Problem 1 – Multiple Choice

1. Which is the correct explanation of "physical data independence" in the three levels of abstractions of a DBMS?

A. Independence between the logical level and the physical level.

B. Independence between the view level and the logical level.

C. Independence between the physical level and the network level.

D. Independence between the logical level and the network level.

Problem 1 – Multiple Choice

2. Given a relation R, which of the following can NOT uniquely identify tuples in R?

A. Primary key of R.

B. Super key of R.

C. Foreign key of R.

D. Candidate key of R.

Problem 1 – Multiple Choice

3. Given an E-R schema, which of following operation in reduction to relations is a proper operation?

A. For one-to-one relationship sets, add extra attributes to both of the tables corresponding to the two entity set, containing the primary key of the other side.

B. For one-to many relationship sets, add an extra attribute to the many side, containing the primary key of the one side.

C. Use single relation to store as much information as possible even if we need to repeat an information to make sure we will not lose any information.

D. Name attributes by a name you can find an attribute fast without consider whether others can understand or not.

Problem 1 – Multiple Choice

4. What is the function of following relational algebra expression?

$$\sigma_{(Department \neq COMP \vee Department \neq ELEC) \wedge (\neg(CGA=8.34 \vee CGA=12))}(Students)$$

A. Select the information for the students who are from the COMP or the ELEC department and whose CGA is neither 8.34 nor 12.

B. Select the information for the students who are from the COMP or the ELEC department and whose CGA is 8.34 or 12.

C. Select the information for every student as the selection condition is always true.

D. Select nothing as the selection condition is always false.

Problem 1 – Multiple Choice

5. Given a relation r defined over the schema $R = \{A, B, C\}$ where the attribute A is the primary key, which of the following is false?

A. The number of tuples in $\pi_{AB}(r)$ should be equal to the number of tuples in $\pi_{A}(r)$

B. The number of tuples in $\pi_{ABC}(r)$ should be equal to the number of tuples in $\pi_{A}(r)$

C. The number of tuples in $\pi_{A}(r)$ should be equal to the number of tuples in $\pi_{B}(r)$

D. The number of tuples in $\pi_{B}(r)$ should be equal to the number of tuples in $\pi_{BC}(r)$

Problem 1 – Multiple Choice

6. Which of the following is true about the weak entity?

A. A weak entity implies existence dependency.

B. A weak entity has a key.

C. Existence dependency implies a weak entity.

D. A weak entity can be identified with its own attributes.

Problem 1 – Multiple Choice

7. Which of the following operation is equivalent to the natural join of two relations *r* and *s* when the relational schemas of *r* and *s* have no common attributes?

A. Set difference.

B. Set union.

C. Set intersection.

D. Cartesian product.

Problem 1 – Multiple Choice

8. Which of the following statements is true about views in relational database?

A. When a view is declared, the DBMS will store a table for the view.

B. We can always delete a tuple in a view.

C. We can always select a tuple in a view.

D. We cannot define a view on more than one table.

Problem 1 – Multiple Choice

9. Which SQL statement is used to insert new data in a database?

A. INSERT TO ...

B. ADD RECORD ...

C. ADD TO ...

D. INSERT INTO ...

Problem 1 – Multiple Choice

10. Which SQL statement is used to return only different values?

A. SELECT UNIQUE ...

B. SELECT DISTINCT ...

C. SELECT DIFFERENT ...

D. SELECT ONLY DIFFERENT ...

Problem 1 – Multiple Choice

11. Which one is not an Aggregate Function in SQL?

A. AVG

B. MIN

C. MAX

D. UNIQUE

Problem 1 – Multiple Choice

12. Which one is correct about foreign keys?

A. Values of a foreign key cannot be null.

B. Values of a foreign key must be unique.

C. One table can have multiple foreign keys.

D. Values of a foreign key cannot be updated.

Problem 1 – Multiple Choice

13. Consider relation $R(A, B, C, D, E)$ with the following functional dependencies: $D \rightarrow C$, $CE \rightarrow A$, $D \rightarrow A$, $AE \rightarrow D$. Which of the following is a key of R?

A. CDE

B. BCE

C. BD

D. A

Problem 1 – Multiple Choice

14. Consider the relation $R(A, B, C, D)$ with the following functional dependencies: $A \rightarrow B$, $B \rightarrow C$. Which of the following is the attribute closure of A (i.e. $A+$)?

A. AB

B. AC

C. ABC

D. $ABCD$

Problem 1 – Multiple Choice

15. Consider the following set of functional dependencies $F = \{A \rightarrow B, A \rightarrow C, B \rightarrow C, B \rightarrow BC\}$. Which of the following is the canonical cover of F (i.e. F_C)?

