

Introduction to Databases

Homework no. 1: Relational algebra

1. The following relations are given (primary keys are underlined):

AUTHOR(AuthorCode, Name, Surname, Department, University)
ARTICLE(ArticleCode, Title, Topic)
AUTHORS_OF_ARTICLE(ArticleCode, AuthorCode)
EDITIONS_OF_CONFERENCE(Conference, Edition, EditionName, StartDate, EndDate, Editor)
AUTHOR_PRESENTS_ARTICLE(AuthorCode, Date, StartTime, EndTime, Room,
ArticleCode, Conference, Edition)

Write the following query in relational algebra

- (a) Show the code and the surname of the authors who have presented at least two articles with topic 'Databases' in the same edition of a conference.

2. The following relations are given (primary keys are underlined):

STUDENT(StudentID, Name, Surname, DegreeProgramme)
ASSIGNMENT_TO_BE_DELIVERED(ACode, Title, Topic, ScheduledExpirationDate)
TEACHER(TeacherID, Name, Surname, Department)
EVALUATION_OF_DELIVERED_ASSIGNMENT(StudentID, ACode, TeacherID,
DeliveryDate, EvaluationDate, Score)

Write the following query in relational algebra

- (a) Show the identifier and surname of the students of the "Computer Science Engineering" degree programme who have *always* delivered their assignments at least 15 days before the scheduled expiration date.
(b) Show the identifier and surname of the students who have delivered *all* assignments.

3. The following relations are given (primary keys are underlined):

TENNIS-PLAYER(TPId, TPName)
SPORTS-COMPLEX(SCCode, SCName, Municipality, Province)
TENNIS-COURT(TCCode, SCCode, Type)
BOOKING(TCCode, SCCode, Date, StartTime, EndTime, TPId)

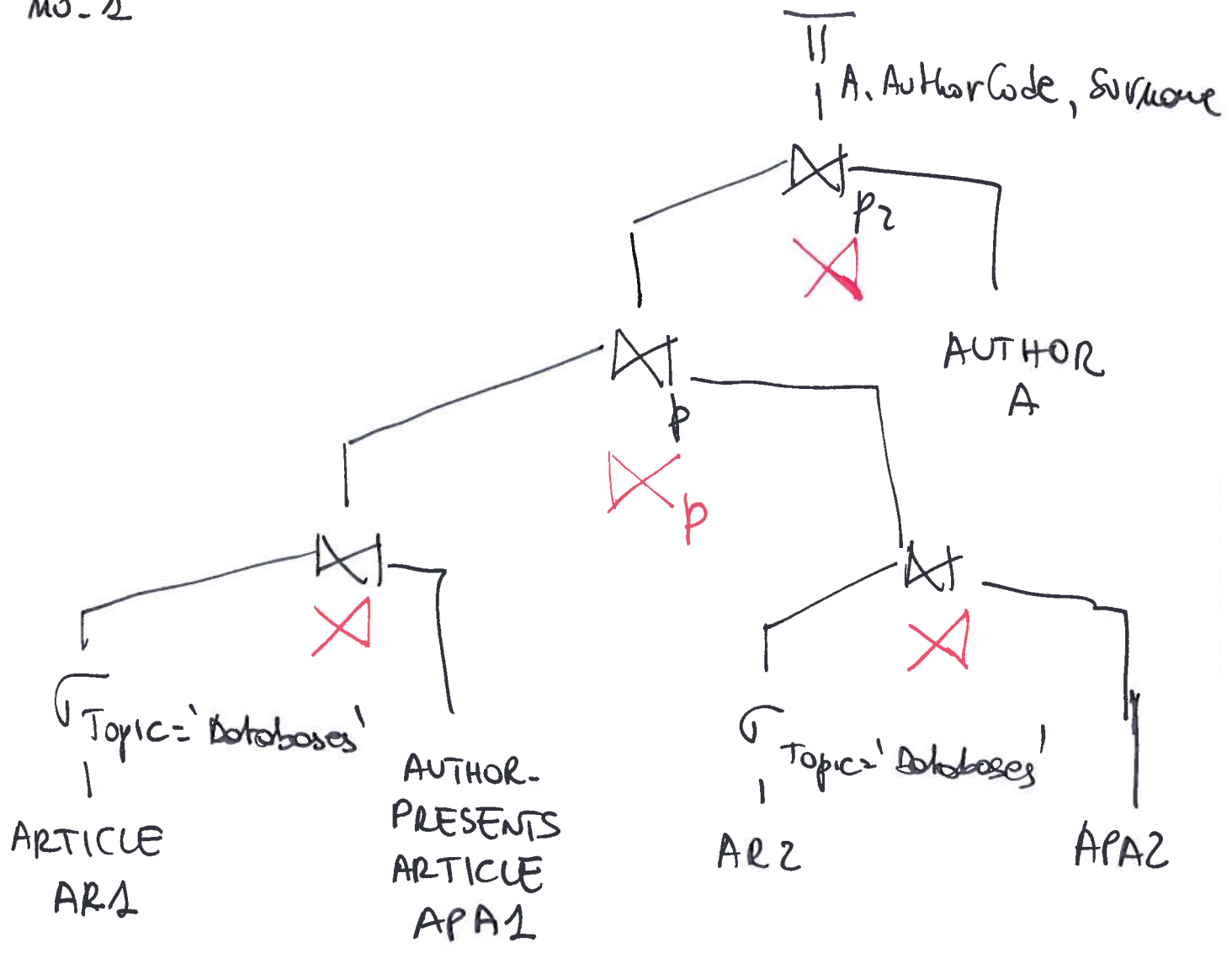
Write the following queries in relational algebra

