Umeå University
Department of Computing Science
TDBC86 — Principles of Database Systems
Examination: December 22, 2004

Name : _____

Personnummer (Swedish ID number): _____

UNIX user ID: _____

1. All answers must be written in English.
2. A pocket engineering calculator (miniräknare) and English/X - X/English dictionary may be used. No other help materials are allowed.
3. The examination consists of two parts. In Part I, which includes Problems 1–5, on pages 1–15, the answers **must** be written on the sheets provided. In general, only the answers will be checked for these questions; derivations need not be provided and will not be evaluated. In Part II, which includes Problems 6–10, the answers must be written on separate sheets.
4. You must turn in this cover sheet, the answer sheets for Part I (pages 1–15), and the pages containing your answers for Problems 6–10. Collate the sheets in numerical order of the problems, and write on only one side of the paper. Write your name and Swedish ID number on the top of each sheet. For questions 6–10, write the question number on the top of each sheet as well.
5. The examination has a total of 800 points, which constitutes 80% of the course grade. The other 20% is based upon the obligatory exercises, as explained in the course syllabus.
6. For Problems 6–10, you have the choice, for each part, to do the problem, or to skip it for partial credit. In the table below, place an X in the position for any problem for which you have attempted a solution, and which you wish to have graded. It is extremely important that you fill in this table properly, because of the following option. For any box which is left blank, the associated question will not be graded, and you will instead be awarded 15% of the points for that question. Your decision to leave a box blank is definitive, so be very careful. For example, If you leave box 8(b) blank, your answer to that question will not be graded, even if it is completely correct. On the other hand, if you place an X in box 8(b), but provide no answer whatsoever to that question, you will not receive 15% of the points for that question. It is strongly recommended that you use a pencil, in case you change your mind! *Note that Problems 1–5 do not carry this option. No points are awarded for skipping them.*

Prob	6	7	8	9	10
(a)					
(b)	[redacted]				
(c)	[redacted]				
(d)	[redacted]	[redacted]	[redacted]	[redacted]	
(e)	[redacted]	[redacted]	[redacted]	[redacted]	

Name and ID number: _____

(1: 120 points total; 8 points for each question) For each of the questions below, place an X between the parentheses to the left of the correct responses to the question. A question will have at least one correct choice, but it may have several. Mark all correct choices.

(a) Suppose that the *wound-wait* method of resolving conflicts is employed. Let T_1 and T_2 be transactions, and suppose that T_1 requires a resource which is held by T_2. Indicate which of the following statements are true.

( ) If T_1 has an earlier timestamp than T_2, then T_2 is aborted and restarted, thus allowing T_1 to continue.

( ) If T_1 has an earlier timestamp than T_2, then T_1 waits while T_2 is allowed to continue.

( ) If T_2 has an earlier timestamp than T_1, then T_1 is aborted and restarted, thus allowing T_2 to continue.

( ) If T_2 has an earlier timestamp than T_1, then T_2 waits while T_1 is allowed to continue.

(b) Suppose that the *wait-die* method of resolving conflicts is employed. Let T_1 and T_2 be transactions, and suppose that T_1 requires a resource which is held by T_2. Indicate which of the following statements are true.

( ) If T_1 has an earlier timestamp than T_2, then T_2 is aborted and restarted, thus allowing T_1 to continue.

( ) If T_1 has an earlier timestamp than T_2, then T_1 waits while T_2 is allowed to continue.

( ) If T_2 has an earlier timestamp than T_1, then T_1 is aborted and restarted, thus allowing T_2 to continue.

( ) If T_2 has an earlier timestamp than T_1, then T_2 waits while T_1 is allowed to continue.

(c) *Statistical tracking* is a method of obtaining information about an individual from a database. Indicate which of the following statements are true about this technique.

( ) To use this method, a query must be known which retrieves the information which is desired (although the system may refuse to answer this query).

( ) An effective measure of protection against such tracking is to require that queries return only statistics computed over a large number of tuples.

( ) An effective measure of protection against such tracking is to partition the database into blocks and require that all statistical queries include either none or all entries in each such block.

( ) An effective measure of protection against such tracking is to make all entries for individuals in the database anonymous by removing personal keys or near-keys such as name, identity number, social-security number, address, telephone number, *etc.*

Name and ID number: _____

(d) Suppose that a database system employs deferred update modified by a *no-steal and no-force* strategy for its DBMS cache. Indicate which of the following are true.

( ) Even for committed transactions, changes made by those transactions may not yet be visible in the database.

( ) Changes made by uncommitted transactions may nonetheless be visible in the database.

( ) At a checkpoint, the contents of all DBMS cache pages which are owned by committed transactions may be written to permanent memory.

( ) Pages in the DBMS cache may remain pinned even after the transaction commits.

( ) Before images must be maintained in the system log.

(e) Suppose that a database system employs deferred update modified by a *steal and force* strategy for its DBMS cache. Indicate which of the following are true.

( ) Even for committed transactions, changes made by those transactions may not yet be visible in the database.

( ) Changes made by uncommitted transactions may nonetheless be visible in the database.

( ) At a checkpoint, the contents of all DBMS cache pages which are owned by committed transactions may be written to permanent memory.

( ) Pages in the DBMS cache may remain pinned even after the transaction commits.

( ) Before images must be maintained in the system log.

(f) Indicate which of the following statements are true regarding the *write-ahead logging protocol*:

( ) A primary motivation for this protocol is to ensure that recovery is possible using information contained in the database log.

