INFO20003 Semester 2, 2019

Assignment 3 – Query Processing and Query Optimisation

Due: 6:00pm Friday 18 October

Submission: Via LMS https://lms.unimelb.edu.au

Weighting: 10% of your total assessment. The assignment will be graded out of 20 marks.

Question 1 (5 marks)

Consider two relations called Parts and Supply. Imagine that relation Parts has 60,000 tuples and Supply has 150,000 tuples. Both relations store 50 tuples per page. Consider the following SQL statement:

```
SELECT *
FROM Parts INNER JOIN Supply
  ON Parts.PartID = Supply.PID;
```

We wish to evaluate an equijoin between Supply and Parts, with an equality condition Parts.PartID = Supply.PID. There are 202 buffer pages available in memory for this operation. Both relations are stored as (unsorted) heap files. Neither relation has any indexes built on it.

Consider the alternative join strategies described below and calculate the cost of each alternative. Evaluate the algorithms using the number of disk I/O's (i.e. pages) as the cost. For each strategy, provide the formulae you use to calculate your cost estimates.

a) Page-oriented Nested Loops Join. Consider Parts as the outer relation. (1 mark)

b) Block-oriented Nested Loops Join. Consider Parts as the outer relation. (1 mark)

c) Sort-Merge Join. Assume that Sort-Merge Join can be done in 2 passes. (1 mark)

d) Hash Join. (1 mark)

e) What would be the lowest possible cost to perform this query, assuming that no indexes are built on any of the two relations, and assuming that sufficient buffer space is available? What would be the minimum buffer size required to achieve this cost? Explain briefly. (1 mark)

Question 2 (5 marks)

Consider a relation with the following schema:

Employee (EmpID, firstname, lastname, department, salary)

The Employee relation has 1200 pages and each page stores 120 tuples. The *department* attribute can take one of six values (“Marketing”, “Human Resource”, “Finance”, “Public Relations”, “Sales and Distribution”, “Operation Management”) and *salary* can have values between 100,000 and 500,000 ([100,000, 500,000]).

Suppose that the following SQL query is executed frequently using the given relation:

```
SELECT *
FROM Employee
WHERE salary > 300,000 AND department = ‘Marketing’;
```

Your job is to analyse the query plans and estimate the cost of the *best plan* utilizing the information given about different indexes in each part.

a) Compute the estimated result size for the query, and the reduction factor of each filter. (1 mark)

b) Compute the estimated cost of the *best plan* assuming that a *clustered B+ tree* index on *(department, salary)* is the only index available. Suppose there are 300 index pages. Discuss and calculate alternative plans. (1 mark)

c) Compute the estimated cost of the *best plan* assuming that an *unclustered B+ tree* index on *(salary)* is the only index available. Suppose there are 200 index pages. Discuss and calculate alternative plans. (1 mark)

d) Compute the estimated cost of the *best plan* assuming that an *unclustered Hash* index on *(department)* is the only index available. Discuss and calculate alternative plans. (1 mark)

e) Compute the estimated cost of the *best plan* assuming that an *unclustered Hash* index on *(salary)* is the only index available. Discuss and calculate alternative plans. (1 mark)

Question 3 (10 marks)

Consider the following relational schema and SQL query. The schema captures information about employees, their departments and the projects they are involved in.

Employee (eid: *integer*, salary: *integer*, name: *char(30)*)
Project (projid: *integer*, code: *char(20)*, start: *date*, end: *date,* eid*: integer*)
Department (did: *integer,* projid: *integer*, budget: *real,* floor: *integer*)

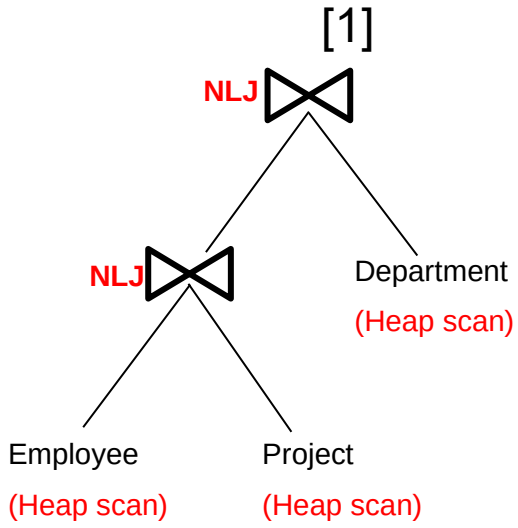
Consider the following query:

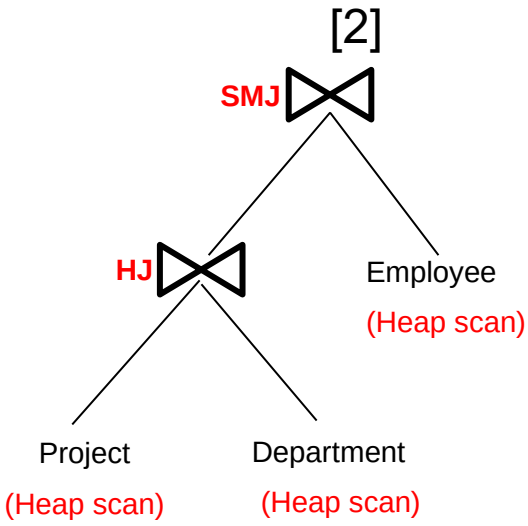
```
SELECT e.name, d.projid
FROM Employee e, Project p, Department d
WHERE e.eid = p.eid AND p.projid = d.projid
  AND e.salary < 300,000 AND p.code = 'alpha 340';
```

