COMP 3311: Database Management Systems

Lecture 16 Exercises
Query Processing: Expression Evaluation

```
select distinct name
from Student;
```

Exercise 1: The Student relation consists of 10,000 tuples sorted on student id. Each student has 5 attributes, each 20 bytes, so the tuple size is 100 bytes. The page size is 1,000 bytes, so $bf_{Student}=\lfloor 1000/100 \rfloor=10$. Therefore, $B_{Student}=\lfloor 10000/10 \rfloor=1000$ pages. Assume that the buffer size *M* is 100 pages and that there are 5,000 different student names. There is no index. We want to evaluate the query on the right.

a) Projection using external sorting

Pass 0 (sort) page I/O cost:

Pass 1 (merge) page I/O cost:

Projection page I/O cost:

Number of query result pages written:

b) Projection using hashing (using 20 partitions)

Partitioning page I/O cost:

Duplicate elimination page I/O cost:

Projection page I/O cost:

Number of query result pages written:

Do not upload this exercise sheet to Canvas.

COMP 3311: Database Management Systems

Lecture 16 Exercises
Query Processing: Expression Evaluation

Exercise 2: Sailor(sailorId, sName, rating, age) Reserves(sailorId, boatId, rDate)

For the Sailor relation, each tuple is 50 bytes, a page can hold 80 tuples and there are 500 full pages. For the Reserves relation, each tuple is 40 bytes, a page can hold 100 tuples and there are 1,000 full pages. There are 10 different sailor ratings and 100 different boats. Assume that sailors are distributed uniformly over the 10 ratings and reservations are distributed uniformly over the 100 boats.

Our goal is to process the query:

```
select sName
from Sailor natural join Reserves
where boatId=30
      and rating>5;
```

Some useful statistics

- On average, each sailor has 2.5 reservations.
- On average, each boat has 1,000 reservations.
- On average, for each rating there are 4,000 sailors.

Estimate the approximate cost of processing the query using a fully pipelined execution method (i.e., do not materialize anything except the query result).

Assume that the Sailor relation contains a clustering B$^+$-tree index with 3 levels on sailorId and the Reserves relation contains a hash index on boatId. Both the B$^+$-tree and hash index can fit 400 index entries per page. For non-clustering indexes, each pointer is assumed to lead to a different page.

a) Cost to evaluate $\sigma_{boatId=30}$

Strategy 1:

Page I/O cost:

Strategy 2:

Page I/O cost:

b) Cost to evaluate $\sigma_{rating>5}$

Strategy:

Page I/O cost:

c) Cost to evaluate Sailor ⋈ Reserves

Strategy:

Page I/O cost:

d) Cost to evaluate π_{sName}

Strategy:

Page I/O cost:

Query page I/O cost:

Do not upload this exercise sheet to Canvas.

Name: (1) ______________________/________________ **Student#:** (1) __________________ **Date:** ___________

Family/Given (PRINT) Given/First (PRINT)

Name: (2) ______________________/________________ **Student#:** (2) __________________

Family/Given (PRINT) Given/First (PRINT)

NOTE: You are highly encouraged to do this exercise with a partner.

COMP 3311: Database Management Systems

Lecture 16 Exercises
Query Processing: Expression Evaluation

select distinct sName
from Sailor;

Exercise 3: The Sailor relation consists of 40,000 tuples sorted on sailorId. Each sailor has 4 attributes, each 10 bytes, so the tuple size is 40 bytes. The page size is 800 bytes so $bf_{Sailor}=\lfloor 800/40 \rfloor=20$. Therefore, $B_{Sailor}=\lceil 40000/20 \rceil=2000$ pages. Assume that the buffer size M is 100 pages and that 5% of sailor names are the same. There is no index. We want to evaluate the query on the right.

a) Projection using external sorting

Pass 0 (sort) page I/O cost:

Pass 1 (merge) page I/O cost:

Projection page I/O cost:

Number of query result pages written:

b) Projection using hashing (using 40 partitions; no optimization)

Partitioning page I/O cost:

Duplicate elimination page I/O cost:

Projection page I/O cost:

Number of query result pages written:

You must upload this completed exercise sheet to Canvas by **11 p.m. today.**

Name: (1) ______________________/________________ **Student#:** (1) __________________ **Date:** ___________

Family/Given (PRINT) Given/First (PRINT)

Name: (2) ______________________/________________ **Student#:** (2) __________________

Family/Given (PRINT) Given/First (PRINT)

NOTE: You are highly encouraged to do this exercise with a partner.

COMP 3311: Database Management Systems

Lecture 16 Exercises
Query Processing: Expression Evaluation

Exercise 4: Student(sId, name, deptId, address) EnrollsIn(courseId, *sId*, semester, grade)

The Student relation contains 10,000 tuples in 1,000 pages and the EnrollsIn relation contains 50,000 tuples in 5,000 pages. There are 25 different departments and 1,000 different courses. All attributes have the same length. Each available index is a B$^+$-tree with 3 levels. For non-clustering indexes, each pointer is assumed to lead to a different page.

Our goal is to process the query:

```
select name
from Student natural join EnrollsIn
where courseId='COMP3311'
        and deptId='COMP';
```

Some useful statistics

- On average, a student enrolls in 5 courses.
- On average, a department has 400 students.
- On average, each course has an enrollment of 50 students.

Estimate the approximate cost of processing the query using a fully pipelined execution method (i.e., do not materialize anything except the query result).

Assume that the Student relation contains a clustering index on deptId and the EnrollsIn relation contains a non-clustering index on sId.

a) Cost to evaluate $\sigma_{\text{courseId='COMP3311'}}$

Strategy:

Page I/O cost:

b) Cost to evaluate $\sigma_{\text{deptId='COMP'}}$

Strategy:

Page I/O cost:

c) Cost to evaluate Student ⋈ EnrollsIn

Strategy:

Page I/O cost:

d) Cost to evaluate π_{name}

Strategy:

Page I/O cost:

Query page I/O cost:

You must upload this completed exercise sheet to Canvas by **11 p.m. today.**