Ceng352 - Database Management Systems
Written Assignment 2

Spring 2020

Q1 Given a B+ tree index on composite search key (age, grade), explain whether it is possible to evaluate the following queries with an index-only plan:

(a)
```
SELECT S.age, MIN (grade)
FROM Student S
WHERE S.age >= 30
GROUP BY S.age
HAVING COUNT(*) > 2
```

(b)
```
SELECT S.age, MIN (grade)
FROM Student S
WHERE S.age >= 30 AND gender = 'Female'
GROUP BY S.age
```

Q2 Consider a relation R(A, B, C, D, E) containing 10,000,000 records, where each data page of the relation holds 10 records. Assume that R.A is a candidate key for R, with values lying in the range 1 to 10,000,000. For each of the following relational algebra queries, state which of the following methods for accessing R is likely to have the least cost. If more than one method seems nearly equivalent in cost for a query, mention both methods.

- Use a heap file (i.e. an unsorted file) storing relation R.
- Use an unclustered B+ tree index on attribute R.A.
- Use a (clustered) B+ tree index on attribute R.A
- Use a hash index on attribute R.A.

(a) $\sigma_{A=500.000}$ (R)

(b) $\sigma_{A<20.000}$ (R)

(c) $\sigma_{A>20.000}$ (R) $\wedge$ $\sigma_{A<20.010}$ (R)

(d) $\sigma_{A\neq500.000}$ (R)

Q3 Consider the following database schema:

```
Actor(aid, name, surname, age)
Movie(mid, title, year, genre)
Acts(aid, mid, salary)
```

For each of the queries below, show an equivalent logical query plan.

(a)

```
SELECT Ac.Salary, Count(DISTINCT A.aid)
FROM Actor A, Movie M, Acts Ac
WHERE A.aid = Ac.aid AND M.mid = Ac.mid AND
          M.genre = 'Horror' And A.age < 50
GROUP BY Ac.Salary
HAVING Ac.Salary > 50000
```

(b)

```
SELECT *
FROM Actor A
WHERE NOT EXISTS (
    (SELECT M.mid
    FROM Movie M
    WHERE M.title LIKE 'Godfather' AND M.year < 1980)
    EXCEPT
    (SELECT Ac.mid
    FROM Acts Ac
    WHERE Ac.aid = A.aid))
```

Q4 Consider the join R $\underset{R.a=S.b}{\bowtie}$ S, given the following information about the relations to be joined. The cost metric is the number of page I/Os unless otherwise noted, and the cost of writing out the result should be uniformly ignored.

- Relation R contains 20.000 tuples and has 10 tuples per page.
- Relation S contains 5.000 tuples and also has 10 tuples per page.
- Attribute S.b is the primary key for S.
- Both relations are stored as simple heap files and neither relation has any indexes.
- 42 buffer pages are available in memory.

(a) What is the cost of joining R and S using a **block nested loops join**, assuming R is the outer relation?

(b) What is the cost of joining R and S using a **block nested loops join**, assuming S is the outer relation?

(c) What is the cost of joining R and S using a **sort-merge join**? Explain how many pages are used in memory and how. Also, specify what exactly is written to disk and when. Then compute the total cost for sort-merge join.

(d) What is the cost of joining joining R and S using a **hash join**? You may assume the hash function works perfectly, creating partitions of equal size. Explain how many pages are used in memory and how. Also, specify what exactly is written to disk and when. Then state the cost of the join.

(e) **First** describe how to join the two relations if S has a

i. clustered

ii. unclustered

index on the **join attribute b.**
Then, estimate the cost of the join using index-nested loop join for each type of index.

Q5 **Cardinality Estimation**

Suppose you are given a table Sales(month, type, price) representing sales of certain types of clothing items from the set swimsuit, mittens, gardening-gloves, in a particular month (encoded as month numerals 1, 2, 3, ... 12). Consider the following query:

$$\sigma_{month \in 10,11,12 \wedge type = swimsuit} \text{ (Sales)}$$

(a) Suppose the database maintains statistics consisting of

i. the total number of records, N;

ii. a one-dimensional histogram on month, whose counts are m1 ... m12;

iii. a one-dimensional histogram on type, whose counts are t_{swim}, t_{mitten}, t_{garden}.

Write a mathematical expression in terms of the variables above, for the estimate the optimizer would use for the cardinality of this query.

(b) Explain the assumptions involved in this estimate. Using this data as an example, propose some reasons why this estimate may be incorrect.

SUBMISSION

You should send a pdf file, named 'eXXXXXXX.pdf' (your seven-digit ID number) contains your answers. You can prepare the pdf using both 'DOCS' or 'Latex', doesn't matter.

For each question, please show the steps that you've followed to find the answer. **You will not get any credits from direct answers without any explanation**.