( ) It applies primarily to in-place (*i.e.*, immediate) update strategies.

( ) It applies primarily to deferred update strategies.

( ) It states that before images must be written to the log before database writes.

( ) It states that after images must be written to the log before database writes.

(g) Indicate which of the following statements are true regarding *dense indices*.

( ) They are an appropriate choice for implementing secondary indices.

( ) They require that the records which are to be indexed be stored in a certain physical order.

( ) They are possible even when the index field of the records is not a key.

( ) They have a distinct index entry for each record which is indexed.

Name and ID number: _____

(h) Indicate which of the following statements are true regarding *cluster indices*.

( ) They are an appropriate choice for implementing secondary indices.

( ) They require that the records which are to be indexed be stored in a certain physical order.

( ) They are possible even when the index field of the records is not a key.

( ) They have a distinct index entry for each record which is indexed.

(i) Suppose that the join of two relations $R[ABC]$ and $S[CDE]$ is to be computed on attribute C, and suppose further that each relation has a non-sequential index on attribute C. Indicate which of the following strategies are appropriate for an efficient implementation.

( ) To identify the joinable pairs elements, use the two indices as the basis for a simple merge process.

( ) Process the tuples of the smaller relation one-by-one, searching for a match with the other relation using the non-sequential index of that relation.

( ) Construct a temporary, common-hash table to identify the joinable elements.

( ) Construct a B^+-tree index for each relation, on attribute C, a nd then merge these two indices to compute the join.

(j) Indicate which of the following are true regarding *conservative* 2PL schedules.

( ) All locks are obtained at the very beginning of the transaction.

( ) All write locks are obtained at the very beginning of the transaction.

( ) Deadlock is not possible if all transactions follow this locking protocol.

( ) All locks are held until the very end of the transaction.

( ) All write locks are held until the very end of the transaction.

( ) Non-recoverable schedules are not possible if all transactions follow this locking protocol.

(k) Indicate which of the following are true regarding *strict* 2PL schedules.

( ) All locks are obtained at the very beginning of the transaction.

( ) All write locks are obtained at the very beginning of the transaction.

( ) Deadlock is not possible if all transactions follow this locking protocol.

( ) All locks are held until the very end of the transaction.

( ) All write locks are held until the very end of the transaction.

( ) Non-recoverable schedules are not possible if all transactions follow this locking protocol.

Name and ID number: _____

(l) Let $R[\mathbf{A}]$ be a relational schema on attribute set **A**, constrained by the set $\mathcal{F}$ of functional dependencies. Identify each condition which guarantees that this schema is in BCNF.

( ) The left-hand side of every FD in $\mathcal{F}$ consists of a single attribute.

( ) The schema is in 2NF and the left-hand side of every FD in $\mathcal{F}$ consists of a single attribute.

( ) The schema is in 3NF and the left-hand side of every FD in $\mathcal{F}$ consists of a single attribute.

( ) The set **A** consists of exactly two attributes.

( ) The set **A** consists of exactly three attributes.

( ) The schema has only one candidate key.

(m) Indicate which of the following statements regarding *unsafe queries* are true.

( ) An unsafe query may return tuples with entries which do not occur in any relation in the database.

( ) It is impossible to express unsafe queries in the relational algebra.

( ) It is impossible to express unsafe queries in the relational tuple calculus.

( ) It is impossible to express unsafe queries in SQL.

(n) Indicate which of the following statements are true regarding the *object-oriented database model*.

( ) In the query language OQL proposed by ODMG, it is possible to express stand-alone queries, just as in SQL.

( ) A major advantage of the object-oriented data model over the relational model is its ability to handle the persistence problem for data.

( ) A major advantage of the object-oriented data model over the relational model is its simplicity; in particular, people who are not experts in computing have a much easier time understanding it.

( ) If the situation to be modelled involves a natural hierarchy, the object-oriented data model is likely to be a superior choice to the standard relational model.

(o) For support of ODBC under the C programming language on a given client system, two components must be installed. Identify them below, and indicate whether they are vendor specific or vendor independent.

( ) For each database vendor, a vendor-specific library of C routines for ODBC must be installed.

( ) Only a single vendor-independent library of C routines specific to ODBC need be installed.

( ) For each database vendor, a vendor-specific ODBC driver must be installed.

( ) Only a single a vendor-independent ODBC driver need be installed.

Name and ID number: _____

(2: 80 points total) Below is a series of questions regarding B-trees and B^+-trees. Place an X between the parentheses to the left of the correct responses to each question, and fill in the blanks which ask for specific numerical values. For each question which asks for a choice amongst alternatives, there is at least one correct choice, but there may be several. Mark all correct choices.

Page 20 of this examination is an appendix which gives several key formulas for both B-trees and B^+-trees; these may be used as a starting point for solving many of these questions.

Note also that the answers to questions (d) and (e) involve a parameter x, which may be chosen to be one of the three values in $\{-1,0,1\}$. This is just a shorthand way of selecting amongst alternatives. For example, if the formula given is $\sqrt[r+1]{R+1}+x$, then choosing $x=-1$ yields $\sqrt[r+1]{R+1}-1$, $x=0$ yields $\sqrt[r+1]{R+1}$, and $x=1$ yields $\sqrt[r+1]{R+1}+1$.

Hint: Instead of trying to remember the appropriate formulas, derive them from those given on page 20, just as if you were solving the problems on Obligatory Exercise 5.

