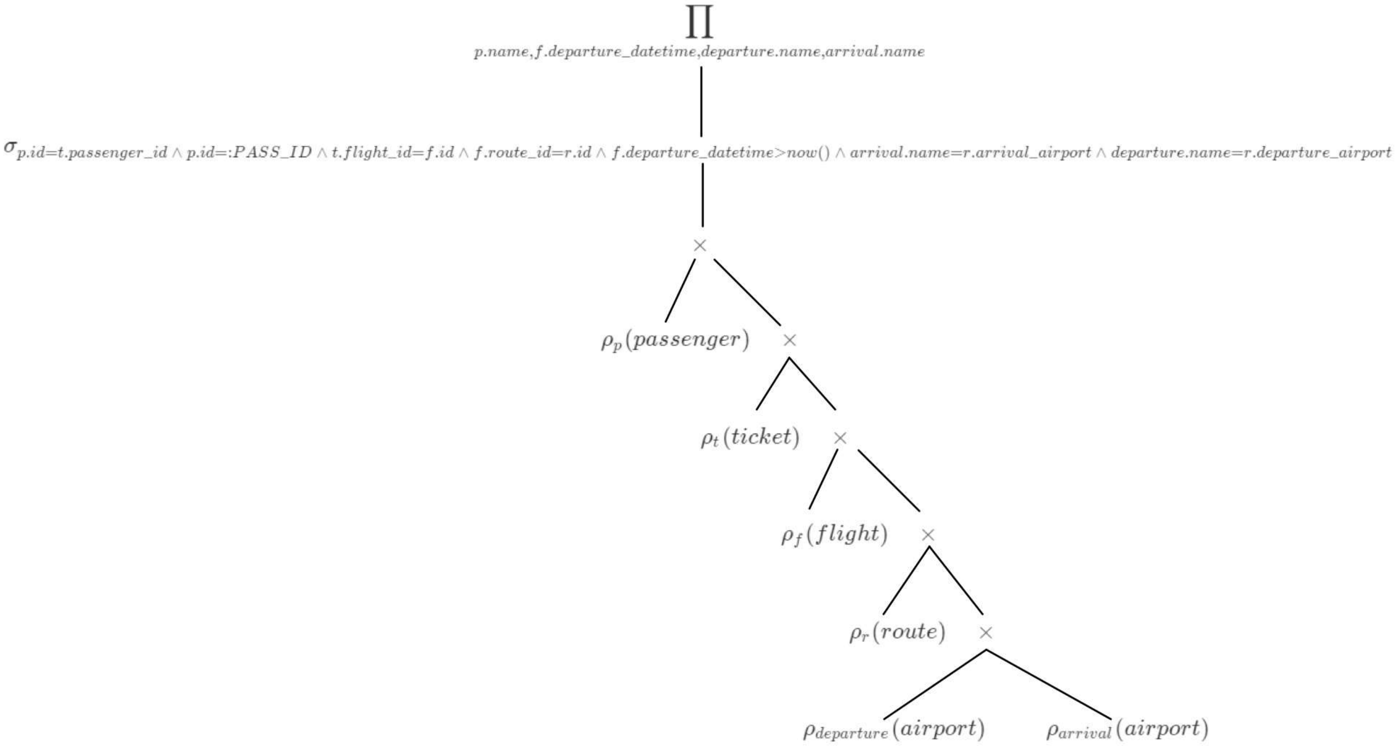
Assignment 5

EECS341 Spring 2019
Yue Shu
Due: Tuesday, April 23, 2019

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

$$\prod_{p.name,f.departure\_datetime,departure.name,arrival.name} (\sigma_{p.id=t.passenger\_id \wedge p.id=:PASS\_ID \wedge t.flight\_id=f.id \wedge f.route\_id=r.id \wedge f.departure\_datetime>now() \wedge arrival.name=r.arrival\_airport \wedge departure.name=r.departure\_airport} (\rho_p(passenger) \times \rho_t(ticket) \times \rho_f(flight) \times \rho_r(route) \times \rho_{departure}(airport) \times \rho_{arrival}(airport)))$$

