Part I [20 pts.] (1pt each) Fill in the blanks with the best answer.

1. The three-level of data abstraction in database system includes: physical level, logical leval and view level.

2. A relational schema R is in first/1 normal form if the domains of all attributes of R are atomic.

3. It is possible to use Armstrong' s axioms to prove the union rule,that is,if $\alpha \rightarrow \beta$ holds and $\alpha \rightarrow \gamma$ holds,then $\alpha \rightarrow \beta\gamma$ holds.

4. A B+-tree of order 4 has following properties:each leaf nodes has between 2 and 3 values.Each non-leaf node other than root has between 1 and 3 values.The root must have at least 2 children.

5. After parsing and translation, the SQL query will be translated into its internal form: relational algebra expression.And then,the query-execution engine will take the execution plan which contains detailed information on how a particular query or a set of queries will be executed.

6. Let E1 and E2 be relational-algebra expression,L1 and L2 be attributes of E1 and E2 respectively.Then, $\prod_{L1 \cup L2}(E1 \bowtie_{\theta} E2)$ = $(\prod_{L1}(E1)) \bowtie_{\theta} (\prod_{L2}(E2))$

7. Secondary indices must be a dense index,which has an index entry for every search-key value and a pointer to every record in the file.

8. The scheme of handling bucket overflows of hash function in DBMS is called closed hashing,that is,the overflow buckets of a given bucket are chained together in a linked list.

9. We assume that a relation has a B+ tree index of height 5,each disk block contains 4 tuples of the relation,and there are 10 tuples satisfying the query.To process the query,database management system have to access disk at least 8 times in the best case.

10. A transaction has the following properties:atomicity,consistency,isolation and durability.

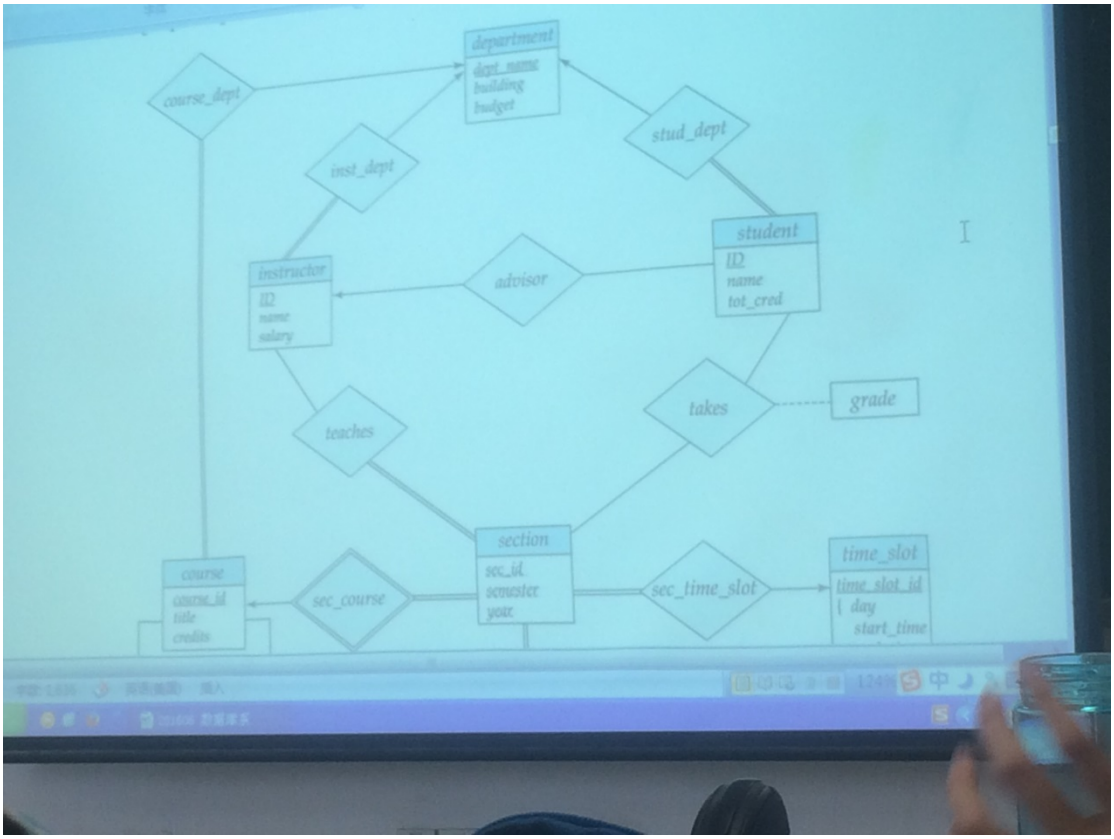
11. The immediate database modification scheme allows database modification to be output to the database while the transaction is still in active state.