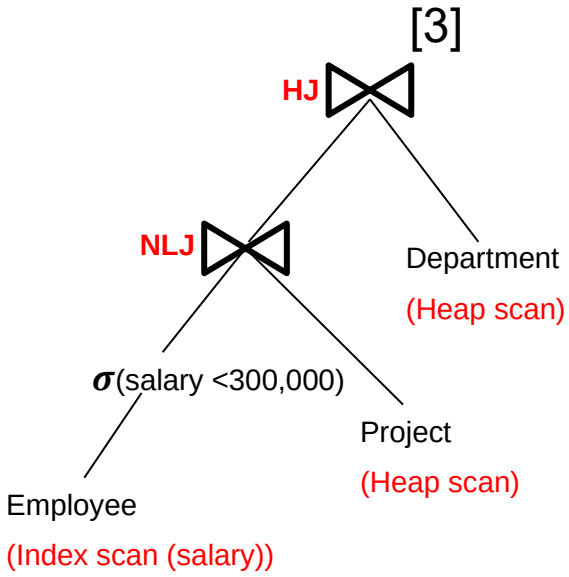
The system's statistics indicate that there are 1000 different *project code* values, and *salary* of the employees range from 100,000 to 500,000 ([100,000, 500,000]). There is a total of 60,000 projects, 5,000 employees and 20,000 departments in the database. Each relation fits 100 tuples in a page. Assume *eid* is a candidate key for Employee, *projid* is a candidate key for Project, and *did* is a candidate key for the Department table. Suppose there exists a *clustered B+ tree* index on *(Project.projid)* of size 200 pages and suppose there is a *clustered B+ tree* index on *(employee.salary)* of size 10 pages.

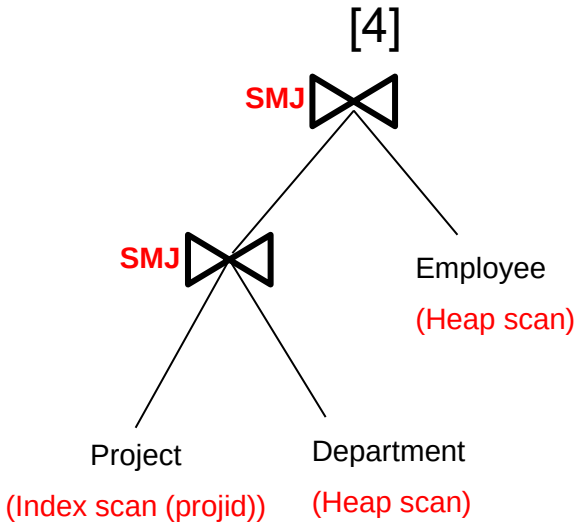
a) Compute the estimated result size and the reduction factors (selectivity) of this query.
(2 marks)

b) Compute the cost of the plans shown below. Assume that sorting of any relation (if required) can be done in 2 passes. NLJ is a Page-oriented Nested Loops Join. Assume that 100 tuples of a resulting join between Employee and Project fit in a page. Similarly, 100 tuples of a resulting join between Project and Department fit in a page. If selection over filtering predicates is not marked in the plan, assume it will happen on-the-fly after all joins are performed, as the last operation in the plan.
(8 marks, 2 marks per plan)









Formatting Requirements:

For each question, present an answer in the following format:

- Show the question number before each question. You do not need to include the text of the question itself.
- Show your answer in **blue** text (please type your answers on a computer).
- Start Question 2 and Question 3 on a new page.

For each of the calculations, provide all the formulae you used to calculate your cost estimates, and show your working, not only the result.

Submission Process:

Submit a single PDF showing your answers to all questions to the Assessment page on LMS by 6pm on the due date of Friday 18 October. Name your file 'STUDENT_ID'.pdf, where STUDENT_ID corresponds to YOUR student id.

Requesting a Submission Deadline Extension

If you need an extension due to a valid (medical) reason, you will need to provide evidence to support your request by *9pm, Thursday 17th of October.* Medical certificates need to be at least two days in length.

To request an extension:

1. Email Farah Khan (khanf1@unimelb.edu.au) from your university email address, supplying your student ID, the extension request and supporting evidence.
2. If your submission deadline extension is granted you will receive an email reply granting the new submission date. Do not lose this email! Replies may take up to 12 hours, so please be patient.

Reminder: INFO20003 Hurdle Requirements

To pass INFO20003 you must pass two hurdles:

- Hurdle 1: Obtain at least 50% (15/30) or higher for the three assignments (each worth 10%)
- Hurdle 2: Obtain a grade of 50% (35/70) or higher for the End of Semester Exam

Therefore, it is our recommendation to students that you attempt every assignment and every question in the exam.

GOOD LUCK!