- (a) Find identifier and name of tennis players who have booked at least two different tennis courts on 15-05-'12 at the same start time.

HOMEWORK No 1

Ex. no. 2

e)



p_1 : $APA1.AuthorCode = APA2.AuthorCode$ ← Same Author
AND

$AR1.ArticleCode \neq AR2.ArticleCode$ ← Different Article
AND

$APA2.Edition = APA1.Edition$
AND

$APA2.Conference = APA1.Conference$ } Same Edition

p_2 : $A.AuthorCode = APA1.AuthorCode$

SQL

```
SELECT AuthorCode, Surname  
FROM AUTHOR  
WHERE AuthorCode
```

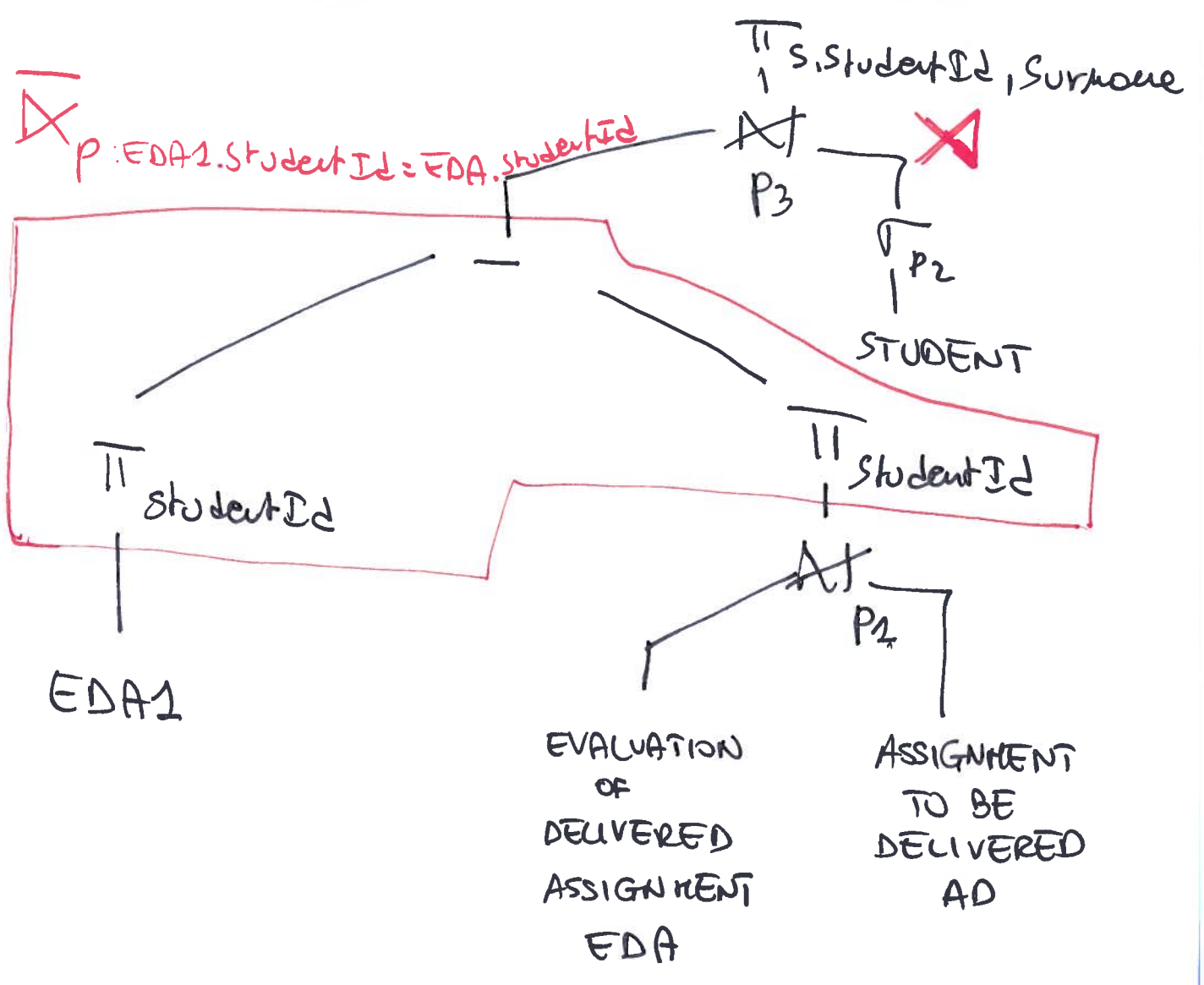
```
IN ( SELECT AuthorCode  
FROM ARTICLE A1,  
AUTHOR-PRESENTS-ARTICLE  
APA  
WHERE APA.ArticleCode =  
A1.ArticleCode AND  
Topic = 'Databases'  
GROUP BY AuthorCode, Conference,  
Edition  
HAVING COUNT(DISTINCT  
ArticleCode) >= 2 )
```

Ex. no. 2

e)

$$\text{Delivery Date} \leq \text{Scheduled Expiration Date} - 15$$

$$\text{Scheduled Expiration Date} - \text{Delivery Date} < 15$$



P_1 : $AD.ACCode = EDA.ACCode$ AND
 $Scheduled\ Expiration\ Date - Delivery\ Date < 15$

P_2 $Degree\ Programme = 'Computer\ Science\ Engineering'$

P_3 : $S.StudentId = EDA1.StudentId$

SQL

2)

SELECT

FROM STUDENT S, EVALUATION.OF.DELIVERED.ASSIGNMENT
EDA

WHERE S.StudentID = EDA.StudentID AND
Degree Programme = 'Computer Science Engineering'

