Introduction to Database Management Systems
Practice Midterm

1. Which of the following are valid (legal) instances of the SQL Table Treatment. The instance must satisfy all constraints. Assume the Physician table was created and populated first and remains unchanged. **4pt each. Circle the letter of each correct answer.**

create table Treatment
(PhysicianID int **not null** ,
PatientID int **not null** ,
Date date **not null** ,
Diagnosis varchar(30),
Location char(5),
primary key(PhysicianID,PatientID,Date),
check (PatientID <> PhysicianID),
foreign key PhysicianID references Physician,
check (Date > any (select Hired from Physician))

create table Physician
(ID int **not null** ,
Hired date,
primary key ID)

Physician

ID	Hired
1	1/1/01
2	1/1/01
3	3/3/03
4	4/4/04
4	4/4/04
5	5/5/05

(a)

PhysicianID	PatientID	Date	Diagnosis	Location
1	2	1/1/02	Glaucoma	Bloor
2	1	1/1/02	Flu	Bloor
3	4	2/2/02	Chicken Pox	Yonge
1	4	3/3/02	Null	Bloor

(b)

PhysicianID	PatientID	Date	Diagnosis	Location
1	3	1/1/02	Chicken Pox	Bloor
2	1	1/1/02	Measles	Bloor
3	2	1/1/02	Chicken Pox	Bloor
3	1	1/1/02	Null	Bloor

(c)

PhysicianID	PatientID	Date	Diagnosis	Location
5	1	1/1/02	Null	Bloor
2	1	1/1/02	Measles	Yonge
7	1	1/1/02	Chicken Pox	Null
4	1	1/1/02	Null	Null

(d)

PhysicianID	PatientID	Date	Diagnosis	Location
1	1	3/3/02	Chicken Pox	Yonge
1	2	1/1/02	Flu	Null
1	3	2/2/02	Measles	Yonge

(e)

PhysicianID	PatientID	Date	Diagnosis	Location
2	1	3/3/02	Flu	Yonge
3	2	3/3/02	Measles	Yonge
2	1	3/3/02	Chicken Pox	Yonge

2. Which of the following queries on the schema below return the set of SIDs of all Sailors who have never reserved a red boat.
   Assume that dates are represented as strings and that <> means "not equal to" or $\neq$. FK means Foreign Key.
   4pt each. Circle the letter of each correct answer.

   `Sailor[` **SID**, `Sname ]`
   `Reserve[` **SID, BID, Date** `]` SID FK Sailor, BID FK Boat
   `Boat[` **BID**, `Color ]`

   (a)
   ```
   select  S.SID
   from    Sailors S
   where S.SID not in (select  R.SID
                       from    Reserve R, Boat B
                       where B.Color = 'red')
   ```

   (b)
   ```
   select  distinct S.SID
   from    Sailors S, Reserve R, Boat B
   where S.SID = R.SID and R.BID = B.BID and B.Color <> 'red'
   ```

   (c)
   ```
   select  S.SID
   from    Sailors S
   where not exists ( select  R.SID
                      from    Reserve R, Boat B
                      where S.SID = R.SID and R.BID = B.BID and B.Color = 'red' )
   ```

   (d)
   ```
   select      distinct R.SID
   from        Reserves R, Boat B
   where       R. BID = B.BID and B.color = 'red'
   group by    R.SID
   having      count(*) < 1
   ```

   (e) $\Pi_{SID}(Sailors) - \Pi_{SID}(\sigma_{Color='red'}(Reserve \bowtie Boat))$

3. Consider the following relational schema.
   `Emp [` **Eid**`, Name, Salary, City ]`

   `Dept [` **Did**`, Mgr, Budget, City ]` , Dept[Mgr] is foreign key of Emp.

   `WorksIn [` **Did, Eid** `]` , WorksIn[Eid] is foreign key of Emp, WorksIn[Did] is foreign key of Dept

   (a) **SQL 15pt** Write an SQL query to find all departments in Toronto that have a budget that is less than the sum of all its employees salaries. Return the set of Dids for these departments.

   (b) **Relational Algebra 15pt** Write a relational algebra query to find the (set of) employees who are managers of a department with at least two employees. The query should return the set of managers eids.

   (c) **Relational Algebra 15pt** Write a relational algebra query to find the (set of) names of all employees who work in exactly one department.

4. **15 pt** Give the result (data) returned by the following queries on the specific relations given below. (Give the **tuples** returned by each query when executed on the given relations. Do **not** describe the result in English.)

R

A	B
1	1
2	1
3	3

S

C	D
1	2
3	4
3	5

(a) $\Pi_{AD}(R \times S) - \rho_{A \leftarrow B}(\Pi_{BD}(R \bowtie_{B=C} S))$

(b) **select** S.C, S.D, sum(R.A)
from R, S
where R.B = S.C
group by S.C, S.D

(c) (**select** B
from R)
union all
(**select distinct** C
from S)