Student ID ……..

VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wananga o te Upoko o te Ika a Maui

EXAMINATIONS — 2003
MID-YEAR

COMP 302

Database Systems

Time allowed: 3 Hours

Instructions: Answer all questions.
Make sure that your answers are clear and to the point.
Calculators and foreign language dictionaries are allowed.
No reference material is allowed.
There are 180 marks on the exam.

CONTENTS:

Question 1. SQL and Relational Algebra [25 marks]

Consider the following part of a relational database schema for recording data from two linguistics experiments on some subjects using a collection of paragraphs and a collection of sentences. One experiment measures the reading time of the subjects on each paragraph, and the other measures the number of words that the subjects can recall from each sentence.

Subject: {{SubjId, Gender, Age},{SubjId}}
ParaExp: {{SubjId, ParagraphNum, Time}, {SubjId+ParagraphNum}}
SentExp: {{SubjId, SentenceNum, WordCount}, {SubjId+SentenceNum}}

Although there are some other relation schemes in the database schema, only these three are needed for the intended queries.

Details of the ParaExp relation are given in the table.

Attribute	Data Type	Max. Length	Null
SubjId	Int	4	No
ParagraphNum	Int	2	No
Time	Bigint	8	No

(a) Write an SQL statement that would define the ParaExp table. Include any referential integrity constraints. Deleting a subject from the Subject table should delete all their experimental results, and changing the SubjId of a subject should not be allowed if there are any experimental results for that subject. [3 Marks]

ANSWER

(b) Write SQL queries for each of the following:

(i) Which subjects took more than 150 seconds to read any paragraph? [3 Marks]

ANSWER

(ii) Which female subjects read any paragraph in less than 100 seconds and remembered at least 15 words from any sentence? [3 Marks]

ANSWER

(iii) List, in increasing order, the average reading time of the paragraphs. [5 Marks]

ANSWER

(iv) List, in increasing order, the reading times on the paragraph experiment of the subject(s) who recalled the most words from sentence 8? [8 Marks]

ANSWER

(c) Write a Relational Algebra expression for a query that retrieves the paragraph numbers of the paragraphs read by female subjects in less than 100 seconds [3 Marks]

ANSWER

Question 2. Enhanced Entity Relationship Data Model [25 marks]

A large legal company wants to construct a database to record all the information about the activities of the company. You are to construct an EER diagram that models the part of the company's activities described below. Show all the participation and cardinality constraints (using one of the notations presented in the course). If there are any constraints that cannot be expressed in the EER diagram, write them in English below the diagram. If the description does not specify information you need to know, make a reasonable assumption and state the assumption you have made.

- The staff of the company consists of a number of lawyers and a number of legal assistants. The number of staff is small enough that they can identify the staff by *name*. Each staff member has a *role* (lawyer or assistant) and a *rate* ($ per hour). The lawyers each have a *specialty*, and some of the lawyers are *partners*, while others are not.
- The primary activity of the company is legal cases for clients. Each active case is identified by a *CaseID* number, and involves a particular *client*.
- At least one lawyer, and possibly more, will work on each case, and each lawyer may work on several cases (or on none). Every *day*, they record how much *time* they have worked on each case.
- Exactly one lawyer is assigned to be in charge of each case. Not all lawyers are in charge of a case.
- The legal assistants are assigned to cases, but, unlike the lawyers, each assistant is assigned to at most one case. A case may have more than one legal assistant assigned to it, and some cases have no legal assistants at all.
- There are several different courts in the country. They are identified by their *level*, (magistrate, superior, appeal), and their *district*.
- The court rooms where trials take place are identified by which court they belong to and a *number*, for example, "Wellington magistrates court, #3". Each court room also has an *address*.
- Judges are identified by their names. Each judge has a *reputation* for how they conduct trials.
- Each case will be tried by one court, though each court may be trying many cases, and the company may or may not have cases with all the different courts.
- The trial of a case must involve at least one, and usually many appearances. Each appearance involves a judge at a particular courtroom at a particular time (specified by the *day* and the *hour*). The different appearances of a case may have different judges and different courtrooms, as long as they are different courtrooms of the same court. There can only be one appearance in any courtroom at any given day and time. Courtrooms and judges are not engaged in an appearance at every time, and some judges and courtrooms are not involved in any of the cases of this company.