(a: 8 points) Suppose that the size of each node in a B-tree is 4096 bytes, that each record occupies 100 bytes, and that each pointer occupies 4 bytes. Identify the *maximum* and *minimum* number of records which may be present in a non-root node.

Maximum: ( ) 38 ( ) 39 ( ) 40 ( ) 41
( ) none of the above.

Minimum: ( ) 17 ( ) 18 ( ) 19 ( ) 20
( ) none of the above.

(b: 8 points) Identify the formula or formulas which identify correctly the number of nodes in a uniform (m,r,d) B-tree containing R records.

( ) $1+\dfrac{R-m}{r}$
( ) $1+\sum_{i=0}^{d-1}(m+1)\cdot(r+1)^i$
( ) $1+\dfrac{(m+1)\cdot((r+1)^d-1)}{r}$
( ) $1+\dfrac{(m+1)\cdot((r+1)^{d-1}-1)}{r}$
( ) none of the above.

(c: 8 points) Suppose that a B-tree contains R records, has height d, and a uniform distribution of records in all nodes, including the root. Identify the formula or formulas which give the average number of records per node.

( ) $\sqrt[d]{R+1}$ ( ) $\sqrt[d]{R+1}+1$ ( ) $\sqrt[d]{R+1}-1$
( ) $\sqrt[d+1]{R+1}$ ( ) $\sqrt[d+1]{R+1}+1$ ( ) $\sqrt[d+1]{R+1}-1$
( ) $\sqrt[d-1]{R+1}$ ( ) $\sqrt[d-1]{R+1}+1$ ( ) $\sqrt[d-1]{R+1}-1$
( ) none of the above.

Name and ID number: _____

(d: 24 points) Suppose that each node of a B-tree may contain at most 17 records.

(i) Identify the formula or formulas which give the *maximum* height that a uniform B-tree containing R records may have. For each formula chosen, indicate furthermore whether the value of x to be used is -1, 0, or 1. No other choices for x are allowed.
Note: You do not need to compute a numerical value for d; just identify the appropriate formula(s).

( ) $\left\lfloor \log_{r+1}\left(\frac{R+1}{m+1}\right)+x \right\rfloor$ with $x =$___ ( ) $\left\lceil \log_{r+1}\left(\frac{R+1}{m+1}\right)+x \right\rceil$ with $x =$___

( ) $\left\lfloor \sqrt[r+1]{\frac{R+1}{m+1}}+x \right\rfloor$ with $x =$___ ( ) $\left\lceil \sqrt[r+1]{\frac{R+1}{m+1}}+x \right\rceil$ with $x =$___

( ) $\lfloor \log_{r+1}(R+1)+x \rfloor$ with $x =$___ ( ) $\lceil \log_{r+1}(R+1)+x \rceil$ with $x =$___

( ) $\left\lfloor \sqrt[r+1]{R+1}+x \right\rfloor$ with $x =$___ ( ) $\left\lceil \sqrt[r+1]{R+1}+x \right\rceil$ with $x =$___

( ) none of the above.

The values for m and r to use are (fill in the blanks):

$m =$ _____ $r =$ _____

(Give the appropriate numerical values for both m and r even though (some of) the formulas which you have chosen do not reflect all of these parameters explicitly).

(ii) Identify the formula or formulas which give the *minimum* height that a uniform B-tree containing R records may have. For each formula chosen, indicate furthermore whether the value of x to be used is -1, 0, or 1. No other choices for x are allowed.
Note: You do not need to compute a numerical value for d; just identify the appropriate formula(s).

( ) $\left\lfloor \log_{r+1}\left(\frac{R+1}{m+1}\right)+x \right\rfloor$ with $x =$___ ( ) $\left\lceil \log_{r+1}\left(\frac{R+1}{m+1}\right)+x \right\rceil$ with $x =$___

( ) $\left\lfloor \sqrt[r+1]{\frac{R+1}{m+1}}+x \right\rfloor$ with $x =$___ ( ) $\left\lceil \sqrt[r+1]{\frac{R+1}{m+1}}+x \right\rceil$ with $x =$___

( ) $\lfloor \log_{r+1}(R+1)+x \rfloor$ with $x =$___ ( ) $\lceil \log_{r+1}(R+1)+x \rceil$ with $x =$___

( ) $\left\lfloor \sqrt[r+1]{R+1}+x \right\rfloor$ with $x =$___ ( ) $\left\lceil \sqrt[r+1]{R+1}+x \right\rceil$ with $x =$___

( ) none of the above.

The values for m and r to use are (fill in the blanks):

$m =$ _____ $r =$ _____

(Give the appropriate numerical values for both m and r even though (some of) the formulas which you have chosen do not reflect all of these parameters explicitly).

Name and ID number: _____

(e: 24 points) Suppose that each internal node of a B^+-tree may contain at most 201 indices, and each leaf node may contain at most 15 records.

(i) Identify the formula or formulas which give the *maximum* height that a uniform B^+-tree containing R records may have. For each formula chosen, indicate furthermore whether the value of x to be used is -1, 0, or 1. No other choices for x are allowed.
Note: You do not need to compute a numerical value for d; just identify the appropriate formula(s).

