CS432/532 Homework 4

1. [30%] Consider the following SQL query for a banking database system. For simplicity, assume that there is only one account type and the status of a customer is bronze, silver, or gold.

   select account_id, customer_name
   from accounts
   where status = 'gold member' and customer_city = 'Boston';

   Assume the following:
   - Each account tuple occupies 200 bytes
   - The size of a page is of 4KB (i.e., 4000 bytes)
   - There are 50 customers (out of total 10,000 customers) with gold member status
   - 100 customers live in Boston.

   Given that, answer the questions below.

a) [15%] If there is a secondary index on status and a secondary index on customer_city, which index the system should use first to optimize the query? Simply say 'status' or 'customer_city' [5%]. Briefly explain your answer [10%].

b) [15%] If there is a primary index on status and a secondary index on customer_city, the system will first use the primary index to find the gold customers and then select the gold customers in Boston. For simplicity, assume that the entire B+ tree is in memory. Given that, answer the questions below:
   1) How many I/O pages the system should do [2%]? Briefly explain your answer [8%].
   2) Are they sequential or random I/O [1%]? Briefly explain your answer [4%].

2. [15%] Consider the join R $\bowtie_{R.A = S.B}$ S, where R and S are two relations. Three join methods, i.e., nested loop, sort merge, and hash join, are discussed in class. Nested loop and sort merge may benefit from the existence of indexes. Identify three different situations such that each of the following claims is true for one situation. In this question, the performance is compared based on the number of I/O pages. Suppose R and S consist of N and R pages, respectively. Assume that N > M. Also, assume that the memory buffer for the join is not large enough to hold the entire R.

a) [5%] Suppose that both tables are large such that none can be entirely held in the memory buffer, there is no index on either joining attribute, both tables are already sorted on the respective joining attributes and at least one of the joining attributes does not have repeating values. Given that, which method above will perform best? Simply name the method. No explanation is needed.

b) [5%] Suppose that the hash table of the smaller table can be held in memory and at least one memory page remains for the larger table, the larger table R cannot be held in memory and it is not sorted on the joining attribute, and no index exists on either joining attribute. Given that, which method above will perform best? Simply name the method. No explanation is needed.

c) [5%] When S is very small and R has a (primary) index on the joining attribute (i.e., R.A), which method above will perform best? Simply name the method. No explanation is needed.

3. [25%] For three tables R1(A, B, C), R2(C, D, E), and R3(E, F), assume the following is true:
   - R1.C is a non-null foreign key referencing R2.C.
   - R2.E is a non-null foreign key referencing R3.E.
   - R1 has 3000 tuples, R2 has 1000 tuples, and R3 has 850 tuples.

   To join the three tables together, we consider two query executions below:
   - Plan A: (R1 ⋈ R2) ⋈ R3
   - Plan B: R1 ⋈ (R2 ⋈ R3)

   Given all the information above, answer the following questions:

   a) [5%] Which plan will be faster? Simply say Plan A or B. No explanation is needed.

   b) [10%] How many tuples in total are produced by Plan A including intermediate tuples produced during the query processing [1%]? Briefly explain your answer [9%].

   c) [10%] How many tuples in total are produced by Plan B including intermediate tuples produced during the query processing [1%]? Briefly explain your answer [9%].

4. [30%] Consider the following three relations:

   Supplier(Supp#, Name, City, Specialty)
   Project(Proj#, Name, City, Budget)
   Order(Supp#, Proj#, Part_name, Quantity, Cost)

   Apply the four heuristic optimization rules discussed in class to find an efficient execution plan for the following SQL query. Assume that the number of suppliers in New York City is much smaller than the number of the projects whose budget is over $100000.

   ```
   select Supplier.Name, Project.Name
   from Supplier, Order, Project
   where Supplier.City = 'New York City' and Project.Budget > 100000
   and Supplier.Supp# = Order.Supp# and Order.Proj# = Project.Proj#
   ```

   a) [20%] Draw the final query tree acquired by applying the four rules.

   b) [10%] Write the relational algebra expression that corresponds to the optimized query tree.