ANSWER

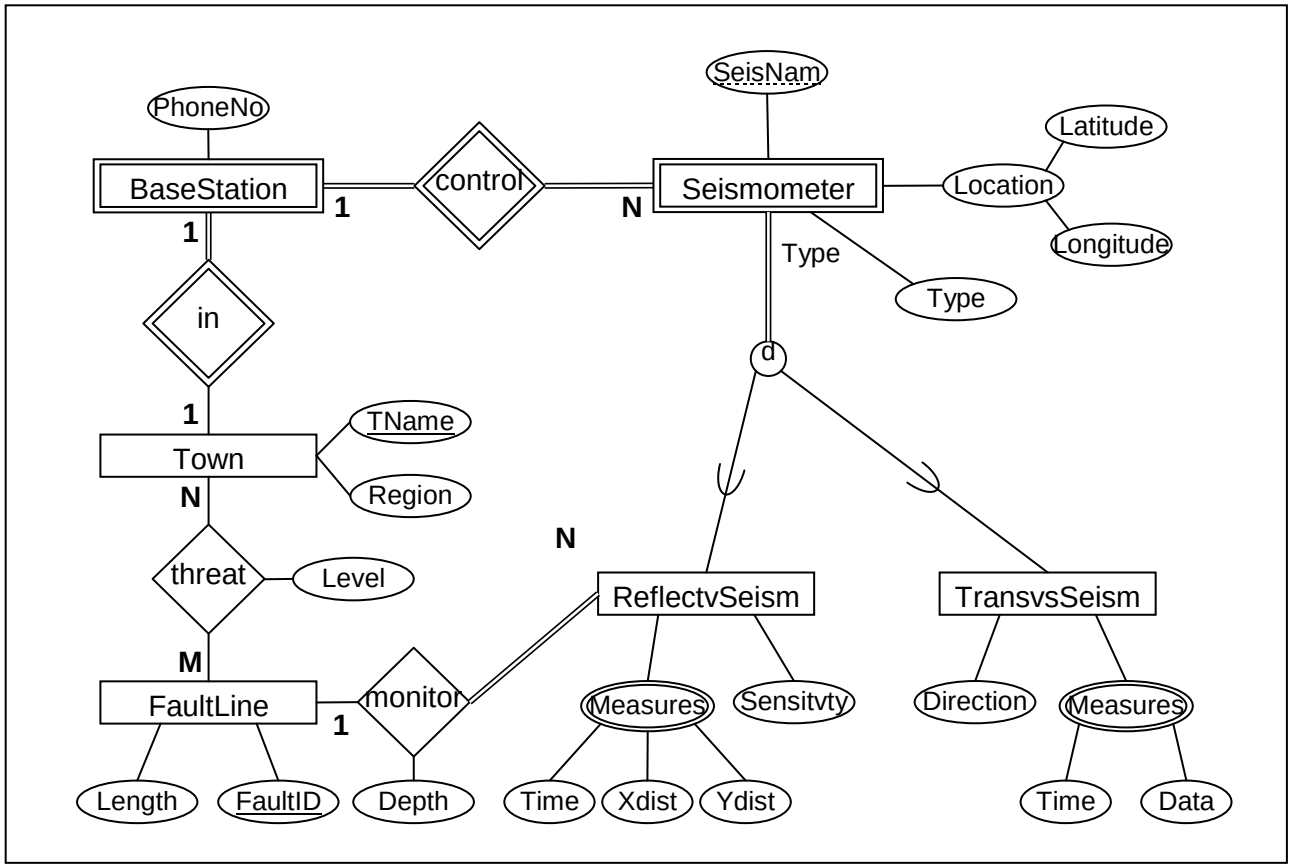
Question 3. Mapping EER to relational data model [30 marks]

The figure below shows an EER model for an database for an earthquake monitoring service. The database records the various base-stations owned by the service. Each base-station is located in a town that is threatened by some known fault lines, and each town has at most one base-station. A base-station has a collection of seismometers (always identified by "A", "B", "C",…) located near the base-station that report their measurements every 10 seconds to the base-station by radio. The base station forwards the measurements by phone line to the central database. There are two kinds of seismometers (transverse and reflective) that make different measurements. Reflective seismometers are placed to monitor a particular fault at a certain depth.