( ) $\left\lfloor \log_{q+1}\left(\frac{R+1}{(m+1)\cdot r}\right)+x \right\rfloor$ with $x=$___ ( ) $\left\lceil \log_{q+1}\left(\frac{R+1}{(m+1)\cdot r}\right)+x \right\rceil$ with $x=$___

( ) $\left\lfloor \sqrt[q]{\frac{R+1}{(m+1)\cdot r}}+x \right\rfloor$ with $x=$___ ( ) $\left\lceil \sqrt[q]{\frac{R+1}{(m+1)\cdot r}}+x \right\rceil$ with $x=$___

( ) $\left\lfloor \log_{q+1}\left(\frac{R}{r}\right)+x \right\rfloor$ with $x=$___ ( ) $\left\lceil \log_{q+1}\left(\frac{R}{r}\right)+x \right\rceil$ with $x=$___

( ) none of the above.

The values for m, r, and q to use are (fill in the blanks):

$m=$ _____ $r=$ _____ $q=$ _____

(Give the appropriate numerical values for each of m, r, and q, even though (some of) the formulas which you have chosen do not reflect all of these parameters explicitly).

(ii) Identify the formula or formulas which give the *minimum* height that a uniform B^+-tree containing R records may have. For each formula chosen, indicate furthermore whether the value of x to be used is -1, 0, or 1. No other choices for x are allowed.
Note: You do not need to compute a numerical value for d; just identify the appropriate formula(s).

( ) $\left\lfloor \log_{q+1}\left(\frac{R+1}{(m+1)\cdot r}\right)+x \right\rfloor$ with $x=$___ ( ) $\left\lceil \log_{q+1}\left(\frac{R+1}{(m+1)\cdot r}\right)+x \right\rceil$ with $x=$___

( ) $\left\lfloor \sqrt[q]{\frac{R+1}{(m+1)\cdot r}}+x \right\rfloor$ with $x=$___ ( ) $\left\lceil \sqrt[q]{\frac{R+1}{(m+1)\cdot r}}+x \right\rceil$ with $x=$___

( ) $\left\lfloor \log_{q+1}\left(\frac{R}{r}\right)+x \right\rfloor$ with $x=$___ ( ) $\left\lceil \log_{q+1}\left(\frac{R}{r}\right)+x \right\rceil$ with $x=$___

( ) none of the above.

The values for m, r, and q to use are (fill in the blanks):

$m=$ _____ $r=$ _____ $q=$ _____

(Give the appropriate numerical values for each of m, r, and q, even though (some of) the formulas which you have chosen do not reflect all of these parameters explicitly).

Name and ID number: _____

(f: 8 points) Suppose that a B^+-tree contains R records, has height d, and a uniform distribution of indices in all internal nodes, including the root, and contains exactly r records per leaf node. Identify the formula or formulas which give the average number of indices per internal node.

( ) $\sqrt[d-1]{\frac{R+1}{r+1}}$ ( ) $\sqrt[d-1]{\frac{R+1}{r}}-1$ ( ) $\sqrt[d-1]{\frac{R}{r}}-1$ ( ) $\sqrt[d-1]{\frac{R}{r}-1}$

( ) $\sqrt[d]{\frac{R+1}{r+1}}$ ( ) $\sqrt[d]{\frac{R+1}{r}}-1$ ( ) $\sqrt[d]{\frac{R}{r}}-1$ ( ) $\sqrt[d]{\frac{R}{r}-1}$

( ) $\sqrt[d+1]{\frac{R+1}{r+1}}$ ( ) $\sqrt[d+1]{\frac{R+1}{r}}-1$ ( ) $\sqrt[d+1]{\frac{R}{r}}-1$ ( ) $\sqrt[d+1]{\frac{R}{r}-1}$

( ) none of the above.

Name and ID number: _____

(3: 70 points total) A simple database system has five data objects, named v, w, x, y, and z. The following four transactions are given.

$$\begin{aligned} T_1 &= \mathsf{r}_1(v)\mathsf{r}_1(x)\mathsf{w}_1(y)\mathsf{w}_1(x) \\ T_2 &= \mathsf{r}_2(x)\mathsf{w}_2(x)\mathsf{w}_2(w) \\ T_3 &= \mathsf{r}_3(w)\mathsf{r}_3(y)\mathsf{w}_3(w)\mathsf{w}_3(y) \\ T_4 &= \mathsf{w}_4(v)\mathsf{w}_4(w)\mathsf{r}_4(y)\mathsf{w}_4(y)\mathsf{r}_4(z) \end{aligned}$$

(a: 10 points) In the table below, for each transaction, circle the names of those data items on which it executes a blind write. Circle *none* just in case the transaction does not execute any blind writes.

T_1 blind writes: v w x y z none

T_2 blind writes: v w x y z none

T_3 blind writes: v w x y z none

T_4 blind writes: v w x y z none

Shown on the next page are three schedules for this set of transactions. For each of these schedules, do the following **on that sheet**:

- Using the nodes provided, draw the edges of the conflict graph.
- Under the name of each property (*serial*, *view serializable*, *conflict serializable*), circle **T** (*true*) or **F** (*false*), depending upon whether or not the schedule possesses that property.
- If the schedule is view serializable, give the transaction sequence (*e.g.*, $T_4T_1T_3T_2$) of an equivalent serial schedule. If the schedule is not view serializable, no answer is required for this part.
- Assume that each transaction begins just before performing its first (read or write) operation, and that it commits just after performing its final (read or write) operation. Under these assumptions, indicates whether or not the schedule is recoverable (under pure immediate update) by placing an $\times$ in the appropriate $\square$ as so: $\boxtimes$. In the case that it is not recoverable, also identify a pair of transactions which may cause a recovery problem by writing the appropriate subscripts in the spaces provided. (In the case that there are several such pairs, only one need be identified.)