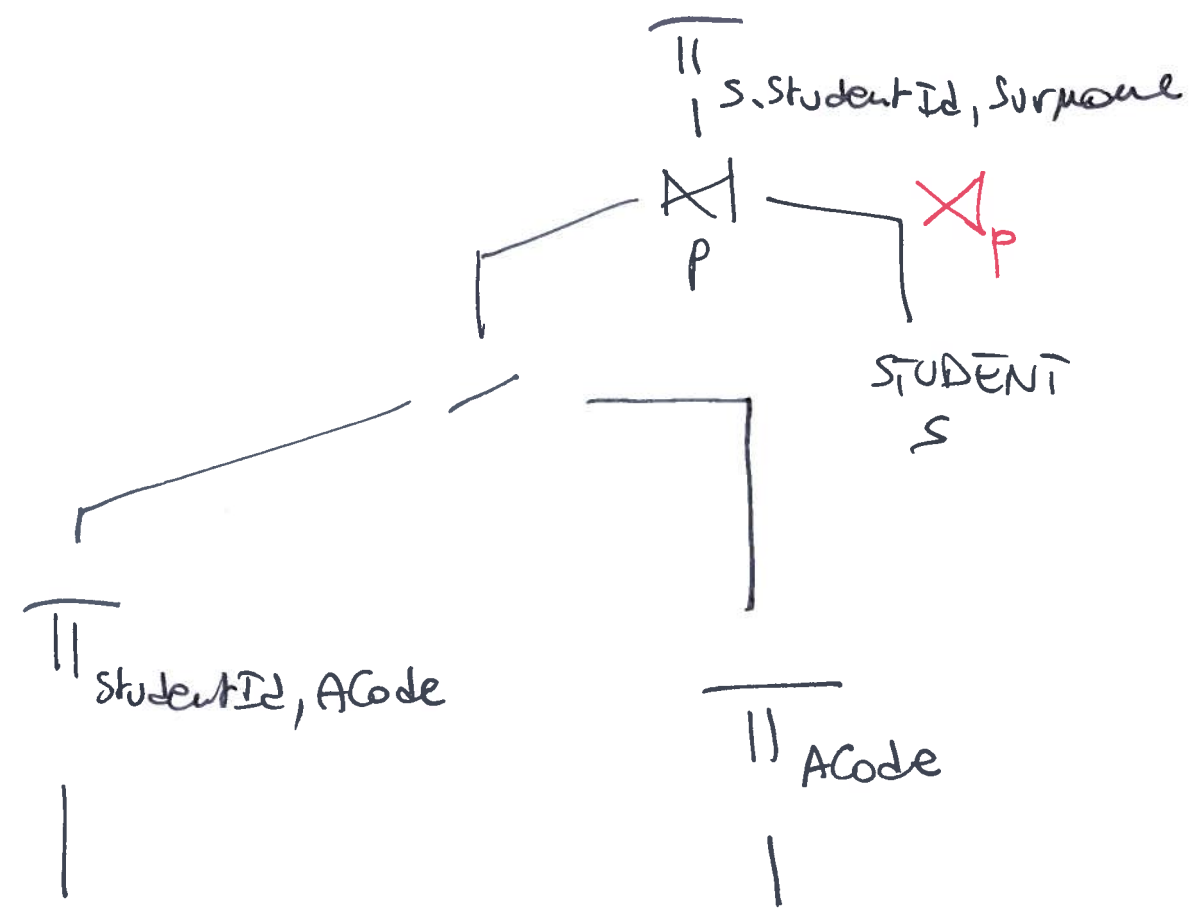
AND S.StudentID NOT IN

```
(SELECT StudentID  
FROM EDA1, ASSIGNMENT.TO_BE_DELIVERED  
AD  
WHERE EDA1.ACode = AD.ACode  
AND  
ScheduledExpirationDate -  
Delivery Date < 15)
```

IN

```
{ SELECT StudentID  
FROM EDA1, AD  
WHERE EDA1.ACode = AD.ACode  
GROUP BY StudentID  
HAVING MIN (ScheduledExpirationDate -  
Delivery Date) >= 15 }
```

2)



EVALUATION_OF
DELIVERED
ASSIGNMENT (EDA)

ASSIGNMENT_TO_BE
DELIVERED
AD

$P: EDA.StudentId = S.StudentId$

b) SQL

```
SELECT StudentId, Surname
```

```
FROM EVALUATION_OF_DELIVERED_ASSIGNMENT EDA,  
     STUDENT S
```

```
WHERE S.StudentId = EDA.StudentId
```

```
GROUP StudentId, Surname
```

```
HAVING COUNT(*) = (SELECT COUNT(*)
```

```
FROM ASSIGNMENT_TO_BE_DELIVERED)
```

