Student ID:201447766

COMP207 Assignment 2 – Query Processing

Issue Date: Monday, 18 November 2019

Submission Deadline: Tuesday, 03 December 2019, 17:00

About This Assignment

This is the second of two assignments for COMP207. It is worth 10% of the total marks for this module. It consists of four questions, which you can find at the end of this document.

Submit your solutions to these questions in PDF format by the given submission deadline. Your solutions must be submitted on Vital (see the detailed submission instructions below).

Accuracy and relevance are more important in your answers, so don't write large volumes in your submission, but do ensure that what you write covers what is asked for and keeps to the problem statement.

Submission Details

Please submit **one PDF file with your solutions**. Name your file as follows:

<your student ID>-Assignment-2.pdf

If your student ID is 12345678, then your file should be named:

12345678-Assignment-2.pdf.

Please submit only this file (no archives).

To act as your 'signature' for the assignment, at the top of your PDF document put your Student ID number.

Your solutions must be submitted on Vital (see Vital for submission instructions).

The submission deadline for this assignment is **Tuesday, 03 December 2019, 17:00**. Earlier submission is possible, but any submission after the deadline attracts the standard lateness penalties. Plagiarism and collusion guidelines will apply throughout the assignment submission. For details on late submissions, how to claim extenuating circumstances, etc., please see the undergraduate student handbook, which can be found at http://intranet.csc.liv.ac.uk/student/ug-handbook.pdf , or in Section 6 of the Code of Practice on Assessment.[1]

Assessment information at a glance

Assignment Number	2 (of 2)
Weighting	10% of the final module mark
Assignment Circulated	Monday, 18 November 2019
Deadline	Tuesday, 03 December 2019, 17:00
Submission Mode	Electronically on Vital
Learning Outcome Assessed	LO2: Demonstrate an understanding of advanced SQL topics
Purpose of Assessment	Assessment of knowledge of SQL query processing
Marking Criteria	See description of this assignment
Submission necessary in order to satisfy module requirements?	N/A
Late Submission Penalty	Standard UoL Policy

[1] https://www.liverpool.ac.uk/media/livacuk/tqsd/code-of-practice-on-assessment/code_of_practice_on_assessment.pdf

Question 1 (10 marks)

The following tables form part of a hotel booking database held in a relational DBMS (primary keys are underlined):

Hotel (hotelNo, hotelName, city)
Room (roomNo, hotelNo, type, price)
Booking (hotelNo, guestNo, dateFrom, dateTo, roomNo)
Guest (guestNo, guestName, guestAddress)

- **Hotel** contains hotel details and hotelNo is the primary key.
- **Room** contains room details for each hotel and (roomNo, hotelNo) forms the primary key.
- **Booking** contains details of the bookings and (hotelNo, guestNo, dateFrom) forms the primary key.
- **Guest** contains guest details and guestNo is the primary key.

Give the relational algebra expressions to return the results for the following two queries:

(a) List the names and cities of those hotels who charge more than £85 for a room. **(5 marks)**

Ans:

$$\pi_{\text{hotelName, city}}(\sigma_{\text{price>85}}(\textbf{Hotel} \bowtie \textbf{Room}))$$

(b) List the names and addresses of guests who have made a booking to stay Christmas Day 2019. **(5 marks)**

Ans:

$$\pi_{\text{guestName, guestAddress}}(\sigma_{\text{dateFrom<='2019-12-25' AND dateTo>'2019-12-25'}}(\textbf{Booking} \bowtie \textbf{Guest}))$$

Question 2 (10 marks)

Consider the following database schema and example instance for a property management system:

property

pId	price	owner	sqrFeet	location
1	100,000	Alice	560	Lake View
2	3,400,000	Bob	2,000	Hyde Park
3	1,200,000	Bob	1,200	Hyde Park
4	5,000,000	Martha	800	Evanston

repairs

rId	pId	company	date	type
1001	1	M.M. Plumbing Ltd.	2013-12-12	Bathroom
1002	2	M.M. Plumbing Ltd.	2013-12-13	Kitchen
1003	4	Rob’s Double Glazing	2012-01-01	Windows

Hints:

- Attributes with a grey background form the primary key of a relation (e.g, *pId* for relation *property*).
- The attribute *pId* of relation *repairs* is a foreign key to relation *property*.

