CS4432: Database Systems II
Spring D-Term
Homework 2

Release Date: March 30, 2014

Due Date: April 6, 2014 (11:59PM)

Total Points: 100

Problem 1 (Buffer Manager) [40 Points (20 each)]

Assume we have a Buffer Manager managing a buffer pool of 7 frames, i.e., the pool has 7 entries.

(a) LRU Replacement Policy: Assume the following requested pattern of the disk pages.

1*, 2*, 3*, 1, 4*, 5*, 3, 4, 1, 6, 7, 8*, 9*, 5, 10

where: X* means request page X and pin the page in memory

X means unpin page X (but it should stay in memory until the replacement policy takes it out)

X (without * or underline) means accessing the page without changing its pin status. If X does not exist in memory, it will be retrieved without pinning it.

Describe the state of the buffer pool after each of the requests in the above pattern. That is, after each request, list the 7 frames and indicate each frame contains which page, and whether or not the page is pinned.

(b) Clock Replacement Policy: Assume the same requested pattern of the disk pages.

1*, 2*, 3*, 1, 4*, 5*, 3, 4, 1, 6, 7, 8*, 9*, 5, 10

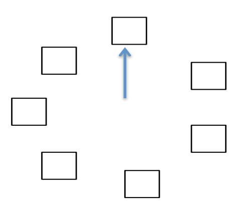
The only difference is that now each frame has a pin flag and also a Ref bit (0 or 1).

And: X* means request page X and pin the page in memory. Also set Ref bit = 1.

X means unpin page X (but it should stay in memory until the replacement policy takes it out). The Ref bit remains the same.

X (without * or underline) means accessing the page without changing its pin status. If X does not exist in memory, it will be retrieved without pinning it. Also set Ref bit = 1.

Imagine the 7 frames are now forming a circle and we have the pointer of the Clock Replacement Policy pointing to the 1st frame as shown in the figure. If the request is in memory, the pointer does not move. If the request is not in memory, the pointer moves clock-wise to find a frame to use (either empty one or one to replace according to the Clock Policy covered in class).



Describe the state of the buffer pool after each of the requests in the above pattern. That is, after each request, list the 7 frames (in a circle shape as shown in the figure) and indicate each frame contains which page, the pin flag, and the Ref bit.

Problem 2 (Indexing) [30 Points (5 each)]

Consider a data file R consisting of 1,000,000 blocks that are contiguous on disk. Each block contains 20 fixed-size records. Let K1 correspond to the primary key of the relation, and that the data file R is sorted by K1. Also, let K2 be another attribute of R. Let values of K1 and K2 be 20 bytes each, a record pointer is 8 bytes long, and a block is 8KB. For the below, assume no spanning of records across blocks is allowed.

(1) Is it possible to construct a dense sequential index (1-level) on K1 over R? Describe the layout, and how large (how many blocks) will the index be?

(2) Is it possible to construct a sparse sequential index (1-level) on K1 over R? Describe the layout, and how large (how many blocks) will the index be?

(3) Is it possible to construct a dense sequential index (1-level) on K2 over R? Describe the layout, and how large (how many blocks) will the index be?

(4) Is it possible to construct a sparse sequential index (1-level) on K2 over R? Describe the layout, and how large (how many blocks) will the index be?

(5) Is it possible to build a second-level index on the one built in (1)? If yes, what will be the size of the index (how many blocks)? Report the size of the second-level alone and the total index size (both levels).

(6) Is it possible to build a second-level index on the one built in (2)? If yes, what will be the size of the index (how many blocks)? Report the size of the second-level alone and the total index size (both levels).

Problem 3 (B+Tree Indexing) [30 Points (5 each)]

Consider the B+ tree in Figure 14.13 from the course textbook. Describe how each of the following operations would proceed. If it modifies the tree, show the revised tree. Assume that an insert operation does not do any re-organization, rather it only does splitting. You can make additional assumptions, but you must always spell them out.

1. Lookup record with search key 35. Indicate which index pages are accessed? Write them in the order of their access.

2. Lookup all records with search key in the range [9, 21]. Indicate which index pages are accessed? Write them in the order of their access.

3. Insert a record with key 4. Show the modified tree.

4. To the tree after the above insertion, insert record with key 14, then record with key 15, then record with key 16. Show the tree after every insert.

5. Lookup records with search keys in the range [6, 13]. Indicate which index pages are accessed? Write them in the order of their access.

6. Delete the record with key 23. Show the modified tree.

What to Submit

- Include your answers in one file (.doc, .docx, or .pdf). This is the only file to submit.
- Include your name inside the file.

Where to Submit

- In WPI blackboard system

Late Submission Policy

- Follows the policy posted on the course website.