CSci 4707
Homework 3 Solution
Fall 2015

Chapter 8 and 12
Due Tuesday, 11/17/2015 13:00

B1a.

A	F	G	H	I

B1b. If we assume the newly inserted buffer will not have 0 pin count

A	G	H	F	I

Otherwise,

A	B	C	I	E

B1c.

A	H	G	I	F

B1d.

A	I	F	G	H

B2. (1) DB may need to force a page before writing another page. **(2)** DB has more context and can do smarter page replacement for its own need.

B3a. MRU to avoid sequential flooding and better hit rate than FIFO for sequential scan.

B3b. LRU or Clock (trivial)

B4a. When the resulting records are within the same page or in a small number of pages.

B4b. No, this is a contradiction. Alternative 1 will always be a clustered B+Tree since Alternative 1 makes all data entries sorted.

B4c. One of the difference is: B+Tree has all its Data Entries on the leaf node while B-Tree can be in any node. **(this problem is omitted since we don't learn about B-Tree in class)**

C1a. For hash index, the search term must match the composite keys completely.
i. No, ii. No, iii. Yes

C1b. For tree index, the search term must match the prefix of the composite key of the tree.
i. No, ii. Yes

C2a. Scan. Most age will be above 10 so using unclustered B+Tree will result in worse performance.

C2b. Clustered B+Tree. Hash will not be good since it is not a clustered hash index and we need to retrieve 90% of total tuples.

C2c. Scan + check the retrieved tuples' category.

C2d. Clustered B+Tree on <state, age> + check the retrieved tuples' category.

C3. We can use the Clustered B+Tree and use an **index only scan** to get the average age for each states.

C4a. Assume: 2 I/Os for probing the index: 100 + 1000 * (2 + 1) = 3100 I/Os

C4b. Assume: 1.2 I/Os for probing the index: 10 + 5000 * (1.2 + 1) = 11010 I/Os

C4c. 100 + 1000 * ((5000/1000) + 2) = 7100 I/Os

C4d. 10 + 5000 * 100 = 500010 I/Os