(a) Map the EER diagram below into a relational database schema: [18 Marks]

- For each relation schema, list the attributes of the relation and a primary key. You do not need to specify the domains of the attributes.
- List a set of non-redundant referential integrity constraints.
- State the highest normal form that your relation database schema is in, and explain why.

Note that the partial key of the BaseStation type is empty.



ANSWER

(b) The "monitor" and "control" relationships in the diagram require total participation of one or both of the entity types involved. Explain how your relational database schema enforces these constraints, if it does. Otherwise, suggest how the constraints could be enforced. [6 Marks]

ANSWER

(c) If your relational database schema enforces the constraint that the specialization of Seismometer is total, explain how it enforces the constraint. If not, give two different examples of how adding or deleting tuples from the database could leave the database in an inconsistent state.

[6 Marks]

ANSWER

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.

Specify the question number for work you do want marked.

Question 4. Functional Dependencies and Normalization [40 marks]

(a) Consider the following set of functional dependencies (fd's):

$$F_1 = \{A \rightarrow BC,\ CDEG \rightarrow DEG,\ G \rightarrow G\ \}.$$

Apply the decomposition inference rule on the set F_1 to produce a new set of fd's F_1', where each fd has only one attribute on its right hand side. [3 marks]

ANSWER

(b) Consider the following set of functional dependencies:

$$F_2 = \{AB \rightarrow C,\ DEG \rightarrow H,\ A \rightarrow C,\ DG \rightarrow J,\ J \rightarrow H\ \}.$$

Transform the set F_2 into a new set of fd's F_2', in which each fd is left reduced (ie has no redundant attributes on its left hand side). [6 marks]

ANSWER

(c) Consider the following set of left reduced functional dependencies:

$$F_3 = \{AB \rightarrow C,\ C \rightarrow D,\ C \rightarrow B,\ D \rightarrow B,\ AB \rightarrow D,\ AD \rightarrow C\ \}.$$

Transform the set F3 into a new, non redundant set of fd's F_3'. [8 marks]

ANSWER

(d) Consider the following set of left reduced and non redundant functional dependencies:

$$F_4 = \{AB \rightarrow C, AB \rightarrow D, AB \rightarrow E, B \rightarrow L, B \rightarrow G, B \rightarrow H, A \rightarrow J\}.$$

Partition the set F_4 into subsets G_i, $i = 1, 2, \ldots$, where each subset contains only those functional dependencies from F_4 that contain the same left hand side. [2 marks]

ANSWER

Build a set S of relation schemas by transforming each subset G_i into a relation schema of the form $N_i(R, K)$, where N_i is the relation schema name, R is the set of attributes, and K is the set of keys. [2 marks]

ANSWER

(e) What is the highest normal form that the set of relation schema S of question 4(d) is in? [2 marks]

ANSWER

(f) Suppose (U, F) is a universal relation schema, where

$U = \{A, B, C, D, E, G\}$ and $F = \{AB \rightarrow C, AC \rightarrow E, AC \rightarrow G, C \rightarrow D, C \rightarrow B, D \rightarrow B\}$.

Suppose that starting from (U, F) the following set

$S = \{N_1(\{A, B, C\}, \{AB\}), N_2(\{C, D\}, \{C\}), N_3(\{D, B\}, \{D\}), N_4(\{A, C, E, G\}, \{AC\})\}$

of 3NF relation schemas has been produced.

