Xi'an Jiaotong-Liverpool University

PAPER CODE	EXAMINER	DEPARTMENT	TEL
CSE201		Computer Science & Software Engineering	

FIRST SEMESTER 2017/2018 Resit EXAMINATIONS

BACHELOR DEGREE – Year 3

DATABASE DEVELOPMENT AND DESIGN

TIME ALLOWED: 2 Hours

INSTRUCTIONS TO CANDIDATES

1、 **Total marks available are 100. This will count for 100% in the final assessment.**

2、 **Answer four questions.**

3、 **The number in the column on the right indicates the marks for each section.**

4、 **Answer should be written in the answer booklet(s) provided.**

5、 **The university approved calculator - Casio FS82ES/83ES can be used.**

6、 **All the answers must be in English.**

THIS PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM

Question 1. Answer the following questions on indexing in database systems.

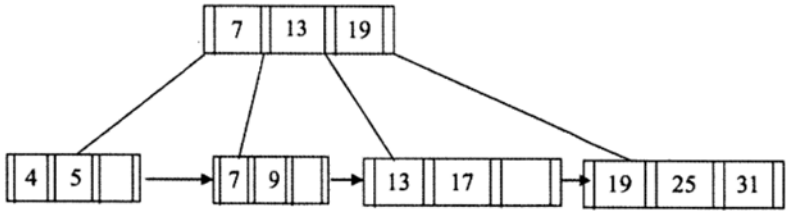
[25 marks]

a) Relation *R* has 1,000 fixed length, fixed format records. One disk block can hold 5 records and no record spans over one block. Suppose that B+ tree is used to index the tuples based on the candidate key. The number of pointers of the B+ tree node, N=10. Assume a dense index has one index entry for each tuple, and a sparse index has one index entry for each block. What is the maximum height for the dense index? What is the maximum height for the sparse index?

[4/25]

b) Consider the following B+ tree. The number of pointers that fits in one node is 4. Draw the trees after inserting three search key values 17, 33 and 11.

[9/25]



c) Based on the result from Question 1.b), delete search key value 25.

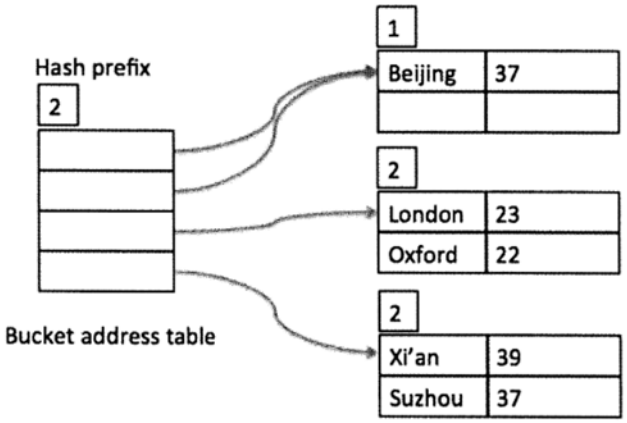
[4/25]

d) Briefly state the advantage(s) of static hashing for indexing in database systems.

[4/25]

e) Consider the following diagram of an extendable hash index built on city names (e.g., Beijing and Suzhou). Suppose another tuple with search key value "Suzhou" is inserted. Describe in detail how the hash index would be updated.

[4/25]



Question 2. Consider the following three relations:

i) *author (authorID, name, affiliation, email)*
ii) *publishes (articleID, authorID)*
iii) *article (articleID, title, journal, year, publisher)*

"*authorId*" and "*articleID*" are the primary keys for relations *author* and *article*, respectively. *authorID* is the join attribute for *author* and *publishes*. *articleID* is the join attribute for *publishes* and *article*. Relation *author* has 5,000 tuples stored on 500 blocks. Relation *publishes* has 100,000 tuples stored on 100 blocks. Answer the following questions.

[25 marks]

a) Describe how the selection $\delta_{year>2014}$ on relation *article* can be evaluated in the most cost effective way by using a primary index on year and a secondary index on year, respectively.

[4/25]

b) Assume that the memory can only hold one block for each relation. Using the blocked nested loop join algorithm and *publishes* as the outer relation, how many block transfers and seeks are needed to compute "*author* $\bowtie$ *publishes*", respectively?

[4/25]

c) Assume that the memory can hold all blocks of the smaller relation *publishes*. Using the blocked nested loop join algorithm, how many block transfers and seeks are needed to compute "*author* $\bowtie$ *publishes*", respectively?

[6/25]

d) Relations have to be sorted first before the merge join algorithm can be used. Assume that the memory size $M=10$, and the buffer for input and output, $b_b=1$. How many block transfers and seeks are needed to sort the relation *author* by using the external sort merge algorithm?

[6/25]

e) Assume that the hash join algorithm is used to evaluate "*author* $\bowtie$ *publishes*". Is the cost (in terms of number of block transfers and seeks) of using *author* as the build input less than the one of using *publishes* as the build input? Suppose neither of the entire build input can be kept in memory. Justify your answer.

