Solution to Problem Set 6

Problem 1. Consider the relation `E(Eno, Ename, Dname, Salary)`. Let the domain of `Dname` be {`CS, EE, ..., History` } and domain of `Salary` be the set of positive integers. The most frequent queries on `E` use the set of simple predicates {`Dname = History, Dname = CS, Salary >= 60000, Salary <= 30000`}. Compute the primary horizontal fragments of E.

Answer. `E` is partitioned into the following 9 fragments

- $E_1 = \sigma_{Dname=\text{"}CS\text{"} and Salary \geq 60000}(E)$
- $E_2 = \sigma_{Dname=\text{"}History\text{"} and Salary \geq 60000}(E)$
- $E_3 = \sigma_{Dname!=\text{"}CS\text{"} and Dname!=\text{"}EE\text{"} and Salary \geq 60000}(E)$
- $E_4 = \sigma_{Dname=\text{"}CS\text{"} and Salary \leq 30000}(E)$
- $E_5 = \sigma_{Dname=\text{"}History\text{"} and Salary \leq 30000}(E)$
- $E_6 = \sigma_{Dname!=\text{"}CS\text{"} and Dname!=\text{"}History\text{"} and Salary \leq 30000}(E)$
- $E_7 = \sigma_{Dname=\text{"}CS\text{"} and 30000 < Salary < 60000}(E)$
- $E_8 = \sigma_{Dname=\text{"}History\text{"} and 30000 < Salary < 60000}(E)$
- $E_9 = \sigma_{Dname!=\text{"}CS\text{"} and Dname!=\text{"}History\text{"} and 30000 < Salary < 60000}(E)$

Problem 2. Consider the relations `P(Pno, Pname, Budget, Loc)` and `A(Eno, Pno, Duration)`. Let `P` be horizontally fragmented into $P_1 = \sigma_{Pno<100}(P)$ and $P_2 = \sigma_{Pno \geq 100}(P)$; let `A` be horizontally fragmented into $A_1 = \sigma_{Pno<50}(A)$; $A_2 = \sigma_{50 \leq Pno < 100}(A)$; $A_3 = \sigma_{Pno \geq 100}(A)$. Transform the following SQL query into a reduced algebraic query tree:

```
SELECT  Duration, Budget
FROM    A, P
WHERE   A.Pno = P.Pno AND P.Pname = "DB"
```

Answer.

```
                                          |
                                          |
                                          |
       ----------------------------------UNION ----------------------------
       |                                 |                               |
       |                                 |                               |
       |                                 |                               |
  PR(Duration,Budget)               PR(Duration,Budget)             PR(Duration,Budget)
       |                                 |                               |
       |                                 |                               |
       |                                 |                               |
  ----NJ(Pno) ----                  ----NJ(Pno) ----                ---NJ(Pno) ----
  |             |                   |             |                 |             |
  |             |                   |             |                 |             |
PR(Pno,Budget)  PR(Pno,Duration)  PR(Pno,Budget)  PR(Pno,Duration)  PR(Pno,Budget)  PR(Pno,Duration)
  |             |                   |             |                 |             |
  |             |                   |             |                 |             |
  |             |                   |             |                 |             |
SEL(Pname="DB") |                 SEL(Pname="DB") |               SEL(Pname="DB") |
  |             |                   |             |                 |             |
  |             |                   |             |                 |             |
  |             |                   |             |                 |             |
  P1            A1                  P1            A2                P2            A3
```

"NJ" stands for natural join, "PR" stands for projection, and "SEL" stands for selection.

Problem 3. Consider the relations `E`, `P` and `A` as defined in the previous two problems. Let `P` be horizontally fragmented into $P_1 = \sigma_{Pno<100}(P)$ and $P_2 = \sigma_{Pno \geq 100}(P)$; let `E` be vertically fragmented into $E_1 = \pi_{Eno,Ename,Salary}(E)$ and $E_2 = \pi_{Eno,Dname}(E)$. Let the horizontal fragmentation of `A` be derived from that of `P`, based on the `Pno` attribute (assume `Pno` is the key of `P`). Reduce the following query:

```
SELECT   Ename
FROM     E, A, P
WHERE    P.Pno = A.Pno AND E.Eno = A.Eno AND P.Loc = "Palo Alto"
```

Answer.

```
                          |
                          |
                     PR(Ename)
                          |
                          |
  ---------------- NJ(Eno)---------------
  |                                      |
  |                                      |
  |                                      |
 E1                           -------- UNION ----------
                              |                       |
                              |                       |
                              |                       |
                              |                       |
                           PR(Eno)                 PR(Eno)
                              |                       |
                              |                       |
                              |                       |
                        -- NJ(Pno) --           -- NJ(Pno) ---
                        |           |           |             |
                        |           |           |             |
                     PR(Pno)  PR(Pno,Eno)    PR(Pno)      PR(Pno,Eno)
                        |           |           |             |
                        |           |           |             |
                        |           |           |             |
              SEL(Loc="Palo Alto")  |  SEL(Loc="Palo Alto")   |
                        |           |           |             |
                        |           |           |             |
                        |           |           |             |
                       P1          A1          P2            A2
```

Problem 4. Compute k_0 for the example shown in slide 20 of lecture 14. (Note: There was a typo in the originally handed out slides. The correction was announced in class - for site 2, the min should be 7 and not 10).

Answer. With reference to the figure shown on slide 20 of lecture 14, we know that $\frac{k_0-5}{5} * 10$ of the tuples from site 1 and $\frac{k_0-7}{10} * 10$ of the tuples from site 2 will be transferred to the first partition. Since we require the two partitions to contain the same number of tuples, we get $2(k_0-5)+(k_0-7) = 10$. This gives us $k_0 = 9$. Therefore all tuples with sort attiribute < 9 will be transferred to the first partition and all those with values ≥ 9 will be transferred to the second partition.