A. $\{A \rightarrow B, A \rightarrow C, B \rightarrow C\}$

B. $\{A \rightarrow C, B \rightarrow C\}$

C. $\{A \rightarrow BC\}$

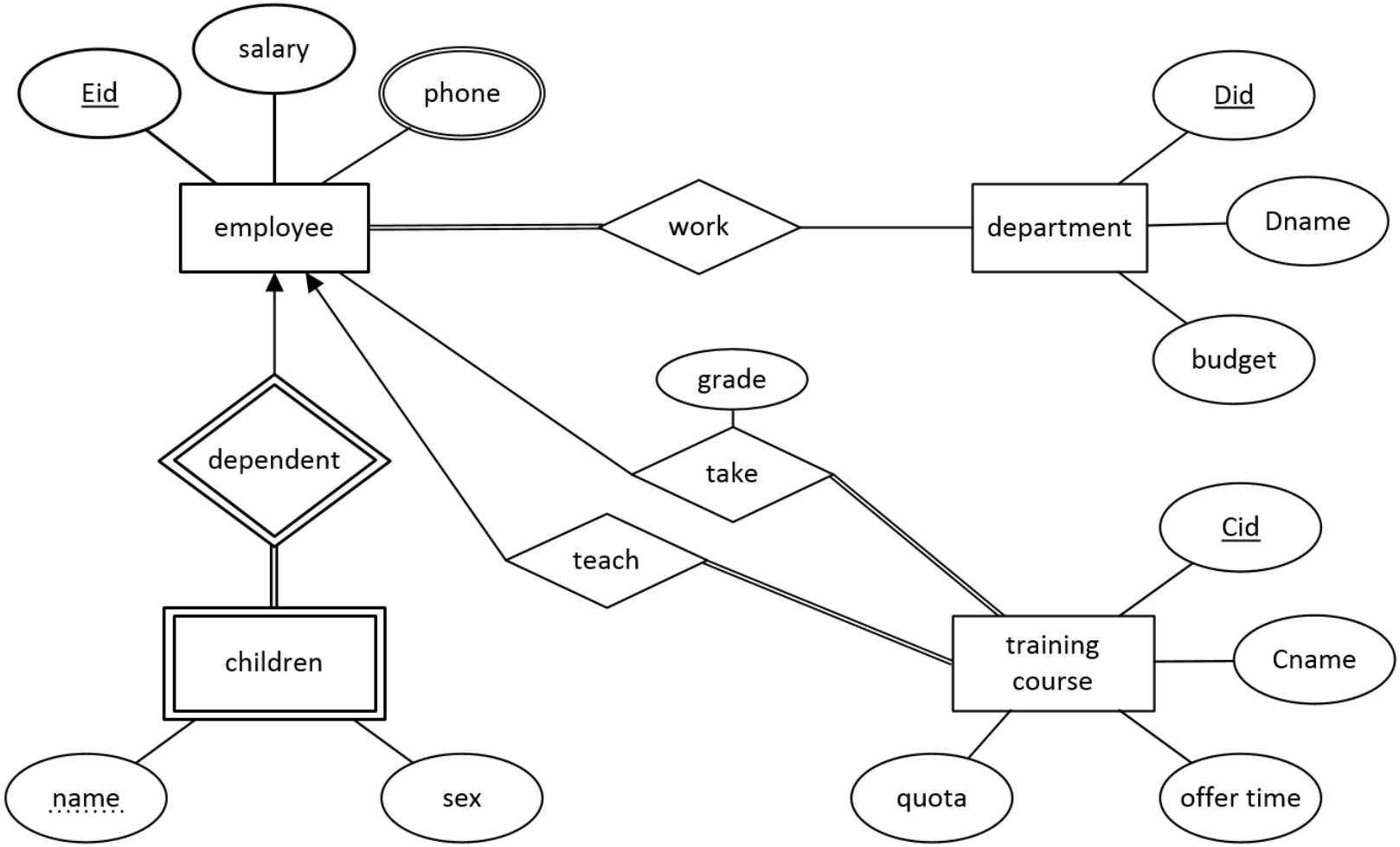
D. $\{A \rightarrow B, B \rightarrow C\}$

Problem 2 – ER Diagram Design

You are hired to design a database for the personnel office of a company. The database keeps information about **employees**, **children** of employees (with name and age as attributes), **departments**, and **training courses** provided. The details of the database are as follows:

- An **employee** is identified by Eid. Each **employee** has a salary and at least one phone number(s). Every **employee** must work in one or more **department(s)**;
- A **child** has a name and a sex. A **child** is dependent on his or her parent. This means a **child** must be identified uniquely by his or her name when the parent (who is an **employee**; assume that only one parent works for the company) is known;
- A **department** is identified by Did. Each **department** has a Dname and a budget;
- A **training course** is identified by a Cid. Each **training course** also has a name, an offer time and a quota;
- Each **training course** is taught by exactly one **employee** and taken by at least one **employee**;
- The grades of **employees** who took **training courses** are also stored.

