

University of Helsinki, Department of Computer Science
Basics of Databases, 17.4.2000, H.Laine

Write your name, date of birth (if you have not registered your social security number), the name of the course, date of the exam and your signature on each answer sheet

1. Let's consider the relations A ($n > 0$ tuples) and B ($m > 0$ tuples). a, b, k and v denote columns. Notations: π projection, \times cross product, $|\times|$ join, σ selection. Which of the following claims are true.
 - a) The cardinality (number of tuples) of relation $\pi_a(A)$ may be less than the cardinality of relation $\pi_{a,b}(A)$.
FALSE, because each value in column a appears also in $\pi_{a,b}(A)$
 - b) Relation $A \cup B$ has at least as many tuples as relation $A \times B$.
FALSE, because $A \cup B$ has $n+m$ rows and $A \times B$ has $n*m$ rows. This is true in n and m are equal to 1 and may be true in small n and m .
 - c) Relation $A |\times|_{a=b} B$ always has fewer tuples than relation $A \times B$.
FALSE. They may be of equal size.
 - d) Let k be the key of relation A and v a foreign key that refers to A. The cardinality of $A |\times|_{k=v} B$ is equal to the cardinality of B.
TRUE. if there are no null values in v.
 - e) Let k be the key of relation A and v a foreign key that refers to A. The cardinality of $A |\times|_{k \neq v} B$ is zero (0).
FALSE. if there are more than one non-null value
 - f) Let k be the key of A. If relation $\sigma_{k='s'}(A)$ is not empty and $\pi_a(A)$ has one tuple then relation A has only one tuple.
FALSE, The size of the table may not be deduced based on the given data

Consider the following stock exchange tables

```
person(pId, name, address)
company(cId, name, address, shares)
share_holder(owner→person, ltd → company, shares, date_updated)
insider(ltd→company, who→person, begin_date, end_date)
bidToSell(bidder→person, ltd→company, dateOfBid, timeOfBid, shares, price)
bidToBuy(bidder→person, ltd→company, dateOfBid, timeOfBid, shares, price)
deal(dealId, buyer→person, seller→person, subject→company, shares, price,
      dateOfDeal, timeOfDeal)
```

All prices are unit prices. Function *sysdate* gives the current date. Share_holder contains the state of ownership at the time specified in date_updated.

2. Express the following queries in SQL. Specify the proper order for the resulting rows.

- a) Bids made today to buy shares of Nokia.

```
select person.name, dateofbid, timeofbid, shares, price
from person, company, bidtobuy
where company.cid=bidtobuy.ltd and
      person.pid=pidtobuy.bidder and
      dateofbid=sysdate and
      company.name='Nokia'
order by timeofbid;
```

- b) Are there any companies an insider of which has made single deals of more than 1000 shares today? Who is this insider and what kind of deals has he/she made?.

```

select person.name, company.name, dateofdeal, timeofdeal, shares, price
from person, company, deal, insider
where company.cid=deal.subject and
      (person.pid=deal.buyer or person.pid=deal.seller) and
      deal.shares>=1000 and
      person.pid=insider.who and
      company.cid= insider.ltd and
      sysdate between insider.begin_date and insider.end_date and
      dateofdeal=sysdate
order by person.name, company.name;

```

- c) Companies, the shares of which nobody has bidd to buy today. (12p)

```

select name
from company
where cid not in
      (select ltd from bidtobuy)
order by name;

```

3. Express the following queries in SQL. Specify the proper order for the resulting rows.

- a) the number of insiders in Nokia company.

```

select count(*)
from insider, company
where name='Nokia' and
      company.cid=insider.ltd and
      sysdate between begin_date and end_date;

```

- b) Companies, whose shares are dealt for more than 10 million marks today.

```

select company.name, sum(shares* price)
from company, deal
where company.cid=deal.subject and
      dateofdeal=sysdate
group by company.name
having sum(shares*price)>1000000;

```

- c) Companies, whose average stock rate (average over deals made) has gone down today (you may assume that stocks were exchanged yesterday). (12p)

```

select company.name, todaysum, yesterdaysum
from company,
      (select sum((t.shares* t.price)/sum(t.shares) todaysum
            from deal t
            where dateofdeal=sysdate
            group by subject) todays,
      (select sum((e.shares* e.price)/sum(e.shares) yesterdaysum
            from deal e
            where dateofdeal=sysdate-1
            group by subject) y
where company.cid=todays.subject and
      company.cid=y.subject and
      todays.todaysum<y.ysterdaysum
order by company.name;

```

4. The state of possessing shares is updated in the process of declaring the deals. Outline how the declaration software should process the database. Explain the procedure and what database

operations are needed.. You need not give the operations exactly in SQL, but explain what they should do. You need not explain any technical detail of, for example, JDBC or things like that. (12p)

Scan the deals made during the day in the sequence they were made (date of deal)

```
(select)
  Find out the position of the seller (select)
  If the seller does not posses enough shares
    append the deal into the declaration list (insert)
  else
    decrease the position of the seller by the number of shares sold (update)
    increase the position of the buyer by the number of shares sold (update)
end loop;
```

5. Explain briefly what views are and what they are used for. (12p)