Connor Brown

CS44800 Homework 3

**Question 1. (0.5 point)**

Specify the following queries in relational algebra (RA) on the COMPANY database schema in

Figure 5.5 in the Textbook. Do not use any other information from the database state on Figure

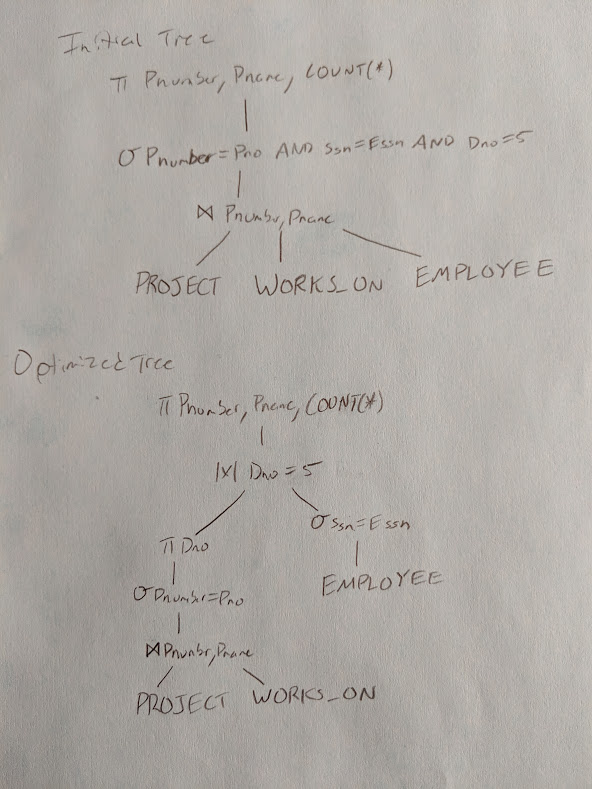
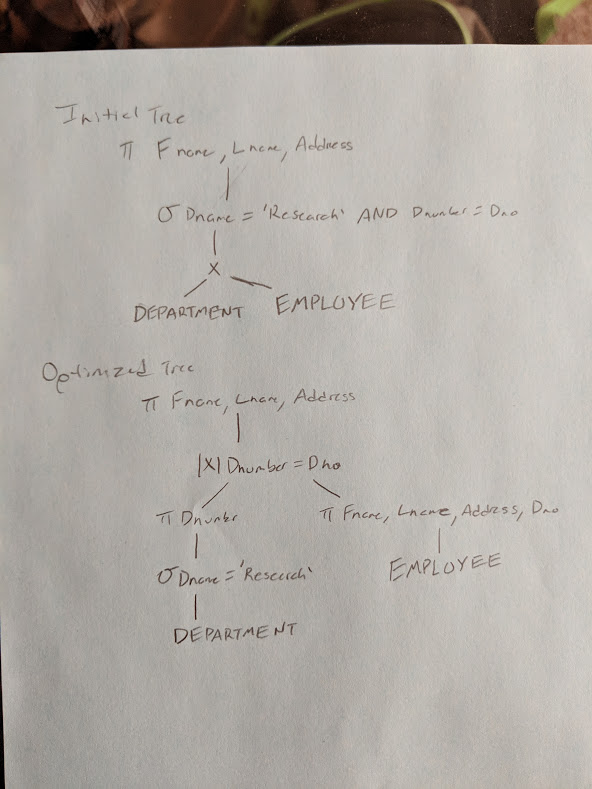
5.6 to build the queries.

1. List names of employees (Fname, Minit, Lname) who are directly supervised by Franklin Wong.
   1. wong\_SSNssn(Lname = ‘Wong’ and Fname = ‘Franklin’(employee))
   2. answerFname,Minit,Lname(employee superssn = ssnwong\_SSN)
2. List names of employees (Fname, Minit, Lname) who are NOT assigned to every project.
   1. working\_employees(ssn)essn(works\_on)
   2. names\_of\_working\_employeesworking\_employees \* employee
   3. answerFname,Minit,Lname(employees - names\_of\_working\_employees)
3. List names of employees (Fname, Minit, Lname) in Department 4 who work more than 10 hours per week on the ‘Newbenefits’ project.
   1. all\_empemployeessn = essn works\_on pno = pnumber project
   2. selected\_empdno = 4 and pname = “Newbenefits’ and hours > 10.0(all\_emp)
   3. answerFname,Minit,Lname(selected\_emp)
4. List the names of department managers (Fname, Minit, Lname) who have no dependents.
   1. dpt\_man(ssn)mgrssn(department)
   2. emp\_with\_depend(ssn)essn(dependent)
   3. answerFname,Minit,Lname(employee \* (dpt\_man - emp\_with\_depend))

**Question 2. (0.5 point)**

Consider SQL queries Q1 and Q4A in Chapter 6.

1. Draw the initial query tree for each of these queries, and then show how the query tree is optimized by the algorithm outlined in Section 19.1.



1. For each query, compare the initial query and the final query tree. Discuss the difference between each tree.
   1. In Q1 the difference between the two is that in the optimized one you are able to get all employees with and then remove the ones that do not have the department or get the department and remove all the departments that are not the one you are looking for. This allows it to get an answer in two different ways.
   2. In Q4 the difference is that in the optimized one getting an employee will allow you to search the project and what is worked on separately.

