CS386 Homework 7

Russell Miller

March 3, 2011

Part 1

1. **a.** This would be closer to the case of (M+N) because if there is a match between Patient.SSN and Provider.SSN there can only be one row returned, since it is a unique value for each table.

   b. This would be closer to the case of (M*N) because it is possible for every Student to be male, and every Staff to be male, in which case the number of rows would be the number of Staff times the number of Students.

2. **a.** Using an external sort the time would be 2*M, or just 2.
   b. The page-oriented nested loop join requires that each table be scanned M times, and the total cost is M+M*N – in this case 2 + 2N.
3. **a.** No.
   b. Clustering indexes improves sort time, but it cannot improve join time. It is still a sequential read for each table, and because of the indexes actually takes longer.

Part 2A

1. The result of the query would be sorted by agent_id.
2. This would be sorted by salary.
3. Because those are the clustered indexes.

Part 2B

1. **a.** For a) the expected cost is 54.60..54.61. For b) the expected cost is 54.60..54.61. So they have the same estimated cost.
   b. Query a) was faster clocking at 5.242 ms. Query b) took 5.367 ms.
   c. Postgres chose to use Hash join for both queries.
   d. It doesn't look like it's using clustered indexes.
2. **a.** For a) the expected cost is 48.24..48.25. For b) the expected cost is 48.24..48.25.
   b. Query b) was faster, clocking 3.822 ms. Query a) timed 3.854 ms.
   c. They were both Hash joins.
   d. There are still no clustered indexes.
3. **a.** For a) the expected cost is 48.24..48.25. For b) the expected cost is 29.12..29.13.
   b. Query b) was faster, clocking 2.141 ms. Query a) timed 3.903 ms.
   c. They were both Hash joins.
   d. Query b) was clustered on agent_salary, but Query a) was not.
4. For many of those queries, none of the rows got filtered out.

Part 3

1. **a.** Yes, the constraint $salary > 0$ is pushed into the inner query.

   b. It seems like it would be more efficient because less rows would be returned to the outer query.

   c. No.

2. **a.** Yes, the constraint $salary < 56000$ is pushed into the inner query.

   b. This time it's definitely more efficient.

   c. Yes, it is clustered on salary.

3. **a.** Yes, the contraint $salary < 56000$ is pushed into the first inner query alongside $salary > 55000$.

   b. I would bet that this is more efficient because of the clustered index.

   c. Yes.