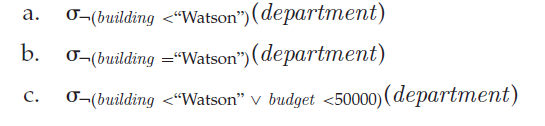
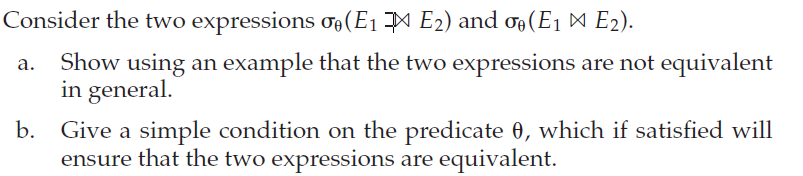


2. Suppose that a B+-tree index on *building* is available on relation *department*, and that no other index is available. What would be the best way to handle the following selections that involve negation?





4. Suppose you need to sort a relation of 40 gigabytes,with 4 kilobyte blocks,using a memory size of 40 megabytes. Suppose the cost of a seek is 5milliseconds, while the disk transfer rate is 40 megabytes per second.

a. Find the cost of sorting the relation, in seconds, with *bb* = 1 and with *bb* = 100.

b. In each case, how many merge passes are required?

c. Suppose a flash storage device is used instead of a disk, and it has a seek time of 1 microsecond, and a transfer rate of 40 megabytes per second. Recompute the cost of sorting the relation, in seconds, with *bb* = 1 and with *bb* = 100, in this setting.

5. Consider the bank database

*branch*(*branch name*, *branch city, assets*)

*customer* (*customer name*, *customer street, customer city*)

*loan* (*loan number*, *branch name, amount*)

*borrower* (*customer name*, *loan number*)

*account* (*account number*, *branch name, balance*)

*depositor* (*customer name*, *account number*)

where the primary keys are underlined.

Construct the following SQL queries for this relational database

1. Write a nested query on the relation *account* to find, for each branch with name starting with B, all accounts with the maximum balance at the branch.
2. Rewrite the preceding query, without using a nested subquery; in other words, decorrelate the query.
3. Give a procedure for decorrelating such queries.