CS585 - Database Systems

Midterm Exam

Please Complete the Following:

<table>
<thead>
<tr>
<th>Name</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Student ID#</td>
<td></td>
</tr>
<tr>
<td>Location (if remote)</td>
<td></td>
</tr>
</tbody>
</table>

Duration: 1 hour and 15 minutes

Please look over the entire exam first,
You may not have enough time to finish the entire exam,
Select easy questions first, Don’t Panic,
relax, and good luck!
**Problem 1:**
Indicate whether each of the following statement is true or false (T/F):

__ 1. The degree of a relationship set in ER model is the same as the cardinality of the relationship set.
__ 3. The higher level entity set in the specialization of EER model is called owner entity set.
__ 4. Easy implementation is the major advantage of 3-tier architecture over 2-tier architecture.
__ 5. In Oracle ORDBMS, a static method is invoked on object type, not on object instances.
__ 6. Assuming reduction to a pure relational model, the following EER schema can be represented by two relations without resulting in inconsistency.

```
 a
 o  
 b  c
```
Problem 2

Consider the following SQL statements:
CREATE TYPE PointType AS OBJECT (  
x NUMBER,  
y NUMBER);  /
CREATE TYPE LineType AS OBJECT (  
end1 PointType,  
end2 PointType,  
MAP MEMBER FUNCTION something1 RETURN NUMBER);  /
CREATE OR REPLACE TYPE BODY LineType AS  
MAP MEMBER FUNCTION something1 RETURN NUMBER IS  
BEGIN  
RETURN SQRT((end1.x-end2.x)*(end1.x-end2.x) +  
(end1.y-end2.y)*(end1.y-end2.y));  
END;  
END;  /
CREATE TABLE Lines of LineType;  
INSERT INTO Lines  
VALUES( LineType( PointType(3.0, 3.0), PointType(6.0, 7.0) ) );  
INSERT INTO Lines  
VALUES( LineType( PointType(0.0, 0.0), PointType(1.5, 2.0) ) );  
INSERT INTO Lines  
VALUES( LineType( PointType(0.0, 0.0), PointType(6.0, 8.0) ) );
a) What does the member function “something1” do?

b) What would be the result after running the following query?
SELECT * FROM lines L
ORDER BY VALUE(L);

Now, assume that we change our type definitions to:
CREATE TYPE PointType AS OBJECT (  
x NUMBER,   
y NUMBER);  
/
CREATE TYPE LineType AS OBJECT (  
end1 PointType,  
end2 PointType,  
ORDER MEMBER FUNCTION something2(line lineType)  
RETURN INTEGER);  
/
CREATE OR REPLACE TYPE BODY LineType AS  
ORDER MEMBER FUNCTION something2(line lineType) RETURN INTEGER IS  
BEGIN  
IF SQRT((end1.x-end2.x)*(end1.x-end2.x) +  
(end1.y-end2.y)*(end1.y-end2.y)) >
SQRT((line.end1.x-line.end2.x)*(line.end1.x-line.end2.x) +
(line.end1.y-line.end2.y)*(line.end1.y-line.end2.y))
THEN
RETURN 1;
ELSIF SQRT((end1.x-end2.x)*(end1.x-end2.x) +
(end1.y-end2.y)*(end1.y-end2.y)) <
SQRT((line.end1.x-line.end2.x)*(line.end1.x-line.end2.x) +
(line.end1.y-line.end2.y)*(line.end1.y-line.end2.y))
THEN
RETURN -1;
ELSE RETURN 0;
END IF;
END;
END;
/
CREATE TABLE Lines of LineType;
INSERT INTO Lines
VALUES( LineType( PointType(0.0, 0.0), PointType(3.0, 4.0) ) );
INSERT INTO Lines
VALUES( LineType( PointType(3.0, 3.0), PointType(4.0, 5.0) ) );
INSERT INTO Lines
VALUES( LineType( PointType(0.0, 0.0), PointType(1.0, 1.0) ) );
INSERT INTO Lines
VALUES( LineType( PointType(0.0, 0.0), PointType(6.0, 8.0) ) );

c) What would be the result after running the following query?
SELECT *
FROM lines L
WHERE VALUE(L) = (LineType(PointType(3.0,4.0),PointType(4.0, 6.0)));
d) What would be the result after running the following query?

```
SELECT MAX(VALUE(L))
FROM lines L;
```
Problem 3

(a). Consider the ER diagram below

(Note: c is a derived attribute)

Table: A
\[
\begin{array}{|c|c|}
\hline
a & d \\
\hline
\end{array}
\]

Table: R
\[
\begin{array}{|c|c|c|c|}
\hline
a & b & r & q \\
\hline
\end{array}
\]

Table: B
\[
\begin{array}{|c|c|}
\hline
b & c \\
\hline
\end{array}
\]
(I). Is the given reduction to tables correct? If not, correct it.

(II). What is the primary key of the table corresponding to the relationship R?
(b). Consider the following entity set.

Assume each employee has a unique SSN# and a unique Name. Determine which of the following is a superkey and which is a candidate key and mark it.

<table>
<thead>
<tr>
<th></th>
<th>Super Key</th>
<th>Candidate Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSN#, Age</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSN#, Name, Age</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name, Age</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSN#</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Age</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Problem 4:

Consider the relational conceptual database schema below for keeping track of course registration of students:

COURSES (Code, Title, Dept)
Registered (Code, SSN)
STUDENTS (SSN, Name, Dept, GPA)

Here, COURSES contains a tuple for each course, recording its code, title, and department offering the course. STUDENTS records the SSN, name, home department, and GPA of the students. REGISTERED keeps the relation between courses and students.

(a) Retrieve the name of each student who registered in the course titled “Database Systems”. (2 points)
(b) Retrieve the title of the course along with the number of students who registered in this course in the descending order of registered student number. (3 points)

(c) Retrieve the name of student(s) who earned the maximum GPA in each department that offers more than 30 courses. (3 points)

(Hint: You can use nested queries in the “FROM” clause of an SQL statement.)
(d) Describe what the following SQL statement does? (6 points)

```sql
SELECT UNIQUE(c.title)
FROM courses c, students s, registered r
WHERE r.code = c.code AND r.SSN = s.SSN AND c.code = (SELECT c1.code
FROM courses c1, students s1, registered r1
WHERE r1.code = c1.code AND r1.SSN = s1.SSN
GROUP BY c1.code
HAVING COUNT(c1.code) >= (SELECT MAX(COUNT(c2.code))
FROM courses c2, students s2, registered r2
WHERE r2.code = c2.code AND r2.SSN = s2.SSN AND c2.dept <> s2.dept
GROUP BY c2.code) )
```
Problem 5

Reduce given ER diagram to relations using pure relational model (i.e., No Object Oriented or Object Relational). Be sure to identify all primary and foreign keys.
Problem 6

What are the requirements for using ODBC to make connection to a database, and what are the advantages of using JDBC over ODBC?