Ceng352 - Database Management Systems
Written Assignment 3

Spring 2019

Q1 Consider the following database schema:

```
Customer (cid, name, surname, phone, email)
Order (cid, pid, date, quantity)
Product (pid, pname, price)
```

For each of the queries below, show an equivalent relational algebra plan.

(a)
```
SELECT O.quantity, COUNT(DISTINCT C.cid),
FROM Customer C, Order O, Product P
WHERE C.cid = O.cid AND
        O.pid = P.pid AND
        C.email LIKE '%gmail.com' AND
        P.pid = 100
GROUP BY O.quantity
HAVING O.quantity > 10
```

(b)
```
SELECT C.name, C.phone
FROM Customer C
WHERE NOT EXISTS (
        SELECT *
        FROM Product P
        WHERE P.price < 50 AND NOT EXISTS (
                SELECT *
                FROM Order O
                WHERE O.cid = C.cid AND O.pid = P.pid
        )
)
```

Q2 Consider the following relations:

```
Professor (pid, name, dept)
Teaching (pid, code, semester)
```

There are 1000 professors in 100 departments. The size of the relation `Professor` on disk is 200 pages. There are 10000 teaching records for 4 semesters. The size of the relation `Teaching` is 500 pages.

(a) Consider the following query:

```
SELECT *
FROM Teaching T
WHERE T.semester = 'F2017'
```

Estimate the cost of executing this query using

i. Table scan
ii. A clustered B+ tree index on `semester`
iii. An unclustered B+ tree index on `semester`

(b) Consider the following query:

```
SELECT *
FROM Professor P
WHERE P.dept = 'CENG'
```

Estimate the cost of executing this query using

i. Table scan
ii. A clustered hash index on `dept`
iii. An unclustered hash index on `dept`

(c) Consider the following query:

```
SELECT *
FROM Professor P
WHERE P.pid > 200 AND P.pid < 400
```

Estimate the cost of executing this query using

i. Table scan
ii. A clustered B+ tree index on `pid` (assume that `pid` values range in between 0 and 1000)
iii. An uclustered B+ tree index on `pid` (assume that `pid` values range in between 0 and 1000)

(d) Consider the following query:

```
SELECT DISTINCT name
FROM Professor
```

Assuming 10 names fit in one page, estimate the cost of executing this query using

i. Hashing based projection (assume $M = 21$ available memory pages)
ii. Sorting based projection (assume $M = 20$ available memory pages)

Q3 Consider two relations $R(a, b)$ and $S(b, c)$ with the following statistics:

```
T(R) = 10000, B(R) = 1000 (each page contains 10 tuples),
V(R, b) = 200 (number of distinct values of attribute b in R)

T(S) = 6000, B(S) = 1500 (each page contains 4 tuples),
V(S, b) = 150 (number of distinct values of attribute b in S)
```

(a) What is the cost of $\mathbf{R} \bowtie \mathbf{S}$ using **page-at-a-time nested loop join** algorithm?

(b) What is the cost of $\mathbf{R} \bowtie \mathbf{S}$ using **block nested loop join** algorithm assuming that available memory $M = 102$ pages?

(c) What is the cost of $\mathbf{S} \bowtie \mathbf{R}$ using **block nested loop join** algorithm assuming that available memory $M = 102$ pages?

(d) Describe how to join the two relations if **S** has a **i)** clustered **ii)** unclustered index on the join attribute **b**. What is the cost of join using **index-nested loop join** for each type of index?

(e) Describe in detail how to join these two relations using **partitioned hash join** given that 101 pages can fit in main memory at a time. Specially, explain how many pages are used in memory and how. Also specify what exactly is written to disk and when. Compute the cost of the partitioned hash join operation.

(f) Describe in detail how to join these two relations using **merge-join** given that only 101 pages can fit in main memory at a time. Specially, explain how many pages are used in memory and how. Also specify what exactly is written to disk and when. Compute the cost of the merge-join operation.