**CS443 -- Lab 6**

1. **Use query tree to optimize the following query**

SELECT reps.name, off.city

FROM salesreps reps, offices off

WHERE reps.rep\_office=off.office

AND reps.sales<reps.quota

AND off.sales<off.target;

This is the initial tree.



There are select conditions on the Salesreps and Offices tables that can be moved down to limit the number of records from these two tables that go into the Cartesian product. We only want records from the Salesreps table where the sales is less than the quota and we only want records from the Offices table where the sales is less than the target. This is the improved tree.



The next improvement is to replace the Cartesian product followed by a join condition with a join operation. This is the next improved tree.



There is also a limited number of fields needed from the Salesreps and Offices tables so we can move these projections down to reduce the number of attributes carried along to the join operation. This is the next and final improved tree.



1. **Use query tree to optimize the following query**

SELECT Order\_Num, Amount, Company, Name, City

FROM Orders, Customers, Salesreps, Offices

WHERE Cust = Cust\_Num

AND Cust\_Rep = Empl\_Num

AND Rep\_Office = Office

AND Amount > 10000

AND Region = “Eastern”

This is the initial tree.



There are select conditions on the Orders and Offices tables that can be moved down to limit the number of records from these two tables that go into the Cartesian products. We only want records from the Orders table where the Amount is greater than 1000 and we only want records from the Offices table where the Region is “Eastern.” This is the improved tree.



The final Cartesian product can be limited by moving join conditions down so they are applied earlier. This is the improved tree.



The next improvement is to replace the Cartesian products followed by join conditions with join operations. This is the next improved tree.



The final improvement is to only select the fields that are needed further up the tree; this is done at several points in the tree. This is the final tree.



1. **Use query tree to optimize the following query**

SELECT product\_id, description, cust\_num, company

FROM ORDERS, CUSTOMERS, PRODUCTS

WHERE cust=cust\_num

AND MFR=MFR\_ID

AND Product=Product\_Id

AND Amount < 1500

This is the initial tree.



There is one select condition on the Orders table that can be moved down to limit the number of records from this table that go into the Cartesian product. We only want records from the Orders table where the Amount is less than 1500. This is the improved tree.



The next improvement is to switch the Customers and Products tables. By combining Products with Orders sooner, the number of records that go into the second Cartesian product will be limited. Combining Customers with Orders does not limit records as much since, in general, all Customers have made Orders. This is the improved tree.



The product information can now be combined after the first Cartesian product to limit the number of records that go into the second Cartesian product. This is the improved tree.



The next improvement is to replace the Cartesian products followed by join conditions with join operations. This is the next improved tree.



The final improvement is to only select the fields that are needed further up the tree; this is done at several points in the tree. This is the final tree.