Name and ID number: _____

(b: 20 points) $\mathsf{w}_4(v)\mathsf{r}_1(v)\mathsf{r}_3(w)\mathsf{w}_4(w)\mathsf{r}_3(y)\mathsf{w}_3(w)\mathsf{w}_3(y)\mathsf{r}_2(x)\mathsf{w}_2(x)\mathsf{w}_2(w)\mathsf{r}_4(y)\mathsf{w}_4(y)\mathsf{r}_1(x)\mathsf{w}_1(y)\mathsf{w}_1(x)\mathsf{r}_4(z)$

(T_1) (T_2) serial T F view serializable T F conflict serializable T F

(T_3) (T_4) Equivalent serial schedule: _____

☐ This schedule is recoverable.

☐ This schedule is not recoverable because transaction T_____ may need to be rolled back after its commit point if transaction T_____ aborts.

(c: 20 points) $\mathsf{r}_1(v)\mathsf{r}_1(x)\mathsf{w}_1(y)\mathsf{w}_1(x)\mathsf{r}_3(w)\mathsf{r}_3(y)\mathsf{w}_3(w)\mathsf{w}_4(v)\mathsf{w}_4(w)\mathsf{w}_3(y)\mathsf{r}_4(y)\mathsf{w}_4(y)\mathsf{r}_2(x)\mathsf{w}_2(x)\mathsf{w}_2(w)\mathsf{r}_4(z)$

(T_1) (T_2) serial T F view serializable T F conflict serializable T F

(T_3) (T_4) Equivalent serial schedule: _____

☐ This schedule is recoverable.

☐ This schedule is not recoverable because transaction T_____ may need to be rolled back after its commit point if transaction T_____ aborts.

(d: 20 points) $\mathsf{r}_3(w)\mathsf{w}_4(v)\mathsf{r}_1(v)\mathsf{r}_1(x)\mathsf{w}_4(w)\mathsf{r}_2(x)\mathsf{w}_2(x)\mathsf{r}_3(y)\mathsf{r}_4(y)\mathsf{w}_4(y)\mathsf{w}_3(w)\mathsf{w}_1(y)\mathsf{w}_1(x)\mathsf{r}_4(z)\mathsf{w}_3(y)\mathsf{w}_2(w)$

(T_1) (T_2) serial T F view serializable T F conflict serializable T F

(T_3) (T_4) Equivalent serial schedule: _____

☐ This schedule is recoverable.

☐ This schedule is not recoverable because transaction T_____ may need to be rolled back after its commit point if transaction T_____ aborts.

Name and ID number: _____

(4: 80 points) A database contains four data objects, A, B, C, and D. Six transactions are given on these objects, specified as follows.

T_1: Read(A)	T_2: Read(A)	T_3: Read(C)	T_4: Read(C)	T_5: Read(B)	T_6: Read(D)
Write(A)	Write(A)	Write(C)	Write(C)	Write(B)	Write(D)
Read(B)	Read(C)	Read(B)	Read(D)		
Write(B)	Write(C)	Write(B)	Write(D)		
Read(C)					
Write(C)					

Shown to the right is a schedule for this set of transactions, including two marked points at which a crash occurs. Using this schedule, answer the questions given on the following two pages. The following points are relevant:

- In the simple model here, only the pure immediate-update protocol and the pure deferred-update protocol will be considered. There is no DBMS cache, and so no need for checkpoints.
- Assume that all active transactions abort at the time of a crash, but that the current state of the database, as well as the log, are not affected. (In other words, assume that this is a crash of primary memory and processor state, with secondary storage, including both the database and the log, completely unaffected.)
- The choice "No further action necessary" is appropriate when the transaction had started before the time of the crash, but no special action on it is necessary for crash recovery. (It may or may not have committed before the time of the crash.)
- If a transaction has not yet begun at the time of a crash, then the appropriate answer for that transaction under both protocols is "Transaction not yet started," (but you must still mark this answer to obtain credit). Of course, the transaction will need to be run, but not as part of the recovery process.

T_1:	Begin
T_1:	Read(A)
T_1:	Write(A)
T_4:	Begin
T_4:	Read(C)
T_2:	Begin
T_2:	Read(A)
T_2:	Write(A)
T_4:	Write(C)
T_4:	Read(D)
T_4:	Write(D)
T_4:	Commit
T_2:	Read(C)
T_2:	Write(C)
T_2:	Commit
T_1:	Read(B)
T_1:	Write(B)
T_3:	Begin
T_3:	Read(C)
T_3:	Write(C)
	Crash(1)
T_1:	Read(C)
T_1:	Write(C)
T_1:	Commit
T_5:	Begin
T_5:	Read(B)
T_5:	Write(B)
T_5:	Commit
T_6:	Begin
T_6:	Read(D)
T_6:	Write(D)
T_6:	Commit
	Crash(2)
T_3:	Read(B)
T_3:	Write(B)
T_3:	Commit

Name and ID number: _____

In answering the questions below, use exactly schedule shown on page 11, assuming that a crash occurs at the point labelled **Crash**(1).

(a) To recover from the system crash in the context of the *pure immediate-update protocol*, for each transaction, indicate which of the following four actions is necessary by placing an $\times$ in the appropriate ☐, as so ☒.