Ex. No. 3

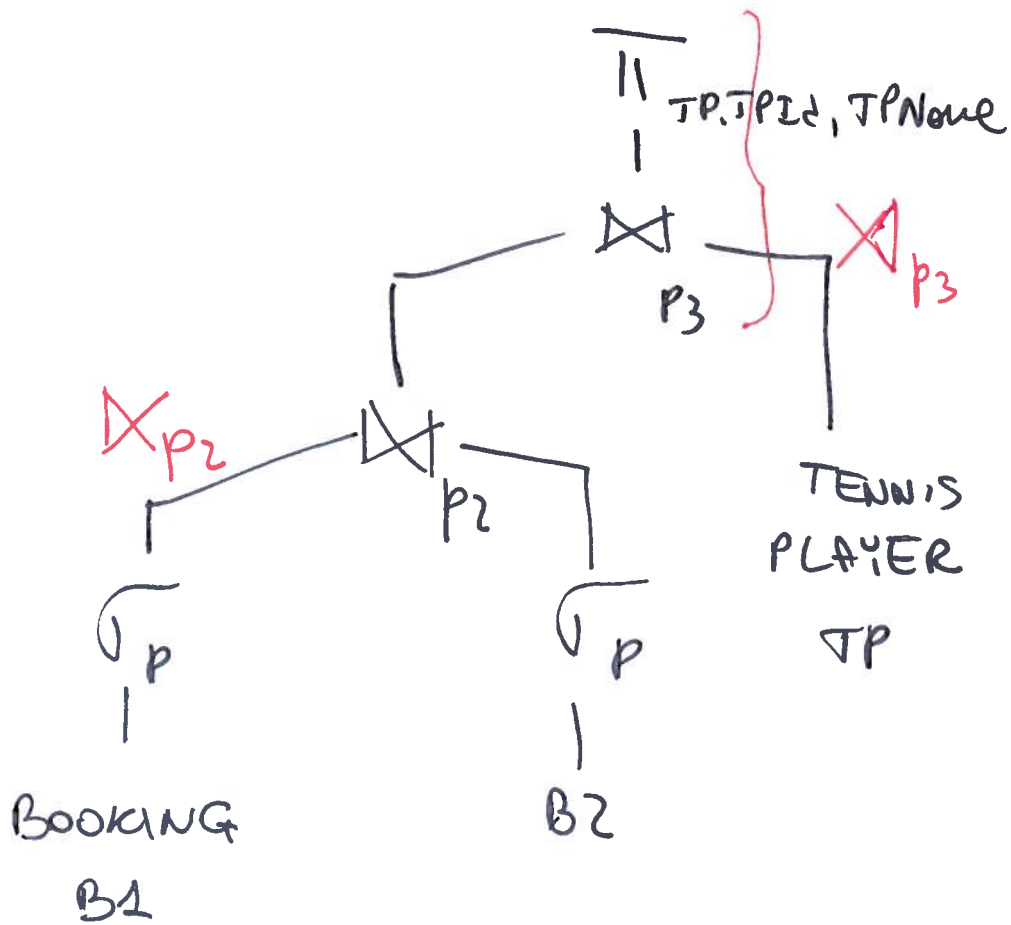
BOOKING

<u>TCCode</u>	<u>SCCode</u>	<u>Date</u>	<u>StartTime</u>	<u>EndTime</u>	<u>TPID</u>
1	A	15-05-2012	15		TP1
2	A	15-05-'12	15		TP1
<hr/>					
1	A	15-05-2012	18		TP2
1	B	15-05-'12	18		TP2

Different tennis courts means

\neq TCCode OR \neq SCCode

(e)



p: Date = 15-5-2012

p2 : B1.TPID = B2.TPID
AND

B1.startTime = B2.startTime
AND

(B1.TCCode <> B2.TCCode
OR
B1.SSCode <> B2.SSCode)

Different
Tennis
Courts

p3 : B1.TPID = TP.TPID

same tennis player

same start
time

SQL (e)

SELECT TP.TPID, TPNome

FROM BOOKING B, TENNIS-PLAYER TP

WHERE Date = 15-05-2012 AND

TP.TPID = B.TPID

GROUP BY TP.TPID, starttime, TPNome

HAVING COUNT(*) >= 2