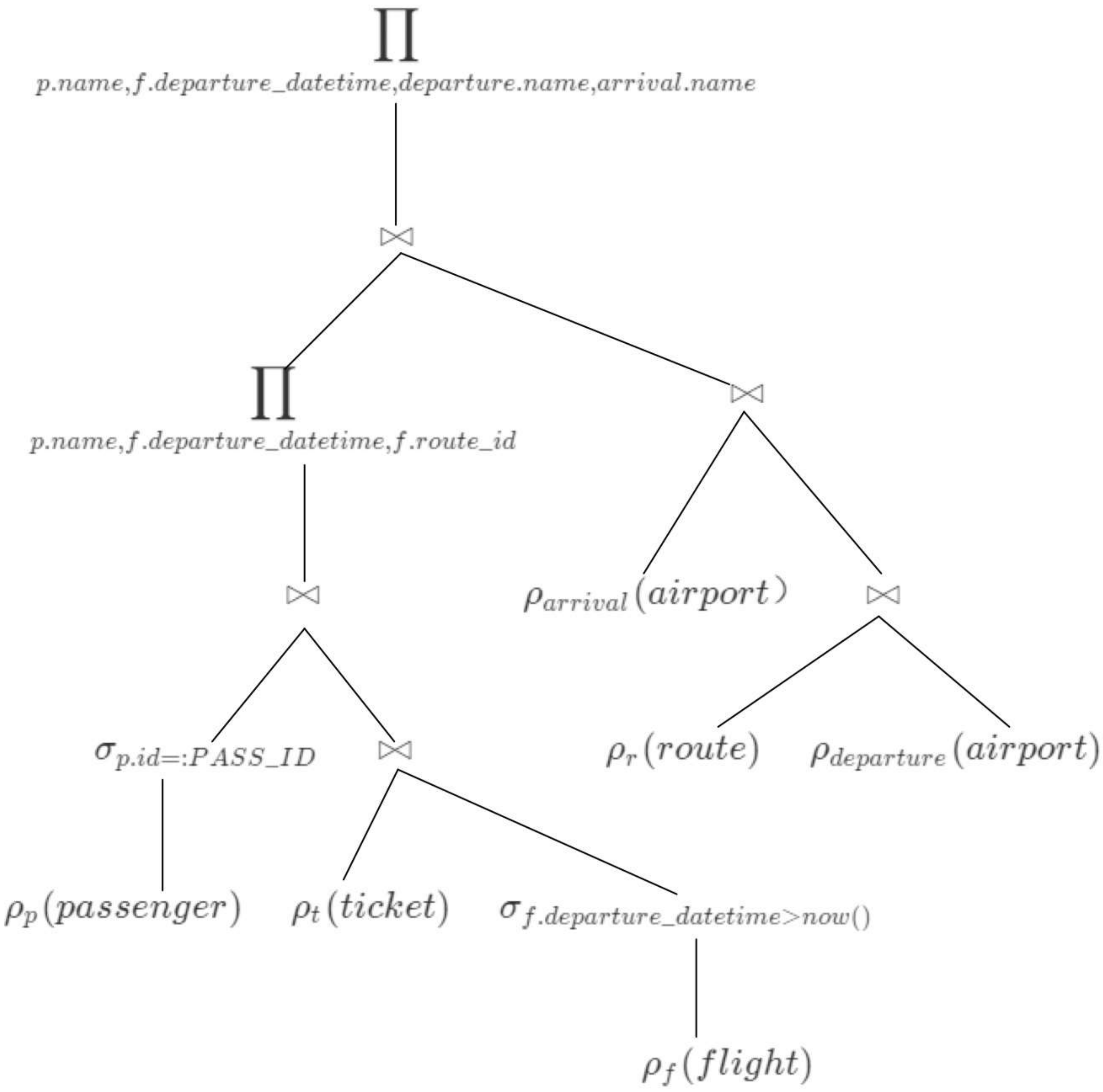
2. Convert the expression from (1) into a parse tree.



$\prod_{p.name,f.departure\_datetime,departure.name,arrival.name}$
$\sigma_{p.id=t.passenger\_id \wedge p.id=:PASS\_ID \wedge t.flight\_id=f.id \wedge f.route\_id=r.id \wedge f.departure\_datetime>now() \wedge arrival.name=r.arrival\_airport \wedge departure.name=r.departure\_airport}$
×
$\rho_p(passenger)$
×
$\rho_t(ticket)$
×
$\rho_f(flight)$
×
$\rho_r(route)$
×
$\rho_{departure}(airport)$
$\rho_{arrival}(airport)$

***CORRECTION**: switch $\rho_r(route)$ with $\rho_{departure}(airport)$, so that $\rho_r(route)$ and $\rho_{arrival}(airport)$ would be in the same depth*

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.



Π p.name,f.departure_datetime,departure.name,arrival.name
⋈
Π p.name,f.departure_datetime,f.route_id
⋈
⋈
ρarrival(airport)
⋈
σp.id=:PASS_ID
⋈
ρr(route)
ρdeparture(airport)
ρp(passenger)
ρt(ticket)
σf.departure_datetime>now()
ρf(flight)

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.

```
CREATE MATERIALIZED VIEW flight_mileage AS (
    SELECT COUNT(DISTINCT flight.id), SUM(route.miles)
    FROM (
        SELECT ticket.passenger_id, COUNT(DISTINCT flight.id), SUM(route.miles)
        FROM ticket, flight, route
        WHERE ticket.flight_id = flight.id
          AND flight.route_id = route.id
          AND flight.departure_datetime < now()
        GROUP BY ticket.passenger_id)
    WHERE ticket.passenger_id = :PASS_ID)
```