CWRU EECS 341 Intro to Databases
Spring 2019
Eamon Johnson

Assignment #5
due April 23, 2019 at noon

Submit a pdf to Canvas by the due date and time, generated using a word processor of your choice or by photographing a clean, legible hand written paper.

Consider the following design for an airline company database:

passenger(id, name, home_airport) FK home_airport references airport.name
ticket(passenger_id, flight_id, price) FKs to passenger.id and flight.id
flight(id, number, departure_datetime, route_id) FK route_id references route.id
route(id, departure_airport, arrival_airport, miles) FKs to airport.name
airport(name)

The following query displays all future flights for a given passenger, identified by the placeholder ":PASS_ID". Note that the "now()" function in the predicate returns the current time.

```
select p.name, f.departure_datetime, departure.name, arrival.name
from passenger p, ticket t, flight f, route r, airport departure, airport arrival
where p.id = t.passenger_id and
        p.id = :PASS_ID and
        t.flight_id = f.id and
        f.route_id = r.id and
        f.departure_datetime > now() and
        arrival.name = r.arrival_airport and
        departure.name = r.departure_airport
```

(1) Write a naive (straightforward) expression of this query in relational algebra using only cross joins, and applying predicate selections after the joins.

(2) Convert the expression from (1) into a parse tree.

(3) Apply equivalence rules and heuristic optimizations to create an optimized parse tree, for example, by using theta joins and pushing down selections and projections.

(4) Suppose that on login we would like to show a passenger how many total flights they have completed and how many miles they have traveled. A materialized view could be used to precompute and cache this information so that information display on login is fast. Write a definition for a view that computes this information for a passenger.