# CSC553: Homework 3

Due: May $8^{th}$, 2022

This assignment is on relational algebra, logical plan equivalence, and indexes.

## 1 Iterators

Write the iterator algorithm for grouping operator and set intersection. Provide only pseudo-code.

## 2 One-pass algorithms

Give a one pass algorithm for left outer join between $R(x, y)$ and $S(y, z)$ assuming R fits in memory.

## 3 Joins

Consider the join between R and S on R$\bowtie_{R.a=S.b}$ 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 10,000 tuples and has 10 tuples per page.
Relation S contains 2000 tuples and also has 10 tuples per page.

Attribute b of relation S is the primary key for S.

Both relations are stored as simple heap files. Neither relation has any indexes built on it.

52 buffer pages are available.

1. What is the cost of joining R and S using a page-oriented simple nested loops join? What is the minimum number of buffer pages required for this cost to remain unchanged?

2. What is the cost of joining R and S using a block nested loops join? What is the minimum number of buffer pages required for this cost to remain unchanged?

3. What is the cost of joining R and S using a sort-merge join? What is the minimum number of buffer pages required for this cost to remain unchanged?

4. What is the cost of joining R and S using a hash join? What is the minimum number of buffer pages required for this cost to remain unchanged?

5. What would be the lowest possible I/O cost for joining R and S using any join algorithm, and how much buffer space would be needed to achieve this cost? Explain briefly.

6. How many tuples does the join of R and S produce, at most, and how many pages are required to store the result of the join back on disk?

# 4 External Merge Sort

You are trying to sort the S table which has 200 pages. Suppose that during Pass 0, you have 10 buffer pages available to you, but for Pass 1 and onwards, you only have 5 buffer pages available.

1. How many sorted runs will be produced after each pass?

2. How many pages will be in each sorted run for each pass?

3. How many I/Os does the entire sorting operation take?

# 5 Partitioned Hash Join

Consider relations R and S with B(R) = 1000 and B(S) = 800. Explain how a DBMS could efficiently join these two relations given that only 21 pages can fit in main memory at a time. Present a solution that uses a hash-based algorithm. Your presentation should be detailed: specify how many pages are allocated in memory and what they are used for; specify what exactly is written to disk and when.