Give the relational algebra expressions to return the results for the following two queries:

(a) Get the pId, owner and location details of all properties that are larger than 900 square feet (sqrFeet). **(5 marks)**

Ans:

$$\pi_{\text{pId, owner, location}}(\sigma_{\text{sqrFeet}>900}(\textbf{property}))$$

(b) Get the names of repair companies (company) that did a repair on a property in Hyde Park. **(5 marks)**

Ans:

$$\pi_{\text{company}}(\sigma_{\text{location='Hyde Park'}}(\textbf{property} \bowtie \textbf{repairs}))$$

Question 3 (20 marks)

(a) Consider the following relation:
studentCourses(StudentID, CourseNo, Quarter, Year, Units, Grade)

The relation contains the grades for the courses completed by students. Assume that in studentCourses there are 200,000 different students, each identified by their StudentID. On average, a student took 40 different courses.

If the file blocks hold 2000 bytes and each studentCourses tuple requires 50 bytes, how many blocks will then be needed to store the relation studentCourses? **(5 marks)**

Ans:

200,000 * 40 *50 / 2000 = 200,000 blocks

(b) A database includes two relations Student (S) and Program (P).

S

Student_No	**F_Name**	**L_Name**	**Prog_Code**
04009991	Alicia	Smith	0001
04009992	Alan	Smith	0002
04009995	Alicia	Bush	0001
04009996	John	Smith	0001

P

Prog_Code	**P_Name**
0001	Computing
0002	Software Engineering

Give a relational expression that could possibly return the following result:

F_Name	L_Name	P_Name
Alicia	Smith	Computing
John	Smith	Computing

(5 marks)

Ans:

$$\pi_{F\_Name,\ L\_Name,\ P\_Name}\,(\sigma_{L\_Name='Smith'\ \text{AND}\ P\_Name='Computing'}\,(\mathbf{S} \bowtie \mathbf{P}))$$

(c) Translate the following relational algebra into SQL:

πstudId,lName(Ϭ course='BSc'(STUDENT)) **(5 marks)**

Ans:

SELECT studId, lName

FROM STUDENT

WHERE course='BSc';

(d) Given these relations, write the SQL statement that produced the equivalent queries below:

Course (courseNo, courseDept, courseLeader)

Student (studNo, name, type, tutorId, courseNo)

Two sample equivalent corresponding queries have been produced:

πstudno,name(Ϭ(type=’undergrad’)^(courseDept=’CompSci’)(Student⋈s.courseNo=c.courseNo Course))
and
πstudno,name(Ϭ type=’undergrad’(Student))⋈s.courseNo=c.courseNo(Ϭ courseDept=’Comp Sci’(Course))

(5 marks)

Ans:

(1)

```
SELECT studno, name
FROM (SELECT *
          FROM Student AS s, Course AS c
          WHERE s.courseNo=c.courseNo)
WHERE type=’undergrad’ AND courseDept=’CompSci’ ;
```

(2)

```
SELECT studno, name
FROM (SELECT *
           FROM Student AS s
           WHERE type=’undergrad’)
           JOIN
          (SELECT *
           FROM Course AS c
           WHERE courseDept=’Comp Sci’)
           ON s.courseNo=c.courseNo ;
```

Question 4 (60 marks)

Consider a database with relations $R(A, B, C)$, $S(D, E)$, and $T\,(F, G)$.