Problem 2 – ER Diagram Design



Problem 3a – Retinal Algebra

Given the four relational schemas:

- Employee(*person_name, street, city*)
- Works(*person_name, company_name, salary*)
- Company(*company_name, city*)
- Manages(*person_name, manager_name*)

- Formulate the following queries into the corresponding *relational algebra* expressions.

Employee(*person_name*,*street*,*city*)

Works(*person_name*,*company_name*,*salary*)

Company(*company_name*,*city*)

Manages(*person_name*, *manager_name*)

1. Find the names of all persons who live in "New York".

$$\pi_{person\_name}(\sigma_{city="New\ York"}(employee))$$

Employee(*person_name,street,city*)

Works(*person_name,company_name,salary*)

Company(*company_name,city*)

Manages(*person_name, manager_name*)

2. Find the names of all persons who live in "London" but not live in "Baker Street".

$$\pi_{person_{name}}(\sigma_{city="London"}(employee) - \sigma_{street="Baker\ Street"}(employee))$$

Or

$$\pi_{person\_name}(\sigma_{city="London" \wedge street \neq "Baker\ Street"}(employee))$$

Employee(*person_name,street,city*)

Works(*person_name,company_name,salary*)

Company(*company_name,city*)

Manages(*person_name, manager_name*)

3.Find the names of all managers who has employee(s) earn less than $5,000

$$\pi_{manager\_name}\left(\sigma_{works.salary<"5000"}\ \left(works\ \bowtie_{works.person\_name=manages.person\_name} manages\right.\right.$$

Employee(*person_name*,*street*,*city*)

Works(*person_name*,*company_name*,*salary*)

Company(*company_name*,*city*)

Manages(*person_name*, *manager_name*)

4. Find the employees that live in Shenzhen but hold a job in ALL companies located in Hong Kong

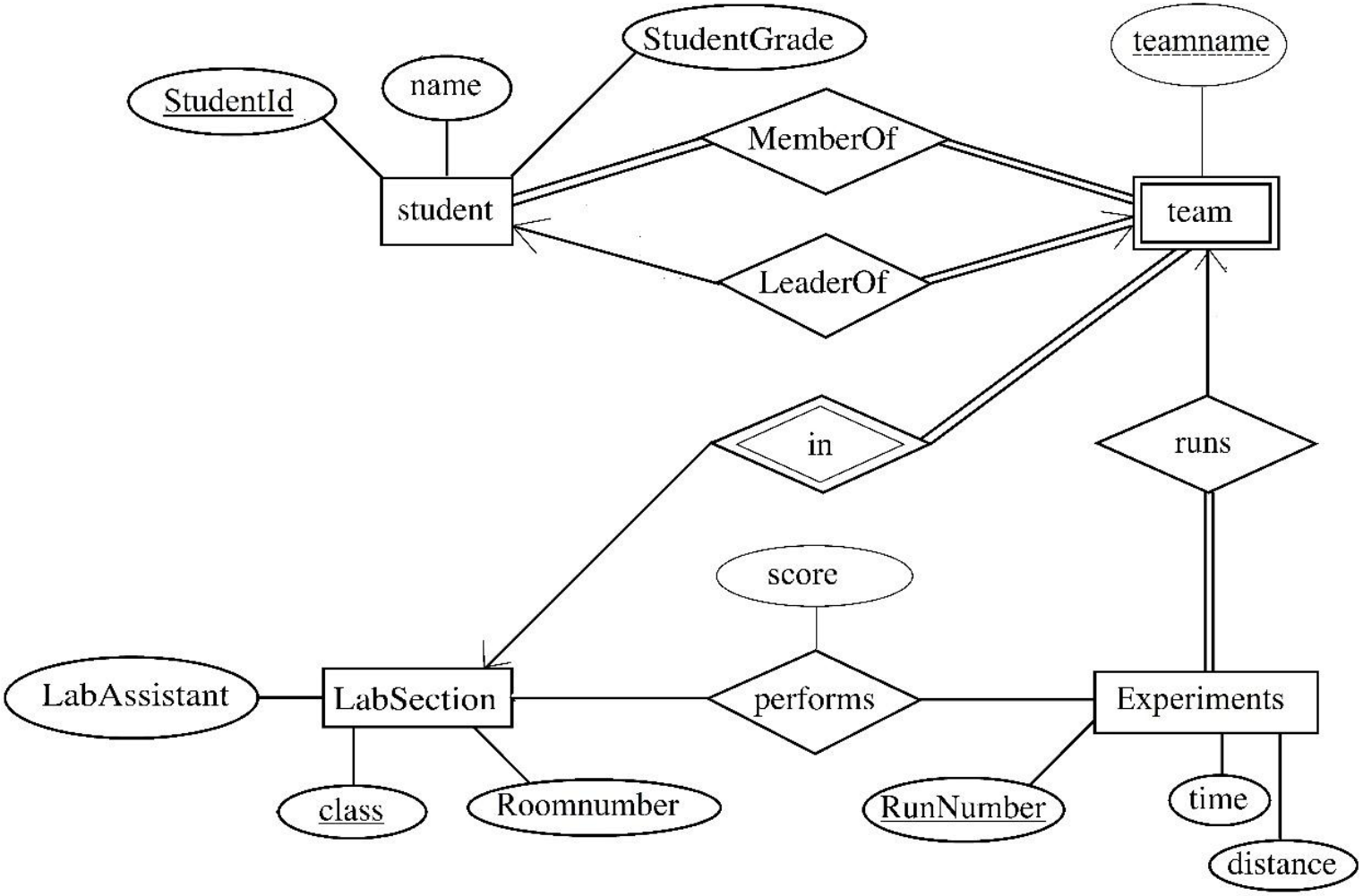
$$\pi_{person\_name}\left(\sigma_{city=\text{“}Shenzhen\text{”}}(employee)\right)$$