T_1: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_2: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_3: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_4: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_5: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_6: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

(b) To recover from the system crash in the context of the *pure deferred-update protocol*, for each transaction, indicate which of the following four actions is necessary by placing an $\times$ in the appropriate ☐, as so ☒.

T_1: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_2: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_3: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_4: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_5: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_6: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

Name and ID number: _____

In answering the questions below, use exactly schedule shown on page 11, assuming that a crash occurs at the point labelled **Crash**(2), but no crash occurred at the point labelled **Crash**(1) (or, equivalently, that recovery from **Crash**(1) was completed before **Crash**(2) occurred).

(c) To recover from the system crash in the context of the *pure immediate-update protocol*, for each transaction, indicate which of the following four actions is necessary by placing an $\times$ in the appropriate $\square$, as so $\boxtimes$.

T_1: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_2: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_3: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_4: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_5: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_6: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

(d) To recover from the system crash in the context of the *pure deferred-update protocol*, for each transaction, indicate which of the following four actions is necessary by placing an $\times$ in the appropriate $\square$, as so $\boxtimes$.

T_1: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_2: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_3: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_4: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_5: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

T_6: ☐ No further action necessary ☐ Re-do from log entries ☐ Rollback and re-run ☐ Transaction not yet started

Name and ID number: _____

(5: 110 points total; 10 points for each part) Important: The answers to this question must be indicated below; do not write the answers to this question on a blank sheet.

The following relational schema with relation $R[ABCDEFG]$ and functional dependencies $\mathcal{F} = \{A \rightarrow B,\ C \rightarrow A,\ BD \rightarrow G,\ EF \rightarrow D,\ G \rightarrow AC\}$ applies to all ten parts of this problem.

Given below are six projections of $R[ABCDEFG]$. Assume that each such projection is constrained by exactly the functional dependencies in $\mathcal{F}^+$ which embed in the projection. In the space provided, give all of the candidate keys for the projection, as well as a basis of the functional dependencies from $\mathcal{F}^+$ which embed in it. Also, circle *all* of the normal-form properties which the projection possesses (even if some are implied by others). Circle the word none just in case none of the normal-form properties holds.

(a) Projection: $R[BCDG]$

Candidate keys: _____

FD basis: _____

Normal forms: None 2NF 3NF BCNF

(b) Projection: $R[ADEF]$

Candidate keys: _____

FD basis: _____

Normal forms: None 2NF 3NF BCNF

(c) Projection: $R[ABDG]$

Candidate keys: _____

FD basis: _____

Normal forms: None 2NF 3NF BCNF

(d) Projection: $R[BEF]$

Candidate keys: _____

FD basis: _____

Normal forms: None 2NF 3NF BCNF

(e) Projection: $R[ABEF]$

Candidate keys: _____

FD basis: _____

Normal forms: None 2NF 3NF BCNF

Name and ID number: _____

(f) Projection: $R[ACG]$

Candidate keys: _____

FD basis: _____

Normal forms: None 2NF 3NF BCNF

(g) Projection: $R[AB]$

Candidate keys: _____

FD basis: _____

Normal forms: None 2NF 3NF BCNF

Shown below are four decompositions of the schema on the previous page. For each decomposition, indicate whether or not it has the properties of *lossless* and *dependency preserving* by circling the appropriate word(s). Circle the word *neither* just in case it is neither lossless nor dependency preserving.

In checking the lossless property, use only pairwise reconstructions (*i.e.*, join trees). Do not test for cyclic decompositions (*i.e.*, decompositions which are lossless, but the join can not be reconstructed pairwise) [It was not explained in the course how to test for such decompositions, so if you do not understand what this means, just use the techniques which were presented in the course.]

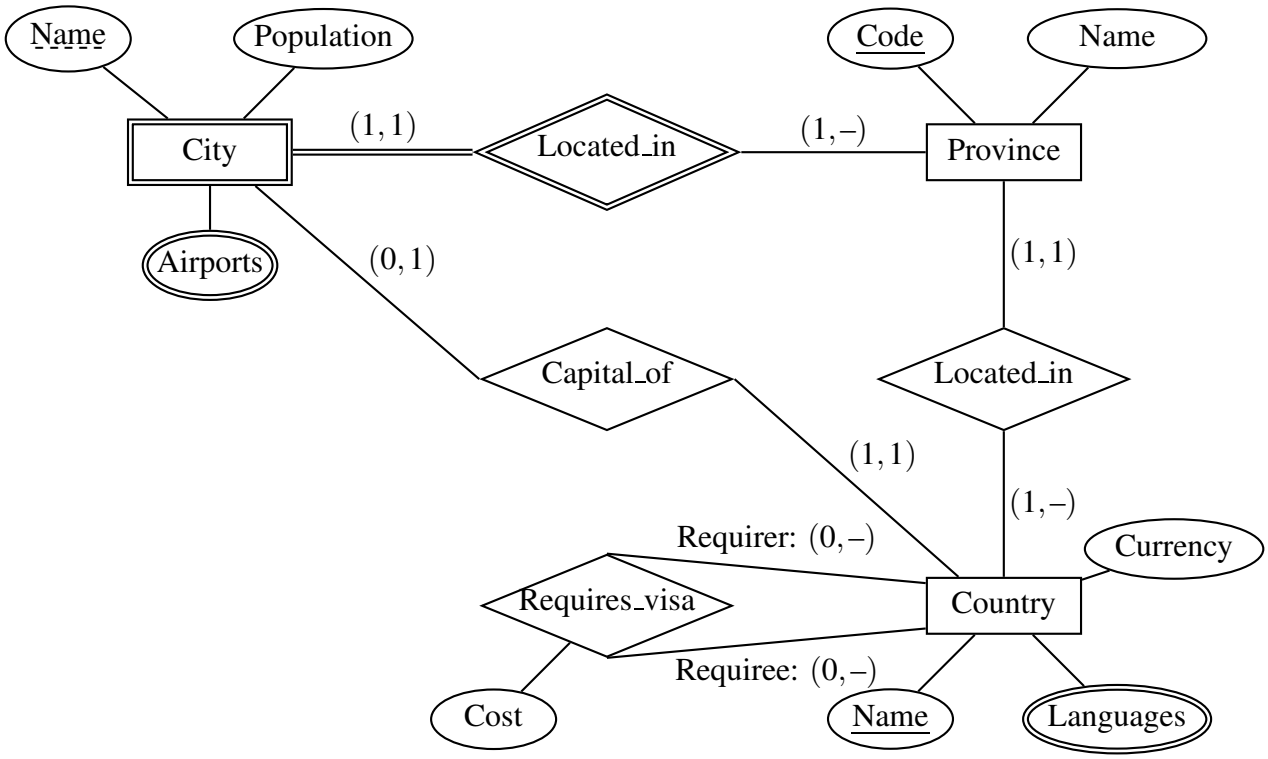
(h) Decomposition: $\{R[ADEF], R[BEF], R[BCDG]\}$.
Neither Lossless Dependency preserving