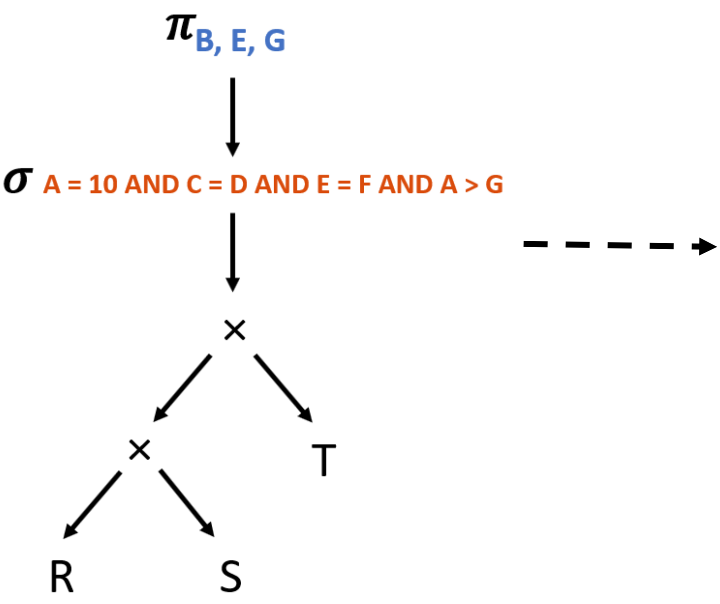
(a) Give the initial query plan (constructed as in Lecture 13) for the SQL query

```
SELECT B, E, G
FROM R, S, T
WHERE A = 10 AND C = D AND E = F AND A > G;
```

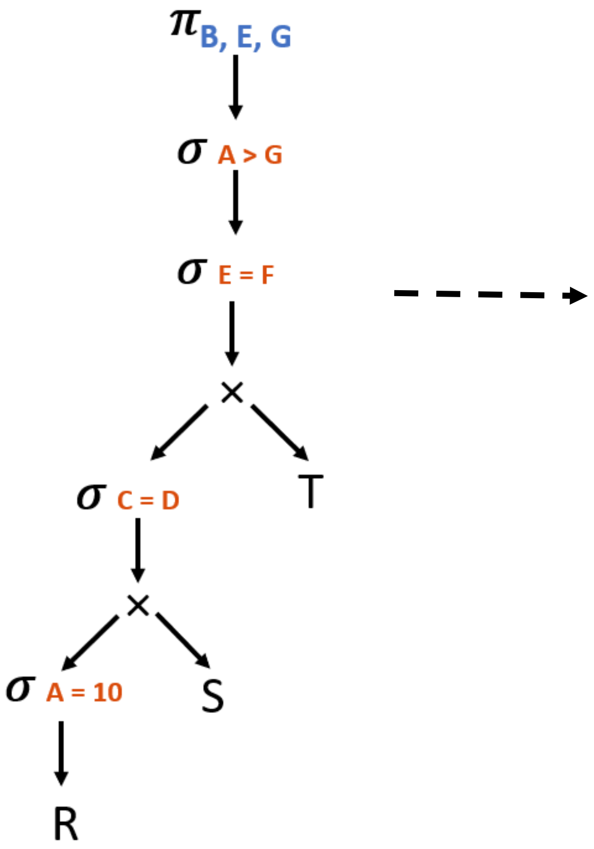
Then use the heuristics from Lecture 16 to transform the initial query plan into an optimised (logical) query plan. Perform the transformation step-wise, pushing a single operator over a single operator in each step, and indicate the heuristics you apply. **(20 marks)**

Ans:

Initial query plan

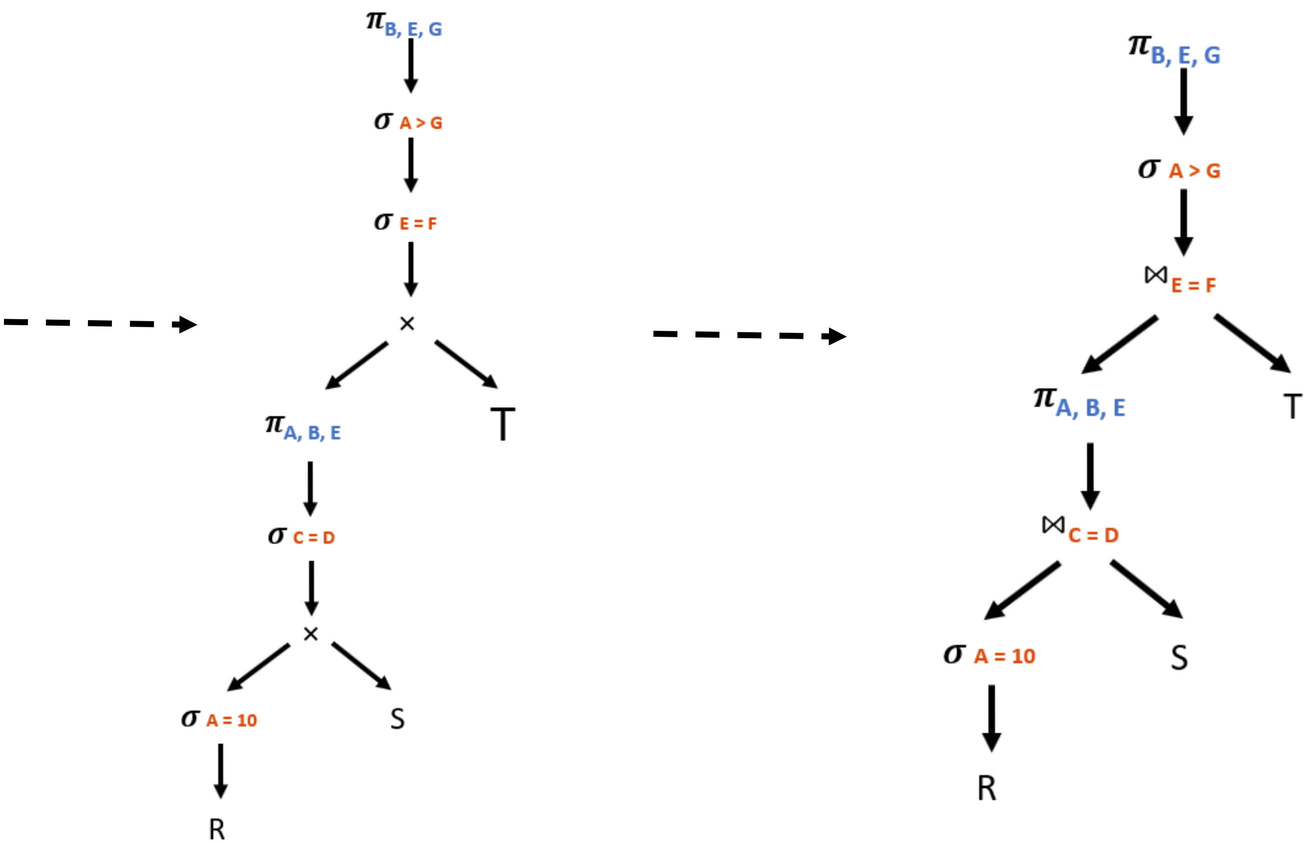


#1: Push selections as far down the tree as possible



#2: "Push" projections as far down the tree as possible,or insert projections where appropriate

#3: If possible, introduce equijoins for × followed by σ



(b) Suppose that

- $|R| = 1000$, $|\pi_A(R)| = 1000$, $|\pi_B(R)| = 100$, $|\pi_C(R)| = 500$;
- $|S| = 5000$, $|\pi_D(S)| = 300$, $|\pi_E(S)| = 10$;
- $|T| = 4000$, $|\pi_F(T)| = 4000$, $|\pi_G(T)| = 1500$.

