浙江大学 2018–2019 学年 春夏 学期

《数据库系统》课程期末考试试卷

课程号：_____， 开课学院：_____

考试试卷：√A 卷、B 卷（请在选定项上打√）

考试形式：√闭、开卷（请在选定项上打√）,允许带一张 A4 纸笔记入场

考试日期：2019 年 7 月 1 日,考试时间：120 分钟

诚信考试，沉着应考，杜绝违纪。

姓名：_____学号：_____所属院系：_____授课教师：_____

题序	一	二	三	四	五	六	七	八	总 分
得分									
评卷人									

Problem 1: Relational Model and SQL (16 points, 4 points each)

Consider the following relational schemas with the primary keys underlined.

Movie(title, type, director)

Comment(title, user_name, grade)

1) Write a *relational algebra expression* to find all the movie titles that are directed by "Yimou Zhang" and exist the comment grade of greater than or equal to 4.

2) Write a *SQL statement* to change the null value of grade to 0.

3) Write a *SQL statement* to find which movies have the highest average grade.

4) Write a SQL statement to find all the movie titles where every user gives higher grade than movie "the avenger".

Answers of Problem 1：

Problem 2: E-R Model (9 points)

We need to store information about football teams in league matches. The information includes matches (identified by match_id, with attributes location, time) as well as the score of each team for the match, teams (identified by name, with attribute city), players (identified by name, with attributes age and several phone numbers), and individual player statistics (such as score, shooting, foul, etc.) for each match. Note that one player only belongs to one team.

Please answer the following questions:

1) Draw an *E-R diagram* for this database model with primary key underlined. (5 points)

2) Transform the E-R diagram into a number of *relational database schemas*, with the primary key underlined. (4 points)

Answers of Problem 2：

Problem 3: Relational Formalization (12 points, 4 points each)

For relation schema R(A, B, C, D, E) with functional dependencies set F={A→B, BC→D, C→A}

1) Find all *candidate keys* of R.

2) Decompose the relation R into a collection of *BCNF relations*.

3) Explain whether above decomposition be *dependency preserving* or not.

Answers of Problem 3：

Problem 4: XML (12 points, 4 points each)

Following is an example XML document describing the information in **Problem 1**.

```
<movie_comment>
    <movie title=”wandering earth”>
        <type>science fiction</type>
        <director> Fan Guo </director>
        <comment>
            <user name>Alice</user name>
            <grade>5</grade>
        </comment>
        <comment>
            <user_name>Bob</user_name>
            <grade>4</grade>
        </comment>
    </movie>
    <movie title=”the avenger”>
        <type>action</type>
        <director> Joss Whedon </director>
        <comment>
            <user_name>John</user_name>
            <grade>4</grade>
        </comment>
    </movie>
</movie_comment>
```

1) Give the *DTD* for the XML representation, requiring that at least one comment for each movie.

2) Give a *Path expression* to find all the action movie titles where Alice gives grade 5.

3) Give an *XQuery expression* to find all the movie titles that are directed by “Yimou Zhang” and have at least one grade 5.

Answers of Problem 4：

Problem 5: B^+-Tree (12 points, 3 points each)

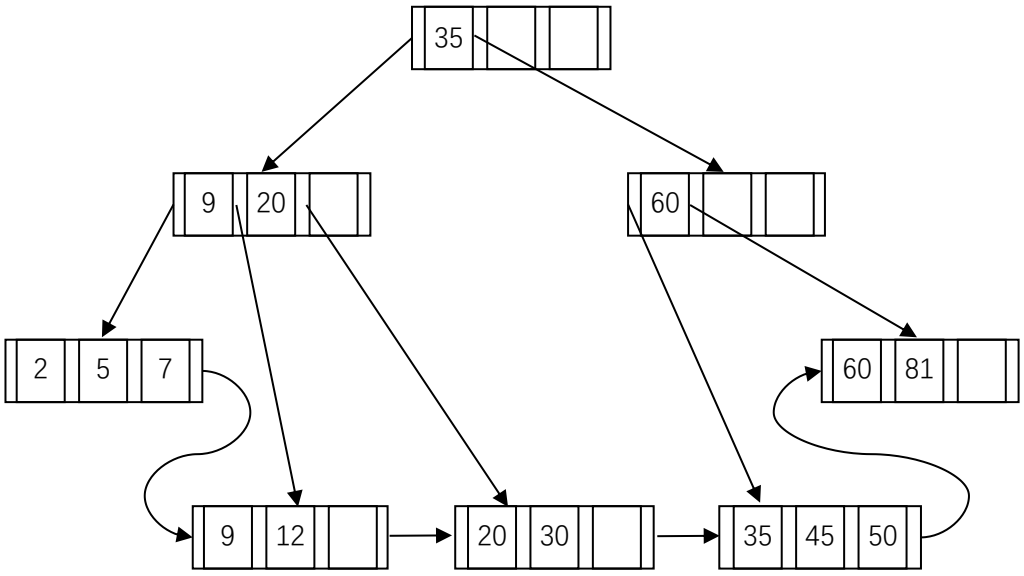
For the following B^+-tree (n = 4), answer following questions:

1) Show the structure of the tree after inserting sequentially 8, 6, and 3 into the original tree.

2) Show the structure of the tree after deleting sequentially 81 and 45 from the original tree.

3) If the height of tree is 5, please tell the minimal and maximal numbers of key values in the tree.

4) Assume that (a) there are 3 blocks in main memory buffer for B^+-tree operation, and at first those blocks are empty; (b) the Least Recently Used (LRU) strategy is used for buffer replacement; and (c) each node of B^+-tree occupies a block. Please count the number of B^+-tree blocks transferred to buffer in order to complete the operations in 2) of this problem.



Answers of Problem 5：

Problem 6: Query Processing (12 points, 4 points each)

For the relational schemas in problem 1, there are following assumptions:

Number of tuples, movie: 5,000, comment:1,000,000;

Blocking factor, movie:50, comment:100;

Number of distinct values, V(director, movie) = 500, V(grade, comment) = 5;

Block size is 4K bytes;

Movie has a B^+-tree index on title, and each index block contains 60 entries (i.e. n=60)

Answer following questions (*We do not consider cost to write output to disk*):

1) Estimate the size of the result of 1) in **Problem 1**.

2) Suppose that the *block nested-loop join* is used to implement movie $\bowtie$ comment, buffer in main memory has 12 blocks, and movie is chosen as inner relation. In order to obtain the best performance, how to assign buffer blocks to movie and comment respectively? Please estimate the number of block accesses and the number of seeks required by the solution.

3) Suppose that the *index nested-loop join* is used to implement $\sigma_{\text{director="Yimou Zhang"}}$ (movie) $\bowtie$ comment. Please estimate the number of block accesses and the number of seeks required by the solution (assume the worst case of memory, and that one extra buffer block is used for the root index block and it needs to be read from disk only once).

Answers of Problem 6：

Problem 7: Concurrency Control (12 points, 4 points each)

Consider following three transactions:

T1: read(A)
Read(B)
Write(B)

T2: read(B)
Read(A)
Write(A)

T3: read(A)
Write(A)

1) For the following schedule, please draw the precedence graph, and explain whether it is conflict serializable.

T1	T2	T3
Read(A)		
Read(B)		
		Read(A)
		Write(A)
	Read(B)	
	Read(A)	
Write(B)		
	Write(A)	

2) For the following schedule, please explain whether it is cascadeless.

T1	T2	T3
Read(A)		
Read(B)		
Write(B)		
	Read(B)	
	Read(A)	
	Write(A)	
		Read(A)
Abort		

3) Please explain whether the two-phase locking protocol can be used to implement the schedule in 1).

Answers of Problem 7：

Problem 8: Aries Recovery Method (15 points, 3 points each)

A DBMS uses Aries algorithm for system recovery. Following figure is a log file just after system crashes. The log file consists of 14 log records with LSN from 1001 to 1014. The figure does not show PrevLSN and UndoNextLSN in log records. Assume that last completed checkpoint is the log record with LSN 1008.

1001: <T1, begin>
1002: <T1, 101.1, 11, 21>
1003: <T2, begin>
1004: <T2, 102.1, 52, 62>
1005: <T2, commit>
1006: <T3 begin>
1007: <T3, 102.2, 73, 83>
1008: checkpoint

Tx	LastLSN
T1	1002
T3	1007

PageID	PageLSN	RecLSN
101	1002	1002
102	1007	1004

1009: <T1, 101.2, 31, 41>
1010: <T4 begin>
1011: <T3, 102.2, 73>
1012: <T3, abort>
1013: <T4, 102.1, 62, 64>
1014: <T1, commit>

Please answer following questions：
1) Which log record is the start point of Redo Pass?
2) Which log record is the end point of Undo Pass?
3) After Analysis Pass, what is the undo list?
4) After recovery, what is the value of data items identified by “102.1” and “102.2”, respectively?
5) What additional log records are appended to log file during recovery?

Answers of Problem 8：