**Question 3. (0.50 point)**

Storing natural joins of base relations can lead to a problem referred as **update anomalies**. These can be further classified in insertion anomalies, deletion anomalies and modification anomalies.

1. Discuss each anomaly.
   1. Insertion Anomalies - occurs when certain attributes cannot be inserted into the database without the presence of other attributes.
   2. Deletion Anomalies - occurs when certain attributes are lost because of the deletion of other attributes.
   3. Modification Anomalies - occurs when one or more instances of duplicated data is updated.
2. What update anomalies occur in the EMP\_PROJ and EMP\_DEPT relations of Figures 14.3 and 14.4?
   1. In 14.3 there is a modification anomaly that occurs.
   2. In 14.4 there is an insert anomaly that occurs.
3. Discuss what these anomalies are considered bad using the example of tables EMP\_PROJ and EMP\_DEPT. How do you fix these anomalies?
   1. There is redundant data which can be fixed by removing them from EMP\_PROJ and EMP\_DEPT.
   2. By having redundant data there is unnecessary updates when modifying different aspects. Again this would be fixed by no longer having redundant data.

**Question 4. (0.5 point)**

Consider the universal relation R = {A, B, C, D, E, F, G, H, I, J} and the set of functional dependencies F = {{A, B}→{C}, {A}→{D, E}, {B}→{F}, {F}→{G, H}, {D}→{I, J}}.

1. What is the closure of the set {A, C}?
   1. The closure of the set would be {D,E,I,J}
2. What is the key for R?
   1. AD and AE
   2. Since AD and DIJ AIJ
   3. Union of A is AADEIJ, thus ABABDEIJ through augmentation
   4. Since ABC ABABCDEIJ
   5. Since BF and FGH, then BGH through transitivity
   6. Thus ABAGH holds and ABAF holds
   7. Union ABABCDEFGHIJ
   8. So therefore AB is a key
3. What is the minimum cover for the set of functional dependencies F?
   1. The minimum cover would be {A}{D,E}
4. Decompose R into 2NF.
   1. R1(A, B, C)
   2. R2(A, D, E, I, J)
   3. R3(B, F, G, H)
5. Decompose R into 3NF.
   1. R1(A, B, C)
   2. R2.1(A, D, E) R2.2(D, I, J)
   3. R3.1(B, F) R3.2(F, G, H)

**Question 5. (0.5 point)**

Discuss the advantages and disadvantages of using (a) an unordered file, (b) an ordered file, and (c) a static hash file with buckets and chaining. Which operations can be performed efficiently on each of these organizations, and which operations are expensive?

1. An unordered file
   1. Advantages - Insertion of simple records are being added at the end of the file, which makes it easier for retrieval. Since it is unsorted removing is easy, but leaves blank spaces
   2. Disadvantages - Blank spaces may appear. Sorting time is can be very long.
2. An ordered file
   1. Advantages - Recording a sequential based file is more efficient as all the files are being stored in order. Helpful when large data is present. Easier to find data.
   2. Disadvantages - Rearranging of the file would be needed for storing, modifying, or deleting any records. Time for functions can be long.
3. A static hash file
   1. Advantages - Much faster than the others. Very efficient with large volume of data.
   2. Disadvantages - Difficult to implement. Very expensive.

The hashing technique would be the most efficient form of organization. It is also the most expensive because it uses complicated structures.

**Question 6. (0.5 point)**

Suppose that a disk unit has the following parameters: seek time s = 20 msec; rotational delay rd = 10 msec; block transfer time btt = 1 msec; block size B = 2400 bytes; interblock gap size G = 600 bytes. An EMPLOYEE file has the following fields: Ssn , 9 bytes; Last\_name , 20 bytes;

First\_name , 20 bytes; Middle\_init , 1 byte; Birth\_date , 10 bytes; Address , 35 bytes; Phone , 12

bytes; Supervisor\_ssn , 9 bytes; Department , 4 bytes; Job\_code , 4 bytes; deletion marker, 1

byte. The EMPLOYEE file has r = 30,000 records, fixed-length format, and unspanned blocking.

Write appropriate formulas and calculate the following values for the above EMPLOYEE file:

1. Calculate the record size R (including the deletion marker), the blocking factor bfr, and the number of disk blocks b.
   1. R = (9 + 20 + 20 + 1 + 10 + 35 + 12 + 9 + 4 + 4 + 1) = 125 bytes
   2. Bfr = floor(B/R) = floor(2400/125) = 19 records per block
   3. b = ceiling(r/bfr) = ceiling(30000/19) = 1579 blocks
2. Calculate the wasted space in each disk block because of the unspanned organization.
   1. Total Capacity = (2400 + 600) \* 1579 = 4737000 = 4.737 MB
   2. Useful Capacity = 2400 \* 1579 = 3789600 = 3.7896 MB
   3. Wasted Capacity = Total - Useful = 4737000 - 3789600 = 947400 = 947.4 kB
3. Calculate the transfer rate tr and the bulk transfer rate btr for this disk unit (see Appendix B for definitions of tr and btr).
   1. Rpm = (60)(1000)(½)(1/10) = 3000
   2. tr = ((4737000 bytes)/(60\*3000rpm)) \* 1000 msec/sec = 26,316.6 bytes/msec
   3. btr = 2400/26316.6 = 0.0912 msec
4. Calculate the average number of block accesses needed to search for an arbitrary record in the file, using linear search.
   1. 20 msec + 10 msec + 1579 blocks \* 0.0912 msec = 174.0048 msec