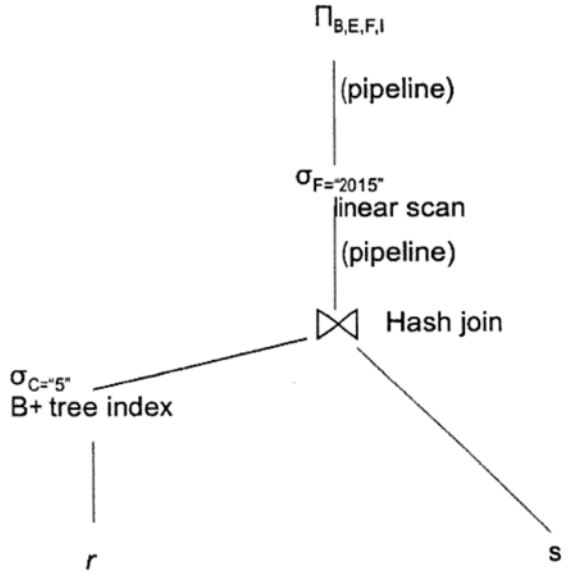
[5/25]

Question 3. Consider two relations *r*(A, B, C, D, E) and *s*(A, F, G, H, I) with a common attribute A. The catalog information is given as follows.

[25 marks]

- number of tuples in relation *r*, $n_r = 1{,}000$
- number of blocks in *r*, $b_r = 200$
- index: a primary B+-tree index of height 4 on attribute *r*.C
- number of distinct values on attribute C in *r*, V(C, *r*) = 50
- number of distinct values on attribute A in *r*, V(A, *r*) = 5
- number of tuples in relation s, $n_s = 10{,}000$
- number of blocks in relation s, $b_s = 1{,}000$
- number of distinct values on attribute A in *s*, V(A, *s*) = 100

A query evaluation plan is shown below.



a) What are the differences between materialisation and pipelining in evaluating an expression?

[4/25]

b) Briefly explain how heuristics can be used to optimise the evaluation plan.

[4/25]

c) Based on the catalog information, what is the estimated size of the selection $\sigma_{C="5"}$?

[4/25]

d) What is the number of block transfers in evaluating $\sigma_{C="5"}$ using the B+ tree index?

[5/25]

e) Assume the indexed nested-loop join is done with the hash index on relation *s*, and for each tuple from $\sigma_{C="5"}$, the average cost of fetching all matching records from the hash index is 10 block transfers. What is the total number of block transfers for the whole evaluation plan?

[8/25]

Question 4. Answer the following questions related to transaction, concurrency and recovery in database systems.

[25 marks]

a) What are meant by the ACID properties in the context of database transaction management?

[4/25]

b) Draw the precedence diagram for the schedule below and determine if it is conflict serialisable.

T1	T2	T3	T4
Read(x)			
Write(x)			
Read(y)			
	Read(x)		
		Write(y)	
	Read(z)		
			Read(y)
			Read(w)
			Write(w)
	Read(w)		

[4/25]

c) What is meant by "a schedule is recoverable"? Is the schedule in Question 4.b) recoverable" and why?

[5/25]

d) Briefly explain the difference between the precedence graph and wait-for graph in the context of transaction management.

[4/25]

e) Consider the following schedule which uses the recovery algorithm with redo/undo operations and checkpoints. If the database failure happens immediately after time=112 (before 113), which transactions need to be redone and which need to be undone? Justify your answer.

Time	T7	T10	T13
100			start
101			read(Y)
102			Y=Y+100
103	start		
104	read(X)		
105	X=X+300		
106			write(Y)
107			commit
108		start	
109		read(X)	
110		read(Y)	
111		Y=Y+X	
112		write(Y)	
113		commit	
114	read(Y)		
115	Y=Y+X		
116	write(X)		
117	----------	---Checkpoint ----	----------
118	commit		

[8/25]

Question 5. Answer the following questions.

[25 Marks]

a) In the context of distributed database systems, what is the two-phase commit protocol used for? Briefly describe how the protocol works during the two phases.

[5/25]

b) Briefly describe how the wait-for graphs are used to detect deadlocks in distributed database systems.

[6/25]

c) Consider the three relations in Question 2. Assume that they are stored in a distributed database at two sites:
University of Liverpool site:
author
publishes
ACM site:
article

To answer the query "*find the names of University of Liverpool staff and titles of articles that were published by them*", one needs to compute the join of the three relations. It is known that only a small number of articles stored at the ACM site were published by authors stored at the University of Liverpool site. Assume that a query is made to the University of Liverpool site; describe how bloom-join can be used to optimise the query (e.g., reduce cost).

[8/25]

d) Consider the following database for customer transactions and answer the following questions:
(1) What is the support for the itemset {X,U} if each transaction is treated as a market basket?
(2) What is the support for the itemset {X,Y,Z} if each transaction is treated as a market basket?
(3) What is the confidence for the association rule "X,Z→Y" if each transaction is treated as a market basket?
(4) What is the support for the itemset {X,Y,Z} if each customer is treated as a market basket?
(5) What is the confidence for the association rule "X,Z→Y" if each customer is treated as a market basket?
(6) Find the associate rule with the confidence of 100%. Assume that each customer is treated as a market basket and that the left-hand side of the rule must be "U,W".

Customer ID	Transaction ID	Items
418	1	{X, Z}
345	2	{U, V, W, X, Y, Z}
323	3	{U, W, Y}
418	4	{V, X, Z}
567	5	{U, Y}
567	6	{W, X, Y}
323	7	{X, Y, Z}
635	8	{U, Z}
345	9	{V, Y}
635	10	{V, W, X}

[6/25]

END OF EXAM PAPER