CMSC 661 – MIDTERM EXAMINATION – 11/1/95

This exam is closed book. Please write your name on the first page of the exam. You have 75 minutes to do all of the questions whose points sum to 100. **Thought for the day:** "If a man neglects education, he walks lame to the end of his life." – Plato.

1 The Datalog model (15 points)

You need to keep information on various software files in a database in order to support a package like the Unix `make` facility. The following describes the situation:

> Each file will have a unique name. For each file, you will know the time that it was last modified, represented as the number of seconds since Bill Gates birth (i.e., *seconds AG*). You will also know the time that a file was last compiled (again, in seconds AG). For each file, you will know the names of the other files that it immediately depends on (e.g., the files it includes or the files which contain definitions of procedures that are called by the code in the file).
>
> You need to have a relation `recompile(File)` that will include a tuple naming each file that needs to be *recompiled* – i.e., that has been modified after it was last compiled or that depends (either directly or indirectly) on a file (F) that has been modified since it (F) was last compiled.

Design a Datalog model for this problem. Specifically, give each of the **extensional database relations** and the **intentional database relations**. For each intentional database relation, give an appropriate datalog definition. You may assume a built-in relation $before(T1, T2)$ that is true if and only if $T1$ and $T2$ are positive integers and $T1 < T2$.

2 Query Languages (15 points)

You have a relational database containing the following four relations:

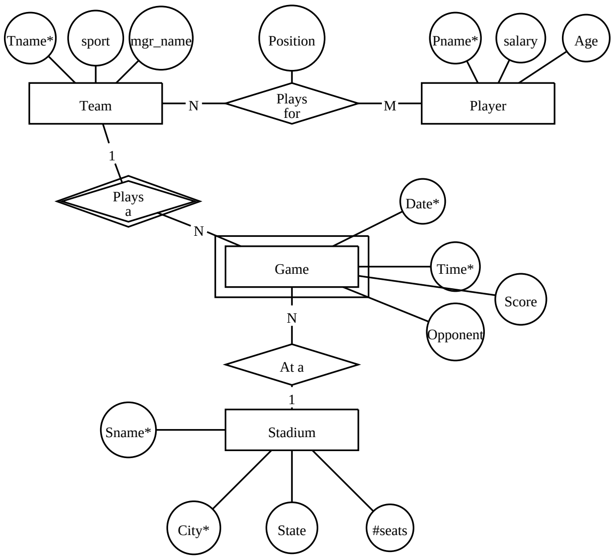
```
Department( Dept#, Dname, Location)
Employee(SSN, Name, Dept#, Job, Salary, Address)
WorksOn(SSN, Project#, PercentTime)
Project(Project#, ProjectDescription, CostCenter)
```

1. Identify the primary key of each relation.
2. Identify any foreign keys in each relation.
3. Give a Datalog query to print the names and SSNs of all employees who don't work on project "P5".
4. Give a relational algebra query to print the names of employees who work on project "P5" for more than 50% of their time.

3 Entity Relationship Diagrams (15 points)

Here is an ER diagram for a sports team database. The keys of each entity are identified with a "*". Convert this to a a set of relations and identify the keys of each relation. Describe the schema informally using a notation like:

```
supplier(Supplier#, supplier_name, status, city)
```



4 Relational Algebra (15 points)

Suppose we have a database consisting of the following relations:

```
faculty(Professor, Department)
teach(Professor, Class)
major(Student, Department)
offers(Department, Class)
enrolled(Student, Class)
```

Interpret the following queries, i.e. state in English what they yield. Assume that each student enrolled at least one class and majors in at least one department. Assume that each professor is a faculty member in one department and teaches at least one class. Recall that $\bowtie$ represents a natural join.

- $\Pi_{department}(faculty \bowtie teaches \bowtie \Pi_{class}(\sigma_{student=\text{``}BillyGates''}(enrolled))$
- $\Pi_{department}\sigma_{professor=\text{``}Turing''}(teaches \bowtie enrolled \bowtie major)$

Write expressions in the relational algebra which answer the following queries.

- "Which professors teach a class in which at least one CSEE major is enrolled?"
- "Which professors teach classes not offered by their own department?"

Assuming the relations above are all in the extensional database, write rules in unrestricted Datalog to answer the following:

- "Faculty teaching at least one class and who have no students in any of their classes who majoring in that faculty member's department."

5 Simple Datalog (10 points)

Suppose we have a Datalog extensional database with the following predicates:

- **person(X)** - true if X is a person.
- **age(X,N)** - true if the age of object X is N.

Write a Datalog rule will find the oldest person in the database. You may define any auxiliary predicates you need. (Hint: use negation).

6 Prolog and Cut (15 points)

Let the predicate $min(N1, N2, Min)$ be true if Min is the smaller of $N1$ and $N2$ (assume all three arguments are integers).

Consider the following definition of $min/3$:

```
min(N1,N2,N1) :- N1=<N2, !.
min(N1,N2,N2).
```

Describe what will happened for each of the following goals:

1. min(1,2,X).
2. min(1,2,2).
3. min(X,3,3).

Write the min/3 predicate without using the cut symbol.

7 Mystery Prolog predicate (15 points)

Consider the following mystery predicate:

```
p(X,[],[X]).
p(X,[First|Rest],[First|NewRest]) :- p(X,Rest,NewRest).
```

- Describe this predicate in a simple sentence.
- Show what each of the following queries would result in:
  - p(h,[a,[b,c],d],X)
  - p(X,Y,[a,b,c,d,e,f,g])
  - p(s,[y,w,c],[s]).
  - p(a,[b],[b,a]).