

# COMP 3311: Database Management Systems

## Lecture 16 Exercises Query Processing: Join Operation

**Exercise 1:** Use the following information about the relations to estimate the page I/O cost to compute the query result using the stated join strategies.

Sailor(sailorId, sName, rating, age)

Reserves(sailorId, boatId, rDate)

Page size: 1000 bytes

buffer size  $M$ : 100 pages

Each attribute/pointer: 20 bytes

Sailor: 10,000 tuples; 12 tuples/page

Reserves: 40,000 tuples; 16 tuples/page

4 reservations/sailor on average

**Query:** Find the names of sailors who have reservations.  
select sName  
from Sailor, Reserves  
where Sailor.sailorId=Reserves.sailorId;

a) block nested-loop join

i. using Sailor as the outer relation

ii. using Reserves as the outer relation

b) indexed nested-loop join with hash index on Reserves.sailorId (assume no overflow)

c) merge join

d) hash join (assume no overflow)

Name: (1) \_\_\_\_\_ / \_\_\_\_\_ Student#: (1) \_\_\_\_\_ Date: \_\_\_\_\_  
Last/Family (PRINT) Given/First (PRINT)

Name: (2) \_\_\_\_\_ / \_\_\_\_\_ Student#: (2) \_\_\_\_\_  
Last/Family (PRINT) Given/First (PRINT)

**NOTE: Only half credit will be given if done without a partner.**

## COMP 3311: Database Management Systems

### Lecture 16 Exercises Query Processing: Join Operation

**Exercise 2:** The relations  $R_1(A, B, C)$  and  $R_2(C, D, E)$  have the following properties:

- $R_1$  has 20,000 tuples      – 25 tuples of  $R_1$  fit on one page      –  $R_1$  requires 800 pages
- $R_2$  has 45,000 tuples      – 30 tuples of  $R_2$  fit on one page      –  $R_2$  requires 1500 pages

Assuming that there are 800 pages of memory available for processing a join, estimate the page I/O cost for each of the following join strategies for  $R_1$  **JOIN**  $R_2$ .

- a) nested-loop join
  - i. using  $R_1$  as the outer relation
  
  
  
  
  
  
  
  
  
  
  - ii. using  $R_2$  as the outer relation
  
- b) block nested-loop join
  - i. using  $R_1$  as the outer relation
  
  
  
  
  
  
  
  
  
  
  - ii. using  $R_2$  as the outer relation
  
- c) merge join (assume that the relations are not sorted initially)
  - i. sorting cost:
  
  
  
  
  
  
  
  
  
  
  - ii. merge cost:
  
  
  
  
  
  
  
  
  
  
  - iii. total cost:
  
- d) hash join (assume no overflow occurs)