If you consider that S is a lossless join decomposition of (U, F) explain why it is.
If you consider that S is not a lossless join decomposition of (U, F), explain why it is not, and transform it into a new set of relation schemas S' that will be at least a 3NF and lossless join decomposition of (U, F). [6 marks]

ANSWER

(g) Questions (a) to (f) represent the steps of a normalization algorithm. What is the name of that algorithm? [1 mark]

ANSWER

(h) Consider the following set of relation schemas (the same as in question (f))

$$S = \{N_1(\{A, B, C\}, \{AB\}), N_2(\{C, D\}, \{C\}), N_3(\{D, B\}, \{D\}), N_4(\{A, C, E, G\}, \{AC\})\}$$

Suppose all functional dependencies in S are a consequence of relation scheme keys, transform S into a set of relation schemes S'' by applying the following algorithm: [5 marks]

Input: S, $F = \{X \rightarrow A \mid (\exists N_i(R_i, K_i) \in S)(X \in K_i \wedge AX \subseteq R_i\}$
Output: S''

$S'' := S$ //Initialization
while $(\exists N_i(R_i, K_i), N_j(R_j, K_j) \in S'')(i \neq j \wedge (X_i)^+_F = (X_j)^+_F)$ {
// Merge $N_i(R_i, K_i)$ and $N_j(R_j, K_j)$ into $N_i'(R_i', K_i')$
$R_i' := R_i \cup R_j$
$K_i' := K_i \cup K_j$
// Replace $N_i(R_i, K_i)$ and $N_j(R_j, K_j)$ with $N_i'(R_i', K_i')$ in S''
$S'' := ((S'' \setminus \{N_i(R_i, K_i)\}) \setminus \{N_j(R_j, K_j)\}) \cup \{N_i'(R_i', K_i')\}$
}

ANSWER

(i) What is the highest normal form that the set of relation schema S'' of question (h) is in? Justify your answer. [5 marks]

ANSWER

Question 5. Query Optimisation [40 marks]

Consider the following description of a part of the *LINGUISTIC_EXPERIMENT* database schema, given in Table 1, and the corresponding parameters of the physical database structure, given in Table 2. In Table 2, L is the tuple length (size), f is the blocking factor, B is the block size, r is the number of tuples, and x is either the primary, or secondary key index height. The values of all the Table 2 parameters pertain to the base relations. Note that the same part of the database is also considered in Question 1.

Attribute	Data Type	Max. Length	Null	Default
Relation name: *Subject*, Primary Key: *SubjId*				
SubjId	Int	4	N	
Gender	Char	1	N	
Age	Int	2	N	
Relation name: *ParaExp*, Primary Key: *SubjIdId* + *ParagraphNum*				
SubjId	Int	4	N	
ParagraphNum	Int	2	N	
Time	Bigint	8	N	
Relation name: *SentExp*, Primary Key: *SubjId*+ *SentenceNum*				
SubjId	Int	4	N	
SentenceNum	Int	2	N	
WordCount	Int	2	N	

Table 1

Relation schema	*L*	*f*	*B*	*r*	*x* (primary)	*x* (secondary)
Subject	*7*	*71*	*497*	*13000*	*2*	*1*
ParaExp	*14*	*35*	*490*	*65000*	*3*	-
SentExp	*8*	*62*	*496*	*130000*	*3*	-

Table 2

For parts (b) to (e), suppose:

- Each subject may have an age between 10 and 69 years (60 different values), and there are an approximately equal number of subjects of every age,
- Each subject was asked to read 10 sentences,
- The intermediate results of the query evaluation are materialized,
- The final result of the query is materialized,
- The size of each intermediate result block should not exceed 500 bytes,
- There is a buffer pool of 6000 bytes provided for query processing in the main memory,
- There are primary indexes on *Subject.SubjId*, *ParaExp.*(*SubjId* + *ParagraphNum*), and *SentExp*.(*SubjId* + *SentenceNum*) available, and
- There is a secondary index on *Subject.Age* available

NOTE:
Some of the formulae you may need when computing the estimated query costs are given at the end of this exam paper.

(a) Draw the heuristic optimization tree that corresponds to the SQL query: [8 marks]

SELECT s.SubjId, Age, ParagraphNum, Time FROM (Subject s NATURAL JOIN ParaExp p) WHERE Age >= 55;

ANSWER

(b) Calculate the lowest execution cost of the query [5 marks]

SELECT * FROM Subject WHERE Age >= 55;

ANSWER

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.

