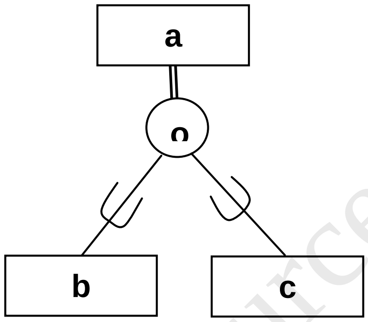
Problem 1: (20 points)

Indicate whether each of the following statements is true or false (T/F):
`

_F_ The degree of a relationship set in ER model is the same as the cardinality of the relationship set.

_F_ In EER model, the higher level entity set in specialization is called the owner entity set.

_F Assuming reduction to a pure relational model, the following EER schema can be represented by two relations without resulting in inconsistency.



__F_ A relationship is an association among two or more attributes.

__T_ One owner entity is associated with one or more weak entities, but each weak entity has a single owner.

_F__ An integrity constraint is a condition specified on a relation instance.

__F_ A foreign key could not refer to the same relation.

__F_ By default, a constraint is checked at the end of every transaction that could lead to a violation.

__T_If a primary key comprises of two attributes, none of their values can be null.

Problem 1 (Cont'd)

_T__ If "group by" keyword is omitted, the whole table is treated as a group.

__F_ If we compare two *null* values using <, >, =, and so on, the result is always true.

__F_ In an R-tree, an optimal split of an MBR is possible in linear time with respect to the number of MBRs within it.

_T__In an OODBMS, there is no need to explicitly come up with primary keys.

__T_ A subquery can reference attributes in the outer query.

_T__ An attribute declared as UNIQUE can have NULL as its value.

__F_ Quad trees have an advantage over kd-trees in that they keep the tree balanced therefore keeping the search efficient at all times.

_T__ In an R-Tree, we may not know exactly which branch of the tree to follow to find an object stored in the database.

_F__OODBMS is not suitable technology for applications running on mobile devices.

_F__In an R-tree, an optimal split of an MBR is possible in linear time with respect to the number of MBRs within it.

_F__Using views one can update derived attributes.

Problem 2: (16 points)

Consider the following schema:

Suppliers(sid: integer, sname: string, address: string)
Parts(pid: integer, pname: string, color: string)
Catalog(sid: integer, pid: integer, cost: real)

The Catalog relation lists the prices charged for parts by Suppliers. Write the following queries in SQL:

1. Find the pnames of parts for which there is some supplier.

```
SELECT P.pname
FROM Parts
Where  P.pid IN (SELECT C.pid
                 FROM Catalog)
```

2. Find sids of suppliers who charge more for some part than the average cost of that part (averaged over all the suppliers who supply that part).

```
    SELECT DISTINCT C.sid
    FROM Catalog C
    WHERE C.cost > ( SELECT AVG (C1.cost)
    FROM Catalog C1
    WHERE C1.pid = C.pid )
```

3. Find sids of Suppliers who supply a red part and a green part

```
    SELECT DISTINCT C.sid
    FROM Catalog C, Parts P
    WHERE C.pid = P.pid AND P.color = 'Red'
    INTERSECT
    SELECT DISTINCT C1.sid
    FROM Catalog C1, Parts P1
    WHERE C1.pid = P1.pid AND P1.color = 'Green'
```

4. For every supplier that only supplies green parts, print the name of the supplier and the total number of parts that she supplies.

```
    SELECT S.sname, COUNT(*) as PartCount
    FROM Suppliers S, Parts P, Catalog C
    WHERE P.pid = C.pid AND C.sid = S.sid
    GROUP BY S.sname, S.sid
    HAVING EVERY (P.color='Green')
```