INF3100: Databasesystemer – Oppgavesett 7

Oppgave 6.3.1: Write the following queries, based on the database schema

```
Product(maker,model, type)
PC(model, speed, ram, hd, price)
Laptop(model, speed, ram, hd, screen, price)
Printer(model, color, type, price)
```

of Exercise 2.4.1. You should use at least one subquery in each of your answers and write each query in two significantly different ways (e.g., using different sets of the operators EXISTS, IN, ALL, and ANY).

a) Find the makers of laptops with a speed of at least 2.0.

b) Find the printers with the highest price.

c) Find the laptops whose speed is slower than that of the fastest PC.

Oppgave 10.x.1: Følgende er et skjema over studentforeninger:

`Studentforening(forening, verv, person)`

Verv er 'leder', 'nestleder', 'kasserer', 'arrangementssjef', mm. Informasjonen kan tolkes som en graf der hver node representerer en forening eller en person, med kanter som viser hvilke verv de enkelte personene har.

b) Finn ut om grafen inneholder noen sykler som starter med 'Siv Dahl'.

Oppgave 14.1.1: Suppose blocks hold either five records, or 20 key-pointer pairs. As a function of n, the number of records, how many blocks do we need to hold a data file and

a) a dense index,

b) a sparse index?

Oppgave 14.2.1: Suppose that blocks can hold either ten records or 99 keys and 100 pointers. Also assume that the average B-tree node is 70% full; i.e., it will have 69 keys and 70 pointers. We can use B-trees as part of several different structures. For each structure described below, determine

(i) the total number of blocks needed for a 100,000-record file, and

(ii) the average number of disk I/O's to retrieve a record given its search key.

You may assume nothing is in memory initially, and the search key is the primary key for the records.

a) The data file is a sequential file, sorted on the search key, with 20 records per block. The B-tree is a dense index.

b) The same as (a), but the data file consists of records in no particular order, packed 20 to a block.

c) The same as (a), but the B-tree is a sparse index.

d) The data file is a sequential file, and the B-tree is a sparse index, but each primary block of the data file has one overflow block. On average, the primary block is full, and the overflow block is half full. However, records are in no particular order within a primary block and its overflow block.

e) Instead of the B-tree leaves having pointers to data records, the B-tree leaves hold the records themselves. A block can hold ten records, but on average, a leaf block is 70% full; i.e., there are seven records per leaf block.

Oppgave 17.4.1: For each of the sequences of log records representing the actions of one transaction T, tell all the sequences of events that are legal according to the rules of undo/redo logging, where the events of interest are the writing to disk of the blocks containing database elements, and the blocks of the log containing the update and commit records. You may assume that log records are written to disk in the order shown; i.e., it is not possible to write one log record to disk while a previous record is not written to disk.

a) `<START T>; <T,A,10,11>; <T, B, 20,21>; <COMMIT T>;`

b) `<START T>; <T,A, 10,21>; <T, B, 20,21>; <T, C, 30,31>; <COMMIT T>;`

Oppgave 17.4.2: The following is a sequence of undo/redo-log records written by two transactions T and U: `<START U>; <U, A, 10, 11>; <START T>; <T, B , 20, 21>; <U, C, 30, 31>; <T, D, 40,41>; <COMMIT T>; <U, E, 50,51>; <COMMIT U>`. Describe the action of the recovery manager, including changes to both disk and the log, if there is a crash and the last log record to appear on disk is:

a) `<START T>`, b) `<COMMIT T>`, d) `<COMMIT U>`