Specify the question number for work you do want marked.

(c) Calculate the lowest execution cost of the query [5 marks]

```
SELECT * FROM Subject ORDER BY SubjId;
```

ANSWER

(d) Calculate the lowest execution cost of the query [10 marks]

```
SELECT * FROM Subject s, ParaExp p
WHERE s.SubjId = p.SubjId;
```

ANSWER

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.

Specify the question number for work you do want marked.

(e) Draw a heuristic optimization tree and calculate the lowest execution cost of the query

```
SELECT SubjId, SUM(WordNum) FROM SentExp
GROUP BY SubjId;
```

[12 marks]

ANSWER

Question 6. Concurrency Control [20 marks]

Suppose a DBMS supports an explicit lock table command with the syntax

LOCK [TABLE] <*table_name*> IN <*lock_mode*> MODE

where *lock_mode* is one of the following:

SHARE ROW EXCLUSIVE, EXCLUSIVE, or SHARE INDEX EXLUSIVE.

SHARE ROW EXCLUSIVE mode locks exclusively those rows that are selected by a subsequent SELECT statement. It conflicts with EXCLUSIVE, SHARE INDEX EXCLUSIVE, and other SHARE ROW EXCLUSIVE locks on the rows selected, but does not prevent reading.

EXCLUSIVE mode locks exclusively the whole table. It conflicts with SHARE ROW EXCLUSIVE, SHARE INDEX EXCLUSIVE, and other EXCLUSIVE locks, and prevents any reading of the locked table.

SHARE INDEX EXCLUSIVE mode locks exclusively those entries in the index that are selected by a subsequent SELECT statement. It conflicts with SHARE ROW EXCLUSIVE, EXCLUSIVE, and other SHARE INDEX EXCLUSIVE locks, but does not prevent reading.

(a) Consider the code fragments of the Java JDBC program on the facing page.
Note that "…" after a comment is in place of the code that implements the action of the comment.

i) Which commands determine transaction boundaries? [3 marks]

ANSWER

ii) Which command performs actual locking in the book table and what is locked? [3 marks]

ANSWER

iii) What will be the contents of the res object if the command res.next(); returns true? [3 marks]

ANSWER

iv) Why is a try catch pair of blocks needed inside the catch block at the end of the code?[3 marks]

ANSWER

```
 . . .
Connection con;
int j_isbn;
try {
   // Establish a connection with the database
    . . .
   con.setAutoCommit(false);
   Statement lockbook=con.createStatement();
   lockbook.executeUpdate("LOCK book IN SHARE ROW EXCLUSIVE MODE");
   PreparedStatement retBook =
       con.prepareStatement("SELECT * FROM book WHERE isbn=?");
   // Obtain data for the variable j_isbn
     . . .
   retBook.setInt(1, j_isbn);
   ResultSet res = retBook.executeQuery();
   if (res.next()) {
       // Do some processing
          . . .
       // Do some database updating
           . . .
   }
   con.commit();
   con.setAutoCommit(true);
}
catch (SQLException ex) {
    try {
       con.rollback();
       con.setAutoCommit(true);
    }
    catch (SQLException e) {
    }
   return "\nSQLException: " + ex.getMessage();
}
// Close connection and exit
  . . .
```

(b) Consider the transaction programs T_1 and T_2 on the facing page. The BEGIN … COMMIT pair of statements defines the transaction boundaries. Suppose the isolation level is READ COMMITED. Transaction T_2 is normally used to enter assignment marks for all students and the transaction T_1 is run afterwards only if scaling up is needed. However, this time the transactions are run concurrently.

i) Concurrent execution of these two transactions leads to a well-known concurrency anomaly. What is the name of this anomaly? [1 mark]

ANSWER

ii) The outcome of running T_2 and T_1 as on the facing page will disadvantage student 9090, since his/her marks will appear as if they had been scaled up although they were not. On the other hand, if the Assignment 4 marks for student 9090 had not been published on the web page, the student would have come to ask about his/her marks, which would result in his/her marks being scaled up.