$$\cap\ (\pi_{person\_name,company\_name}(works)/\pi_{compa}$$

Problem 3b – Retinal Schema

Convert the following E-R Diagram into relational schemas and underline clearly the primary keys with a single line and underline the foreign keys with a double line. For example, one schema is student(*StudentId, name, StudentGrade, class, teamname*).

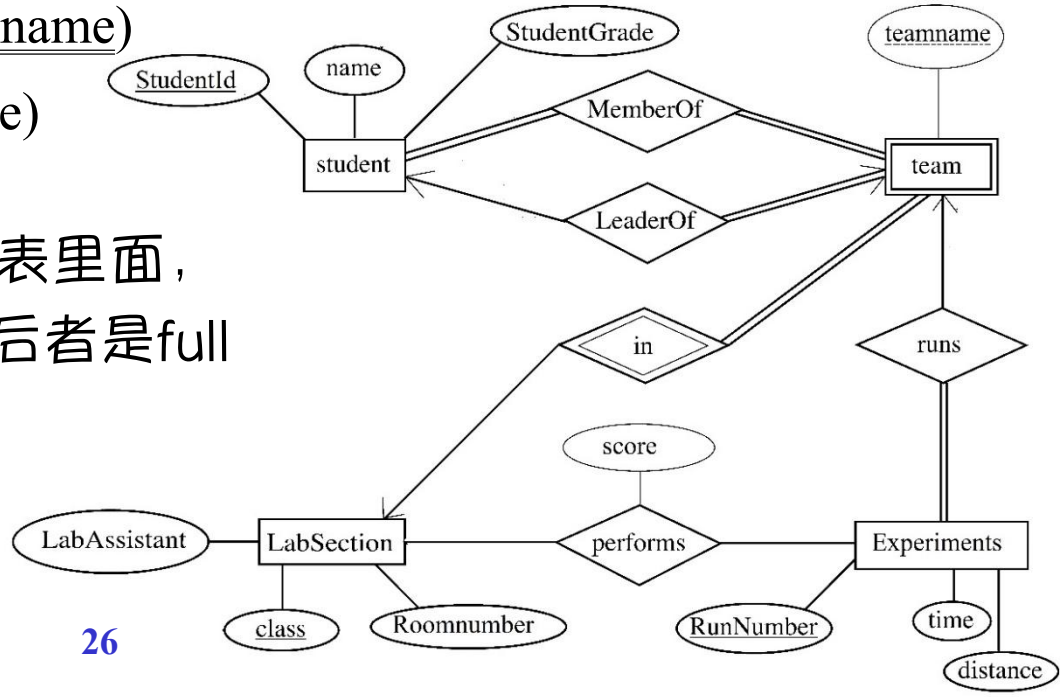
Problem 3b – Retinal Schema



Problem 3b – Retinal Schema

- student(StudentId, name, StudentGrade, *class, teamname*)
- LabSection(class, Roomnumber, LabAssistant)
- Experiments(RunNunber, time, distance, *class, teamname*)
- team(class, teamname)
- MemberOf(StudentId, class, teamname)
- performs(class, RunNumber, score)

