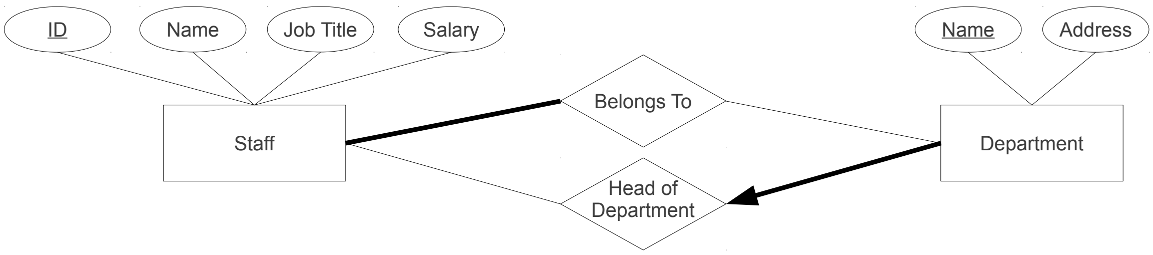
Informatics 1: Data & Analysis
Session 2010–2011, Semester 2

Coursework Assignment: Past Exam Questions, Student Notes

These notes give some answers and comments on the coursework assignment for Inf1-DA. They are not model answers, or necessarily complete solutions; instead, they are intended to help you understand and take note of your tutor's comments on your own solutions. They include some information about common mistakes, and how particular questions were answered by other students.

1. (a) The following is a suitable ER diagram:



This could be refined with a subclass HoD of the Staff entity, but this is not essential.

Some students wrongly put both Name and Address as key fields on Department. This isn't necessary as a department name would be unique within the university, and keys should be minimal.

Some students wrongly put the staff ID of the head of department as an attribute on the HeadOfDepartment relationship. This is unnecessary, as each instance of that relationship already indicates who is the head.

(b) There are several possible additional constraints. For example:

- Every department should have at least one member. This could be incorporated by a participation constraint from Department to the BelongsTo relationship.
- Every head of department should belong to every department of which they are head. This cannot be incorporated in the ER diagram.
- A member of staff should be head of at most one department. This can be incorporated by a key constraint on Staff in the HeadOfDepartment relationship.

(c) Here is a suitable set of table declarations.

```
create table Staff (
id            char(6),
name          char(30),
jobtitle      char(10),
salary        integer,
primary key   (id))

create table Departments (
deptname      char(20),
address       char(40),
hod           char(30) not null,
primary key   (deptname),
foreign key   (hod) references Staff)
```

```
create table BelongsTo (
id          char(6),
deptname    char(20),
primary key (id,deptname),
foreign key (id) references Staff,
foreign key (deptname) references Departments)
```

Notice that because a **foreign key** always references the primary key of another table, declaring that hod in Departments references Staff implicitly means the id field there. This could be made explicit as Staff(id), but it is not essential.

(d) The key constraint on HeadOfDepartment is enforced by the use of a foreign key hod in Department: each department can have no more than one head.

The participation constraint on HeadOfDepartment is enforced by the **not null** declaration on hod: every department must have a head.

The participation constraint on BelongsTo has not been implemented: the declaration does not enforce that every staff member must belong to at least one department.

(e) *Find the name and salary of every head of department.*

Relational algebra:

$$\pi_{\mathsf{name,salary}}(\mathsf{Staff} \bowtie_{\mathsf{id=hod}} \mathsf{Departments})$$

Tuple-relational calculus:

$$\{R \mid \exists S \in \mathsf{Staff}.\ \exists D \in \mathsf{Departments}.\ (R.\mathsf{name} = S.\mathsf{name} \ \wedge \ R.\mathsf{salary} = S.\mathsf{salary} \ \wedge \ D.\mathsf{hod} = S.\mathsf{id})\}$$

SQL:

```
SELECT S.name, S.salary
FROM Staff S, Departments D
WHERE S.id = D.hod
```

(f) *Count the number of heads of department that belong to at least one department of which they are not head.*

```
SELECT COUNT DISTINCT (S.id)
FROM Staff S, Departments D1, Departments D2, BelongsTo B
WHERE D1.hod = S.id AND D2.hod <> S.id
AND B.id = S.id AND B.deptname = D2.name
```

Note that it is not enough that a head of one department should be in another; they must also not be head of that other department.

It's also essential to use **DISTINCT**, so that we don't count duplicates; and to count S.id rather than name or anything else, because two people may share the same name.

2. (a) The XML document is as follows:

```
<stext>
<s>
<w pos="pron"> I </w>
<w pos="verb"> shall </w>
<w pos="verb"> hear </w>
<w pos="prep"> in </w>
<w pos="subst"> heaven </w>
<c> ! </c>
</s>
</stext>
```

(b) <s> is an element annotating sentences.

<w> is an element annotating words.

<c> is an element annotating punctuation.

pos is an attribute whose value carries part-of-speech information.

pron, verb, prep, subst label different parts of speech.

(c) The DTD below is one possibility.

```
<!ELEMENT stext ((s)∗)>
<!ELEMENT s  ((w|c)∗)>
<!ELEMENT w  (#PCDATA)>
<!ELEMENT c  (#PCDATA)>
<!ATTLIST  w  pos CDATA #REQUIRED>
```

Other possibilities would be to use '+' instead of '∗', or to give a list of allowed values for c or for pos.

Using '(w∗,c∗)' for the sentence element is not sufficient, as the comma enforces element ordering. In this case that would only allow sentences to have punctuation at the end.

(d) XPath expressions:

(i) *All punctuation marks*

```
//c/text()
```

(ii) *All verbs*

```
//w[pos='verb']/text()
```

(iii) *All verbs that appear in sentences that contain an exclamation mark.*

```
//s[c='!']/w[pos='verb']/text()
```

3. (a) The *information retrieval task* is to find those documents relevant to a user query from among some large collection of documents.

   The underlying assumptions are:

   (i) There is a large document collection being searched

   (ii) The user seeks information related to a particular query (typically some number of keywords)

   (iii) The task is to find all and only the documents relevant to the query.

   One example is an internet search engine. The document collection is a number of web pages; the information needed is given as a keyword query; and the search engine is to return a list of relevant web pages.

   Notice that the task is to retrieve some documents out of several; not to find individual words or phrases in a document. Information retrieval is also more general than just searching for keywords: in the web search example, pages may be relevant because there are keywords in links which point to them, not in the pages themselves.

   (b) The *recall* is the proportion of those documents in the full collection which are relevant to the query that are returned by the information retrieval system.

   (c) The terms and their definitions are:

   - *TP* is *true positives* — the number of items returned by the system that are relevant to the query.
   - *FP* is *false positives* — the number of items returned by the system that are not relevant to the query.
   - *FN* is *false negatives* — the number of items not returned by the system that are relevant to the query.

   (d) System 1: $TP = 4$; $FP = 1$; $FN = 16$; so precision $P = 4/5$ and recall $R = 1/5$.

   System 2: $TP = 15$; $FP = 5$; $FN = 5$; so precision $P = 3/4$ and recall $R = 3/4$.

   (e) A system that returns all documents has 100% recall, and a system that returns just one document, which is relevant, has 100% precision. Neither of these systems are very useful, so it is vital to take both measures into account.

   (f) The harmonic mean is given by:

$$F = \frac{1}{\frac{1}{2}(\frac{1}{P} + \frac{1}{R})} = \frac{2PR}{P + R}$$

   For system 1, we get $F = 8/25$.

   For system 2, we get $F = 3/4$.

   Thus the F-score evaluates system 2 as the better system.