Make adjustment(s) to the program T_1 that will result in the most efficient behaviour in a mutiprogramming environment. Make your correction(s) to the program clearly visible. Briefly explain what you have done and why. [7 marks]

ANSWER

time

T_1	T_2
// Assignment({*StudId*, *AssigNo*, *Marks*}, {*StudId*, *AssigNo*})	
BEGIN;	
LOCK Assignment IN SHARE ROW EXCLUSIVE MODE;	
SELECT * FROM Assignment WHERE AssigNo = 4;	
StudId_| AssigNo |_Marks 3030 | 4 | 25.0 1010 | 4 | 28.0 (2 rows)	
// Assignment 4 was too hard; scale up all marks by 20%	
UPDATE Assignment SET Marks = 1.2*Marks WHERE AssigNo = 4;	
	// Insert marks of an approved late submission into the database BEGIN; INSERT INTO Assignments VALUES (9090, 4, 15); COMMIT;
// Read new marks and publish them on the course web page	
SELECT * FROM Assignment WHERE AssigNo = 4; StudId_| AssigNo |_Marks 3030 | 4 | 30.0 1010 | 4 | 33.6 9090 | 4 | 15.0 (3 rows)	
COMMIT;	

* * * * * * * * * * * * * * * * * * * *

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.

Specify the question number for work you do want marked.

FORMULAE FOR COMPUTING QUERY COST ESTIMATE

Blocking factor: $f = \lfloor B / L \rfloor$

Number of blocks: $b = \lceil r / f \rceil$

Selection cardinality of the attribute A : $s(A) = r / d(A)$, where $d(A)$ is the number of different A values

Number of buffers $n = \lfloor K / B \rfloor$, where K is the size of the buffer pool

C (project) = $b_1 + b_2$

C(project_distinct) = $b_1 + b_2 + 2b_2(1 + \lceil \log_m b_2 \rceil) + b_2 + b_3$, $m = n - 1$

C (select_linear) = $b_1 + \lceil s / f \rceil$

C (select_sec_key) = $x + s + \lceil s / f \rceil$

Costs of join algorithms

(o stands for the outer loop relation, and i stands for the inner loop relation)

C (nested_join) = $b_o + b_i \lceil b_o / (n - 2) \rceil + \lceil js * r_o * r_i / f \rceil$

C (single_join) = $b_o + r_o * f(index_i) + \lceil js * r_o * r_i / f \rceil$ (minimum of 4 buffers required)

C (sort_join) = $b_1(3 + 2\lceil \log_m b_1 \rceil) + b_2(3 + 2\lceil \log_m b_2 \rceil) + \lceil js * r_1 * r_2 / f \rceil$, $m = n - 1$

C(partition_join) = $3(b_1 + b_2) + \lceil js * r_1 * r_2 / f \rceil$, $n \geq \lceil (3 + (1 + 4b_1)^{1/2})/2 \rceil$, $b_1 < b_2$

C (sort) = $2b(1 + \lceil \log_m b \rceil)$

f ($index$):

primary index: $f(index) = x + 1$

secondary index: $f(index)$: $x + s$

Approximate formulae for choosing the most efficient join algorithm

Mandatory condition: $b_o << b_i$

(o stands for the outer loop relation, and i stands for the inner loop relation)

C(single) < C(nested) $\Leftrightarrow r_o * f(index_i) < b_i \lceil b_o / (n - 2) \rceil$ // Providing $r_o * f(index_i) >> b_o$

C(single) < C(sort) $\Leftrightarrow r_o * f(index_i) < b_i (3 + 2\lceil \log_m b_i \rceil)$ // Providing $r_o * f(index_i) >> b_o$

C(single) < C(partition-hash) $\Leftrightarrow r_o * f(index_i) < 3b_i$ // Providing $r_o * f(index_i) >> b_o$

C(nested) < C(sort-merge) $\Leftrightarrow b_o < (n - 2)(3 + 2\lceil \log_m b_i \rceil)$

C(nested) < C(partition-hash) $\Leftrightarrow \lceil b_o / (n - 2) \rceil \leq 3$

C(sort-merge) < C(partition-hash) $\Leftrightarrow \perp$