practice we would need to have an overflow bucket or another file.