(i) Decomposition: $\{R[ADEF], R[BCDG], R[ABDG]\}$.
Neither Lossless Dependency preserving

(j) Decomposition: $\{R[ADEF], R[AB], R[BCDG], R[ACG]\}$.
Neither Lossless Dependency preserving

(k) Decomposition: $\{R[ADEF], R[ABDG], R[ACG]\}$.
Neither Lossless Dependency preserving

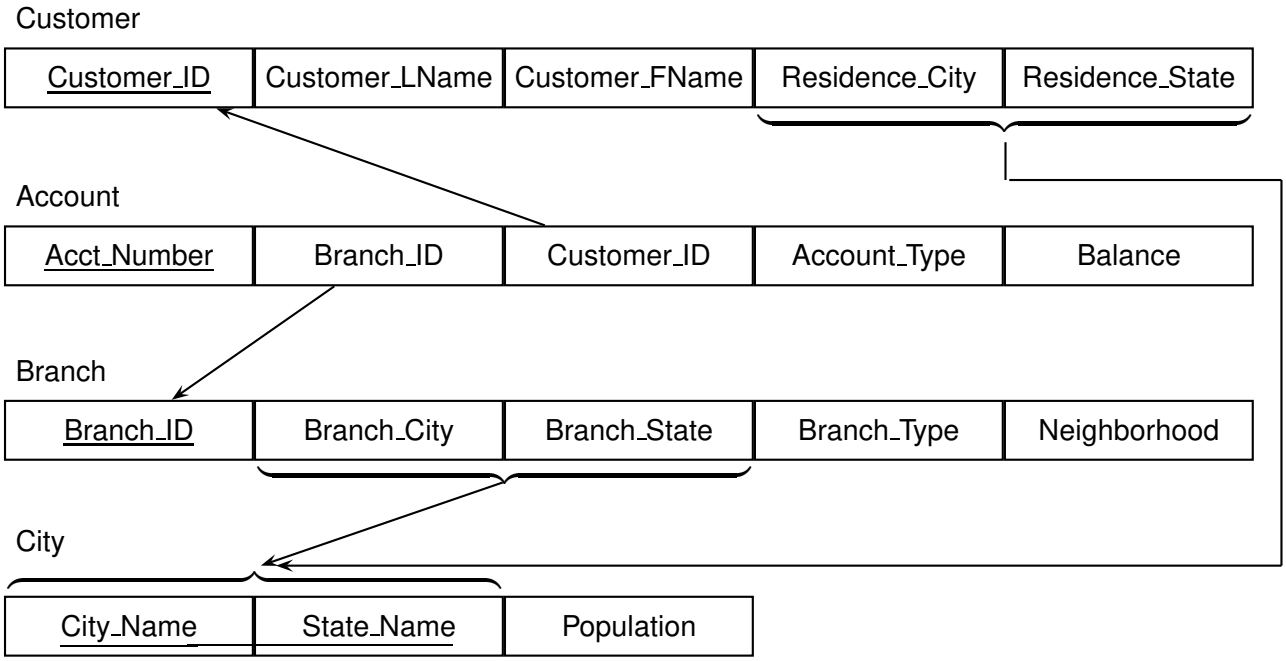
(6: 60 points) Shown below is an ER-diagram for a geographical database. Using the techniques developed in the course, map this diagram into an equivalent relational schema. Show all keys, primary and foreign, and link foreign keys to their primary partners.