Estimate the number of tuples returned by the following queries. Explain your calculations.

i) $\sigma_{A=10}(R)$ **(6 marks)**

Ans:

Suppose *U*= σA=10(R),

T(U) = T(R) / V(R, A)

= |R| / |πA(R)|

= 1,000/ 1,000

= 1

ii) $\sigma_{A=10 \text{ OR } B=b}(R)$ **(6 marks)**

Ans:

Suppose *V*= $\sigma_{A=10 \text{ OR } B=b}(R)$

T(V) = ***T(R) / V(R, A) + T(R) / V(R, B)***

=|R| / |πA(R)| + |R| / |πB(R)|

=1 + 10

=11

iii) $R \bowtie_{C=D} S$ **(6 marks)**

Ans:

Suppose $W = R \bowtie_{C=D} S$

T(W) = T(R) T(S) / max (V(R, C), V(S, D))

$= |R| * |S| / \max(|\pi_C(R)|, |\pi_D(S)|)$

= 1,000 * 5,000 / 500

= 10,000

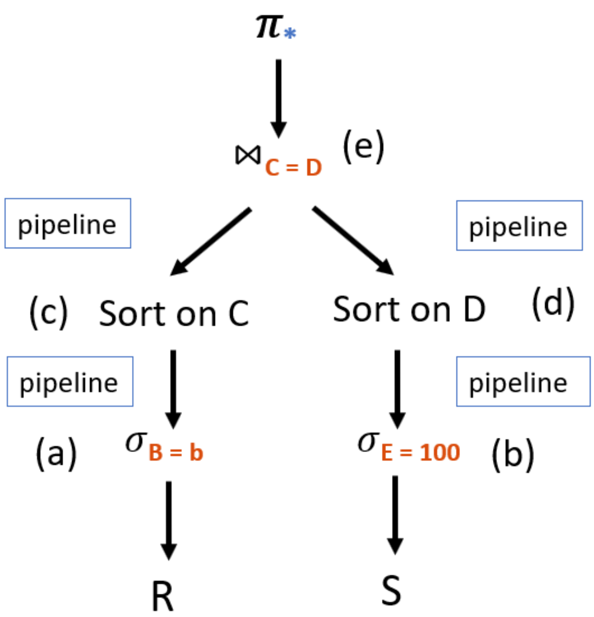
(c) Suppose that in addition to the assumptions on *R*, *S*, and *T* from part (ii), we also have the following:

- Each disk block can hold up to 10 tuples.
- All relations are stored in consecutive blocks on disk.
- No indexes are available.

What is the best physical query plan (in terms of the number of disk access operations) you can find for $\sigma_{B=b \text{ AND } E=100}(R \bowtie_{C=D} S)$? Describe your plan and show the calculation of the number of disk access operations. **(22 marks)**

Ans:

The best physical query plan for $\sigma_{B=b \text{ AND } E=100}(R \bowtie_{C=D} S)$:



Firstly, select B=b and E=100 from R and S. Pipeline is used here too reduce the disk access operations. Then before join, the selected tuples are sorted on C and D using merge sort algorithm. Finally, sorted tuples will be joined together to reach the query plan.

Total number of disk access operations (Suppose the RAM size=M)

= (*T(R) + T(R) / V(R, B)*) (a)

+ *T(S) + T(S) / V(S, E)*) (b)

+ (T($\sigma_{B=b}$(R)) * T($\sigma_{E=100}$(S)))/max(T($\sigma_{B=b}$(R), C) , T($\sigma_{E=100}$(S), D)) (e)

) / 10

+ O(N_c/B * $\log_M$(N_c/(MB))) (c)

+ O(N_D/B * $\log_M$(N_D/(MB))) (d)

= (1000 + 1000/100 + 5000 + 5000 / 10 + 10 * 500 / max(5, 30)) / 10+O($\log_M$(1/M))+O(50 $\log_M$(50/M))

$= (1010 + 5500 + 5000 / 30) / 10 + O(\log_M(1/M)) + O(50 \log_M(50/M))$
$= 668 + O(50\log_M(50/M))$

Thus total number of disk access operations is $668 + O(50\log_M(50/M))$.