12. Since a failure may occur while a update is taking place,log must be written out to stable storage before the actual update to database to be done.

13. Two-phase locking protocol requires that each transaction issue lock and unlock requests in two phases:growing phase and shrinking phase.

14. A schedule S is cascadeless if for each pair of transactions T_i and T_j in S such that T_j reads a data item previously written by T_i,the commit operation of T_i appears before (before/after) the read operation of T_j.

Part II [80 pts.] Answer the following question.



P168

a)[3points]list the entity sets and their primary keys

1.classroom, primary key:(building,room number)
2.department, primary key:(dept_name)
3.course, primary key:(course_id)
4.instructor, primary key:(ID)
5.student, primary key:(ID)
6.section, primary key:(course_id,sec_id,sem,year)
7.time_slot, primary key:(time_slot_id)

b)[3points]give appropriate relation schemas for the entity sets.

1.classromm(building,room_number,capacity)
2.department(dept_name,building,budget)
3.course(course_id,title,credits)
4.instructor(ID,name,salary)
5.student(ID,name,tot_cred)
6.section(course_id,sec_id,sem,year)
7.time_slot(time_slot_id,day,start_time,end_time)

c)[4pionts]list the relationship sets and their primary keys.

1.Teaches, primary key:(ID,course_id,sec_id,semester,year)
2.takes, primary key:(ID,course_id,sec_id,semester,year)
3.prereq, primary key:(course_id,prereq_id)
4.advisor, primary key:(s_ID)
5.sec_course, primary key:(course_id,sec_id,semester,year)
6.sec_time_slot, primary key:(course_id,sec_id,semester,year)
7.sec_class, primary key:(course_id,sec_id,semester,year)
8.inst_dept primary key:(ID)
9.stud_dept primary key:(ID)

10.course_dept primary key:(course_id)

d)[4points]give appropriate relation schemas for the relationship sets.

1.teachers(ID, course_id, sec_id, semester, year)
2.takes(ID, course_id, sec_id, semester, year)
3.prereq(course_id, prereg_id)
4.advisor(s_ID, ID)
5.sec_course(course_id ,sec_id, semester, year)
6.sec_nme_slot(course_id, sec_id,semester, year, nme_slot_id)
7.sec_class(course_id ,sec_id, semester, year, building, room_number)
8.inst_dept(ID, dept_name)
9.stud_dept(ID, dept_name)
10.course_dept(course_id, dept_name)

e)[4points]optimize the relation schemas you given above, i.e., remove schema redundancies and incorporate some schemas if necessary.

~~inst_dept~~ ,instructor = {ID, name, dept_name, salary}
~~stud_dept~~,student = {ID, name, dept_name, tot_cred}
~~course_dept~~ ,course = {course_id, title, dept_name, credits}
~~sec_course~~
~~sec_class~~ ,section = {course_id, sec_id, semester, year, building, room_number}
~~sec_time_slot~~ ,section = {course_id, sec_id, semester, year, building, room_numbe, time_slot_id}

f)[4points]Give a SQL DDL definition for the relation schemas:

instructor, teaches, department, course

with appropriate data-type defined in standard SQL. Identify referential-integrity constrains that should hold, and include them in the DDL definition.

```
create table instructor
(ID varchar (5),
name varchar(20) not null,
dept_name varchar(20),
salary numeric(8,2),
primary key(ID),
foreign key(dept_name) references department);

create table department(
dept_name varchar(20),
building varchar(15),
budget numric(12, 2) check(budget >0),
primary key (dept_name) );

create table course
(course_id varchar(8),
title varchar(50),
dept_name varchar(20),
```

长度为8，小数位2

credits numeric(2,0) check (credits>0),
primary key(course_id),
foreign key(dept_name) references department
on delete set null);

丢弃这一行的值

department 中删对应里null　　on delete cascade

2. Let R=(A, B, C)be a relation with functional dependency F={A→BC,B→C,A→B,AB→C}and key ={A}
a)why we need 3NF
There are some situations where BCNF is not dependency preserving and efficient
checking for FD violation on update is important

3NF allows some redundancy btut functional dependencies can be checked on individual relations without computing a join. There is always a lossless-join, dependency-preserving decomposition into 3NF　P197

b)Please decompose this relation into 3NF

First, we compute the canonical cover
Combine A→BC and A→B into A→BC, we get
{A→BC, B→C, AB→C}
A is extraneous in AB→C, we get
{A→BC, B→C}
The canonical cover is {A→B,B→C}