Notes:

- *Province* is a division of a country similar to *state* in the USA or *län* in Sweden.
- *Code* is an identifier which uniquely identifies the province; *e.g.*, *NH* for *New Hampshire* in the USA or *AC* for *Västerbotten* in Sweden. For the purposes of this exercise, assume that these codes are unique over all provinces in all countries included in the database.
- The *Requires_visa* relationship recaptures that one country (the *requirer*) requires a visa for all citizens from a second country (the *requiree*).
- The (x, y) notation gives the minimum and maximum number of times that a given instance of the entity may participate in the relationship. Thus, $(1, 1)$ means exactly one, and $(0, –)$ means any number.
- If a primary and/or foreign key consists of more than one attribute, make sure that your notation identifies and links these composite keys as groups.
- Use design decisions which produce the simplest relational schema. If there is a choice between creating an additional relation and adding attributes to an existing one, choose the latter. Similarly, a foreign key which cannot take on null values is to be preferred to one which can.

The following schema applies to Problems 7 through 10. In each case, the primary key is underlined, and the arrows run from foreign keys to their primary counterparts.



The following queries apply to Problems 5 through 7, and reference the schema defined above.

(a) Find the names (last and first) of those customers who have both an account of type *checking* and an account of type *savings*.

(b) Find the names (last and first) of those customers who have at least two distinct accounts which are held at the same branch (*i.e.*, with the same Branch_ID).

(c) Find the names (last and first) of those customers who have an account of every type (*i.e.*, Account_Type) which is listed in the database.

Note: In the above schema, the attributes Customer_LName and Customer_FName identify the last and first names of the customer, respectively.

(7: 60 points total; 20 for each part) Solve each of the three queries (a) - (c) in the relational algebra. Functional operators, such as count and average, may not be used. Use the join notation described in the course notes: (The symbol $\bowtie$ is used to denote join; in the absence of subscripts it denotes the natural join; any other join conditions must be specified explicitly using subscripts.)

(8: 60 points total; 20 for each part) Solve each of the three queries (a) - (c) in the tuple relational calculus. Functional operators, such as count and average, may not be used.

(9: 60 points total; 20 for each part) Solve each of the three queries (a) - (c) using SQL. To keep the solutions simple and consistent, the following restrictions apply:

(i) Directives involving the keyword JOIN may not be used. Express the join conditions directly using conditions in the WHERE clause.

(ii) The solution must consist of a single SQL directive. Creation and subsequent use of temporary tables is not permitted.

(iii) Subqueries involving SELECT are allowed only in the WHERE and HAVING fields of queries and subqueries.

(iv) Aggregate operators, such as COUNT, SUM, AVG, MAX, and MIN may **not** be used.

(10: 100 points total; 20 points for each part) Provide solutions, in SQL, to the following additional queries. Restrictions (i)-(iii) from Problem 8 apply here as well. On the other hand, aggregate operations may be used, but only in the SELECT and HAVING clauses of queries and subqueries.

(a) For each value of Branch_ID which is listed in the Branch relation, display (in addition to Branch_ID) the Branch_City and Branch_State, as well as the total balance and the average balance of all accounts which are held at that branch.

(b) For each city-state pair (*i.e.*, key) which is listed in the City relation, list the total number of accounts held by customers who reside in that city and state, as well as the average balance of those accounts. If the total number of accounts for a city-state pair in the City relation is zero, then give the average balance as zero as well.

(c) Display the customer ID, last name, and first name of those customers whose total balance in all accounts of type *savings* is is greater than the balance of any single account in the entire database, regardless of its type.

(d) For every account of type *savings* which is housed in a branch which is located in a neighborhood named *French*, change the type of the account to *dépôt*.

(e) Give an SQL CREATE TABLE command which creates the relation Branch of the figure on page 17. You may assume that the appropriate definitions for the other relations have already been made. Assume further the following:

- The field Branch_ID consists of a sequence of exactly five characters.
- The field Branch_City consists of strings of at most twenty characters.
- The field Branch_State consists of a sequence of exactly two characters.
- The field Branch_Type is an integer.
- The field Neighborhood consists of a string of at most twelve characters.
- In addition to the indicated primary key, the set {Branch_City, Branch_State, Neighborhood} is a candidate key.
- All primary and candidate key fields must be non-null.
- All foreign key fields must be non-null.
- All other fields may be null.

Further note regarding Problems 7-10: Although it is stated in Problem 10(e) above that {Branch_City, Branch_State, Neighborhood} is a candidate key for the relation Branch, your solutions in Problems 7-9 and 10(a)-(d) may not make this assumption. Your solutions must work correctly even in the case that Branch_ID is the only key for the relation Branch.

Appendix

A useful formula for uniform (m,r,d) B-trees:

$$R(m,r,d) = (m+1) \times (r+1)^d - 1$$

with

$$\begin{aligned} m &= \text{the total number of indices at the root node.} \\ r &= \text{the total number of records in each non-root node.} \\ d &= \text{the depth of the tree; } \textit{i.e.}, \\ &\quad \text{the length of a path from the root to a(ny) leaf node.} \\ R(m,r,d) &= \text{the total number of records in the tree.} \end{aligned}$$

A useful formula for uniform (m,q,r,d) B^+-trees:

$$R(m,q,r,d) = (m+1) \times (q+1)^{d-1} \times r$$

with

$$\begin{aligned} m &= \text{the total number of indices at the root node.} \\ q &= \text{the total number of indices in each non-root, non-leaf node.} \\ r &= \text{the total number of records in each leaf node.} \\ d &= \text{the depth of the tree; } \textit{i.e.}, \\ &\quad \text{the length of a path from the root to a(ny) leaf node.} \\ R(m,q,r,d) &= \text{the total number of records in the tree.} \end{aligned}$$