

# Final Exam (500pts)

May 31, 2002

NUMBER: \_\_\_\_\_

(Write this number on the back of each sheet  
in the exam body -10pts if your name appears in  
the exam body)

NAME: \_\_\_\_\_

PERSON NUMBER: \_\_\_\_\_

EMAIL: \_\_\_\_\_@cs.umu.se

## 1 True or False - 150 points

Each question is worth 10 points. If you mark a question wrong, then you lose 10 points. The lowest possible score on the entire section will be 0 points.

- 1.) Database researchers are encouraged with the constructs in DTD based XML – especially the ability to handle updates to client-side caches \_\_\_\_\_.
- 2.) Inclusion dependencies must be enforced over superclass/subclass relations \_\_\_\_\_.
- 3.) In Allen’s interval logic does  $(x < y) \wedge (y f z) \models \neg(x d z)$ ? \_\_\_\_\_.
- 4.) When locating the objects at a given point, multiple paths may be traversed through an R+ tree index. \_\_\_\_\_.
- 5.) PostgreSQL supports row types and collection types. \_\_\_\_\_.
- 6.) PostgreSQL inheritance enforces inclusion dependencies. \_\_\_\_\_.
- 7.) SQL1999 proposes linear recursion capability. \_\_\_\_\_.
- 8.) Nested relations are in first normal form, but not necessarily in second normal form. \_\_\_\_\_.
- 9.) if  $\phi \vdash \varphi \Rightarrow \phi \models \varphi$  then the reasoning algorithm is sound, but not necessarily complete. \_\_\_\_\_.
- 10.) Prolog uses a bottom up evaluation strategy \_\_\_\_\_.
- 11.) The following rule is safe:  $\text{Alive}(X) :- \text{Person}(X, \text{AGE}), \sim \text{Impossible\_age}(\text{AGE})$ . \_\_\_\_\_.
- 12.) LDL’s evaluation strategy is based directly on horn clause resolution.
- 13.) A major effort in data ware housing has been to provide simultaneous OLAP and OLTP over the same database.
- 14.) for the item set  $X$  (and  $Y$ ),  $\text{support}(X) > \beta \Rightarrow \text{support}(X \cup Y) > \beta$ . \_\_\_\_\_.
- 15.) In general you may convert a decision tree to a single LDL rule. \_\_\_\_\_.

## 2 Written Answers

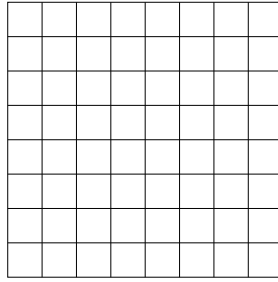
### 2.1 EER – 50 points

A *person* has a first name, last name, and person number. A *company* has a unique name and a stock symbol. A person or a company may be a *plaintiff* or a *defendant involved* in a *law suit*. A law suit has a description and involves one or more plaintiff (those who are bringing the suit) and one or more defendants (those who are accused). A *Judge* is a person who *presides* over a law-suit. In addition a Judge has a politics attribute (for example either “conservative” or “liberal”).

1.)(25pts) Draw an EER diagram that represents the above model. (Use categories)

2.)(25pts) Design a object-relational schema that represents this EER diagram (Use surrogate keys). Express inheritance by drawing an double lined arrow from sub-relation to super-relation. Identify foreign keys with single lined arrows.

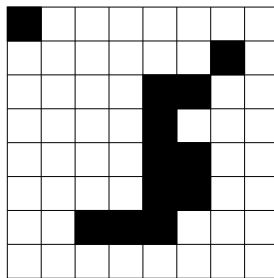
## 2.2 Quad Trees – 50 points



1.)(15pts) Following an “upper left quadrant and clockwise” fill-in protocol, fill the image above with the quad tree:

(1 (1 0 (1 0 1 0) 0) 0 (1 0 0 1))

2.)(15pts) Using the same protocol please give a quad tree for the following image:



3.)(20ptp) Justify (or refute) the following claim: “Subtraction between two quad tree images is faster on average than intersection.”



## 2.4 Temporal Databases – 50 points

```
(Fossil Number, Description, Species, Dated, TST, TET)
    001 Skull Fragment, Homo Erectus, 3500, 1946, 1978
    001 Skull Fragment, Homo Habulus, 3780, 1979, uc,
    002 Leg Bone Home Sapien, 450, 1996, uc,
...

```

In SQL (or Datalog if you wish):

- 1.) (15pts) Give the collection of fossils thought to belong to Homo Sapiens as of 1979.
- 2.) (15pts) Give the fossils that have been classified to two of more different species at different times.
- 3.) (20pts) In pseudo code show how to update all the Skull fragments dated before 500 and classified to Homo Sapien, to be classified to Homo Erectus. Assume this transaction to be carded out in 2002.

## 2.5 Datalog Queries – 50 points

Assume the EDB relations:

```
CaresFor(name1, name2). // name1 cares for name2.  
Dislikes(name1, name2). // name1 dislikes name2.  
Person(name, gender, age)
```

Assume that the integrity constraint  $\sim\text{CaresFor}(X,Y) :- \text{Dislikes}(X,Y)$  holds.

1.) Define the IDB predicate for `Friends(Name, Name)` Two people are friends if the mutually care for one another.

2.) Define the IDB predicate for `FriendlyConnection(Name, Name)`.

For example if `Friends('mike', 'al')` and `Friends('al', 'jose')` then `FriendConnected('mike', 'jose')`.

3.) Define the IDB predicate `Clique(Name1, Name2)` where the `name1` and `name2` are friendly connected and do not have a friendly connection with someone who dislikes (or is disliked) by the other.

## 2.6 DLs – 50 points

Assume the primitive concepts:  $Product$ ,  $TV \subset Product$ , and  $Stereo \subset Product$ .

Assume the roles *worksWith* and *manufacturedBy*.

State whether the following hold:

1.)  $TV \sqcap \forall ManufacturedBy."Sony" \sqsubseteq Product \sqcap \exists.ManufacturedBy$

2.)  $TV \sqcap \forall ManufacturedBy."Sony" \sqsubseteq Product$

3.) Please write the expression for “the Sony TVs that work with all Sony products and work with AIWA Stereos but do not work with Panasonic Stereos.”



## 2.7 Possible Worlds – 50 points

Consider the schema:

`Agrees(Person1 , Person2, Witness)`

Where `Person1`, `Person2`, and `Witness` are from the set `{kim, mike, neil, al, jamal}`

Note that a person may agree with themselves and be the witness as well. So all combinations are possible.

1.) How many distinct databases may be constructed over the schema? (Please, for your own sake, do not attempt to manually count this set – derive a formula.)

2.) Now consider the “integrity constraint” that `Person1`, `Person`, and `Witness` must be distinct. How many distinct database states now?

3.) Finally consider if `Witness` is the key to the relation. How many distinct database states now?

Please marvel at how many database states are squeezed out when we include integrity constrains.