Second, we generates new schemas
R_1=(A,B)
R_2=(B,C)
Since R_1 contains a key A, the decomposition is done

3.[24points]Consider the following relational database
employee(employee-name, street, city)
works(employee-name, company-name, salary)
company(company-name, city)
manages(employee-name, manager-name)
Answer:(Note : We will use S for SELECT, P for PROJECTION, * for NATURAL JOIN, - for SET DIFFERERNCE, F for AGGREGATE FUNCTION)
a)[4points] Find the names of all employees who live in BEIJING.
(relational algebra)
$P_{employee\text{-}name}(S_{city}="BEIJING"(employee))$　$\Pi_{employee\_name}(\sigma_{city='BEIJING'}(employee))$
b)[4points] Find the names and cities of residence of all employees who work for XYZ Bankand have more than $8,000 salary. (relational algebra)
$P_{person\text{-}name,city}(S_{(company\text{-}name="XYZ"\wedge salary>8000)}works*employee)$
c)[4points] Find the name of company that has the most employees.
select company-name
from works
group by company-name
having count (distinct employee-name)>=all
(select count (distinct employee-name)

```
select company_name
from works
group by company_name
having count(employee_name)>=
    all(select count(employee_name)
        from works
        group by company_name)
```

from works
group by company-name)

d)[4points] Find the names of all employees in this database who live in the same city as the company for which they work (SQL)

select employee-name
from works, employee, company
where works.emplyoee-name = emloyee.employee-name
and works.Company-name=company.Company-name
and employee.City=company.City

e)[4points] Find the names, street address, and cities of residence of all employees whose manager lives in SHANGHAI.(SQL)

select e1.employess-name, e1.street, e1.city
from manages, employee e1, employee e2
where g.employee-name = e1.employee-name
and g.manager-name = e2.employee-name
and e2.city = "SHANGHAI"

f)[4points] Give all managers in this database a 12 percent salary raise.(SQL)

update works
set salary = salary *1.12
where employee-name in
(select manager-name
from manages)

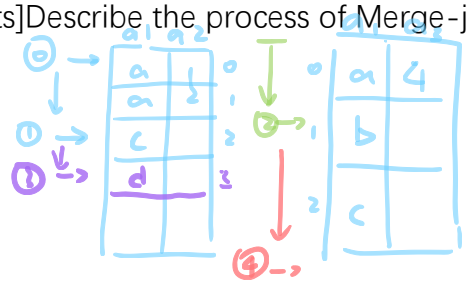
4.[16 points] Query Processing , Optimization and Transaction

a)[4points] *student(ID, name, dept_name, tot,cred)* is a relation instance of 10000 tuples. Each disk block contains 100 tuples, student has a primary index on attribute ID and a secondary index on attribute *name*. The primary index is a B+ tree of height 5. The secondary index is a B+ tree of height 4.

A user sends a query "select * from student where name=WangMing" and retrieves 3 tuple. Please compute the cost of this query, i.e. the number of block transfers from disk and the number of seeks.

overhead:
visiting the index on ID cost 4 block transfers and 4 seeks;
retrieving 3 tuple cost 3 block transfers and 3 seeks;
total cost : 7 block transfers and 7 seeks;

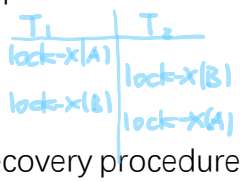
b)[4points]Describe the process of Merge-join. P324



①将 0,1 条放入 S_0
②0 条和 S_0 中(0,1条)依次合并
③将 3 放入 S_1 (已clean)
④将 2 条和 S_1 中(3)合并

c)[4points] Please describe the two-phase locking protocol and prove that it ensures conflict-serializable schedules and does not ensure freedom from deadlocks

假设无死锁，即有个T的锁操作可提前，而后



d)[4points] Below we show some log of a DBMS, please describe the recovery procedure using deferred database modification

取消

(a)	(b)	(c)
<T0 start>	<T0 start>	<T0 start>
<T0, A, 950>	<T0, A, 950>	<T0, A, 950>
<T0, B, 2050>	<T0, B, 2050>	<T0, B, 2050>
<T0 commit>	<T0, commit>	
<T1 start>	< T1 start>	
<T1, C, 600>	<T1, C, 600>	
<T1 commit>		
Redo T0 Redo T1	Redo T0 Undo T1	Undo T0

(a)(b)(c)是3道题

Redo 已Do
Undo 未Do