这里的member of 放到了student表里面，
实际上放到team表里更好，因为后者是full participation



Problem 4 – SQL

Consider the following Supplier-Part-Project database

- Suppliers: S(*SID*, *SNAME*, *SCITY*)
- Parts: P(*PID*, *PNAME*, *COLOR*)
- Projects: J(*JID*, *JNAME*, *JCITY*)
- Shipment: SPJ(*SID*, *PID*, *JID*, *QTY*).

The significance of an SPJ record is that the specified supplier (*SID*) supplies the specified part (*PID*) to the specified project (*JID*) in the specified quantity (*QTY*). Each shipment must have a specified quantity. Note that the same supplier may supply different parts to the same project.

Write the following queries in SQL. State your further assumptions if necessary.

Suppliers:S(*SID*,*SNAME*,*SCITY*)
Parts:P(*PID*,*PNAMECOLOR*)
Projects:J(*JID*,*JNAMEJCITY*)
Shipment: SPJ(*SID, PID, JID, QTY*).

1. Use the CREATE TABLE statement for creating the SPJ table. Include all the referential integrity constraints in the table. Assume that all attributes are integers.

- ```
  CREATE TABLE SPJ
  ( SID INTEGER NOT NULL,        (NOT NULL OPTIONAL)
    PID INTEGER NOT NULL,        (NOT NULL OPTIONAL)
    JID INTEGER NOT NULL,        (NOT NULL OPTIONAL)
    QTY INTEGER NOT NULL,
    PRIMARY KEY (SID, PID, JID),
    FOREIGN KEY (SID) REFERENCES S,
    FOREIGN KEY (PID) REFERENCES P,
    FOREIGN KEY (JID) REFERENCES J);
  ```
  ```

Suppliers:S(*SID*,*SNAME*,*SCITY*)
Parts:P(*PID*,*PNAMECOLOR*)
Projects:J(*JID*,*JNAMEJCITY*)
Shipment: SPJ(*SID, PID, JID, QTY*).

2. Create a view "LONDON_SUPPLIERS" which only contains the names of suppliers in London.

```
CREATE VIEW LONDON_SUPPLIERS AS (
    SELECT SNAME FROM S WHERE SCITY = 'London');
```

Suppliers:S(*SID,SNAME,SCITY*)
Parts:P(*PID,PNAMECOLOR*)
Projects:J(*JID,JNAMEJCITY*)
Shipment: SPJ(*SID, PID, JID, QTY*).

3.Get the (distinct) colors of the parts supplied by suppliers in London to projects in Paris. Display the results in alphabetical order.

- select distinct P.COLOR
  from S,P,J,SPJ
  where (S.SCITY = 'London') and
  (J.JCITY = 'Paris') and
  (S.SID = SPJ.SID) and
  (P.PID = SPJ.PID) and
  (J.JID = SPJ.JID)
  order by P.COLOR;

- You can use in operation to do joins, but you can't use '=' to indicate in

Suppliers:S(*SID*,*SNAME*,*SCITY*)
Parts:P(*PID*,*PNAMECOLOR*)
Projects:J(*JID*,*JNAMEJCITY*)
Shipment: SPJ(*SID, PID, JID, QTY*).

4. Get the total number of parts supplied by supplier with identify S1 to each project.

```
select JID, count(distinct PID)
        from SPJ
        where SID = 'S1';  (or where SID = S1)

OR

select JID, SUM(QTY)

From SPJ

where SID = S1 (or where SID = 'S1')

group by JID
```

Suppliers:S(*SID,SNAME,SCITY*)
Parts:P(*PID,PNAMECOLOR*)
Projects:J(*JID,JNAMEJCITY*)
Shipment: SPJ(*SID, PID, JID, QTY*).

5. Get project numbers without duplication for projects that are supplied by every supplier in London.

```
select distinct JID
        from SPJ spj1
        where not exists ((select SID from S where SCITY = 'London) except
                          (select SID from SPJ spj2 where spj1.JID =
spj2.JID));
```

Can't use JName instead of JID, because it's not a key.