CMPT 454 (Spring 2010)
Assignment 2: Data Storage and Query Answering

Due: Monday, 2010-03-01, 11:29AM (at the beginning of class)

Instructions on Assignment Submission: Hard copy only!

- Option 1: drop off your assignments in the assignment box (with label **CMPT 454**) in CSIL.
- Option 2: bring your assignments to the class on the due day, and the instructor will collect the assignments **at the beginning of class**.

Please write legibly or typeset your answers using your favorite word processor. Late assignments will not be accepted unless there is a documented medical reason.

Problem 1.

(25 points) [Disk]

Consider a disk with a sector size of 512 bytes, 2000 tracks per surface, 50 sectors per track, 5 double-sided platters, average seek time of 10 milliseconds.

(1) What is the capacity of a track in bytes? What is the capacity of each surface? What is the capacity of the disk?

(2) How many cylinders does the disk have?

(3) Give examples of valid block sizes. Is 256 bytes a valid block size? 2048? 51200?

(4) If the disk platters rotate at 5400 rpm (revolutions per minute), what is the maximum rotational delay?

(5) Assuming that one track of data can be transferred per revolution, what is the transfer rate?

For the following sub-questions, suppose that a block size of 1024 bytes is chosen. Suppose that a file containing 100,000 records of 100 bytes each is to be stored on such a disk and that no record is allowed to span two blocks,

(6) how many records fit onto a block? How many blocks are required to store the entire file? If the file is arranged sequentially on disk, how many surfaces are needed?

(7) How many records of 100 bytes each can be stored using this disk?

(8) What is the time required to read a file containing 100,000 records of 100 bytes each sequentially?

(9) What is the time required to read a file containing 100,000 records of 100 bytes each in some random order?

Problem 2.

(25 points) [Elevator Algorithm]

Suppose that we are scheduling I/O requests for one type of commercial disks. For simplicity, we assume that the seek time for the header to move across n cylinders is calculated as $1 + \frac{n}{4000}$

(milliseconds). The sum of average latency and transfer time is 4.3 milliseconds. Initially the head is at cylinder 32000, and then the following requests come in:

- time = 0; request for block on cylinder 8000 arrives;
- time = 1; request for block on cylinder 48000 arrives;
- time = 10; request for block on cylinder 4000 arrives;
- time = 20; request for block on cylinder 40000 arrives;

(1) If we use the elevator scheduling algorithm, at what time is each request serviced completely?

(2) If we use a first-come-first-served scheduler, at what time is each request serviced fully?

Problem 3.

(25 points) [B+-tree]

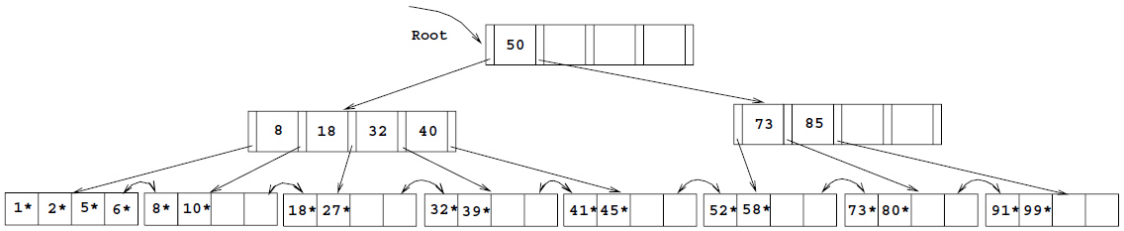

Figure 1: Problem 3.

Consider the B+ tree index ($n = 4$) shown in Figure 1.

(1) Show the B+ tree that would result from inserting a data entry with key 3 into the original tree.

(2) Show the B+ tree that would result from deleting the data entry with key 8 from the original tree, assuming that the left sibling is checked for possible redistribution.

(3) Show the B+ tree that would result from deleting the data entry with key 8 from the original tree, assuming that the right sibling is checked for possible redistribution.

(4) Show the B+ tree that would result from successively deleting the data entries with keys 32, 39, 41, 45, and 73 from the original tree.

Problem 4.

25 points) [Hash Table]

Consider the Extensible Hash table in Figure 2 with a block capacity of 2.

(1) Show the Extensible Hash table after the insertion of search keys with the following hash values: 0010, 0110, 1001.

(2) Give a hash key whose insertion (into the above hash table without the insertions from (1)) would require the creation of an overflow block. Show the resulting hash table.

(3) Give a hash key whose deletion (from the above hash table without the insertions from (1)) would lead to a halving of the directory. Show the resulting hash table.

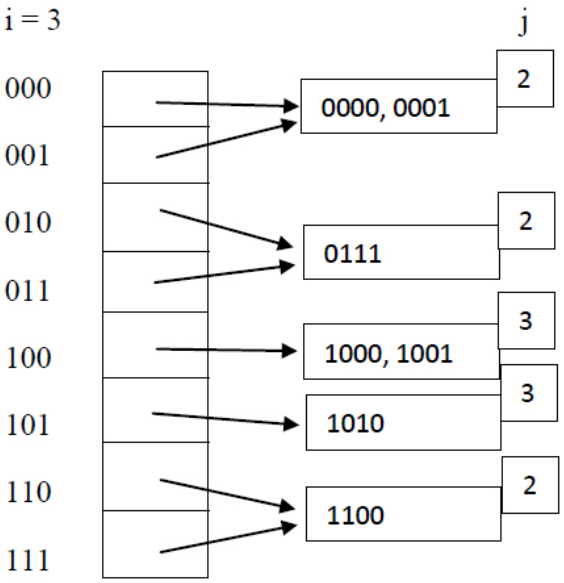

Figure 2: Problem 4.