Assignment 6
Computer Science
Fall 2017
B461

Steven Myers

October 31, 2017

Answers

1. (a)

$$\pi_{sid,bookno}(\sigma_{Buys.bookno \neq Buys2.bookno \wedge Student.name='Eric' \wedge Buys.bookno \neq '2010'}(Student \times Buys \times Buys2))$$

(b) i. Original Query

```
SELECT DISTINCT b.bookno, b.title
FROM book b, student s
WHERE b.price = SOME(select b1.price
                     from buys t, book b1
                     where b1.price > 50 and
                     s.sid = t.sid and
                     t.bookno = b1.bookno);
```

ii. Convert from using SOME to an exists statement

```
SELECT DISTINCT b.bookno, b.title
FROM book b, student s
WHERE EXISTS(select 1
             from buys t, book b1
             where b1.price > 50 and
             s.sid = t.sid and
             t.bookno = b1.bookno);
```

iii. Push down student s relation into subquery as $s1$.

```
SELECT DISTINCT b.bookno, b.title
FROM book b, student s
WHERE EXISTS(select 1
             from buys t, book b1, student s1
             where b1.price > 50 and
             s1.sid = t.sid and
             t.bookno = b1.bookno);
```

iv. The inner query can be translated as:

$$\pi_{b1.price}(\sigma_{b1.price>50 \wedge s1.sid=t.sid \wedge t.bookno=b1.bookno}(T \times B1 \times S1))$$

v. Finally, we perform the semi-join with the inner query and the student and book relation:

$$\pi_{b.bookno,b.title}(S \times B) \ltimes \pi_{b1.price}(\sigma_{b1.price>50 \wedge s1.sid=t.sid \wedge t.bookno=b1.bookno}(T \times B1 \times S1))$$

(c) i. Original Query

```
SELECT b.bookno
FROM book b
WHERE b.bookno IN
(SELECT b1.bookno FROM book b1 WHERE b1.price > 50)
UNION
(SELECT c.bookno FROM cites c);
```

ii. RA representation of inner query (simple translation)

$$\pi_{bookno}(\sigma_{price>50}(book)) \cup \pi_{bookno}(cites)$$

iii. Since the IN predicate is equivalent to saying that there exists one bookno for which a bookno in the inner query matches, we can use the semi join

$$\pi_{bookno}(book) \ltimes \pi_{bookno}(\sigma_{price>50}(book)) \cup \pi_{bookno}(cites)$$

(d) i. Original Query

```
SELECT b.bookno FROM book b
WHERE b.price >= 80 and
NOT EXISTS(SELECT b1.bookno
           FROM book b1
           WHERE b1.Price > b.Price);
```

ii. First, push down book b relation into subquery as book b2.

```
SELECT b.bookno FROM book b
WHERE b.price >= 80 and
NOT EXISTS(SELECT b1.bookno
           FROM book b1, book b2
           WHERE b1.Price > b2.Price);
```

iii. Now, we can properly translate the inner query as:

$$\pi_{b1.bookno}(\sigma_{b1.price>b2.price}(B1 \times B2))$$

iv. To preserve the semantics of the original outer query, we need to perform an anti-semi join:

$$\pi_{b.bookno}(\sigma_{b.price>=80}(B \overline{\ltimes} \pi_{b1.bookno}(\sigma_{b1.price>b2.price}(B1 \times B2))))$$

(e) i. Original query:

```
SELECT s.sid
FROM Student s
WHERE EXISTS(SELECT 1
          FROM Book b
          WHERE b.price > 50 AND
          b.bookno IN (SELECT t.bookno
                    FROM Buys t
                    WHERE s.sid = t.sid AND
                    s.sname = 'Eric'))
```

ii. First, push down parameterized values in the inner-most query. We will push down the student s relation as $s1$.

```
SELECT s.sid
FROM Student s
WHERE EXISTS(SELECT 1
          FROM Book b
          WHERE b.price > 50 AND
          b.bookno IN (SELECT t.bookno
                    FROM Buys t, Student s1
                    WHERE s1.sid = t.sid AND
                    s1.sname = 'Eric'))
```

iii. Next, convert the first inner subquery IN expression into an EXISTS statement. We can do this by pushing the book relation completely into the inner most subquery and adding its conditions

```
SELECT s.sid
FROM Student s
WHERE EXISTS(SELECT t.bookno
                    FROM Buys t, Student s1, Book b
                    WHERE s1.sid = t.sid AND
                    b.price > 50 AND
                    b.bookno = t.bookno
                    s1.sname = 'Eric')
```

iv. We can translate the inner query as:

$$\pi_{t.bookno}(\sigma_{s1.sid=t.sid \ \wedge \ b.price>50 \ \wedge \ b.bookno=t.bookno \ \wedge \ s1.sname='Eric'}(S1 \times B \times T))$$

v. Finally, we can translate our outer exists expression with a semijoin and project the student sid.

$$\pi_{s.sid}(S \ltimes \pi_{t.bookno}(\sigma_{s1.sid=t.sid \ \wedge \ b.price>50 \ \wedge \ b.bookno=t.bookno \ \wedge \ s1.sname='Eric'}(S1 \times B \times T)))$$

(f) i. Original query:

```
SELECT s1.sid, s2.sid
FROM student s1, student s2
WHERE s1.sid <> s2.sid AND
NOT EXISTS(SELECT 1
           FROM Buys t1
           WHERE t1.sid = s1.sid AND
           t1.bookno NOT IN (SELECT t2.bookno
                             FROM Buys t2
                             WHERE t2.sid = s2.sid));
```

ii. For the deepest query, we need to translate the NOT IN statement to NOT EXISTS.

```
SELECT s1.sid, s2.sid
FROM student s1, student s2
WHERE s1.sid <> s2.sid AND
NOT EXISTS(SELECT 1
           FROM Buys t1
           WHERE t1.sid = s1.sid AND
           NOT EXISTS (SELECT 1
                       FROM Buys t2
                       WHERE t2.sid = s2.sid AND t1.bookno <> t2.bookno));
```

iii. We now need to push down our relations recursively from the top and middle queries to the deepest query.

```
SELECT s1.sid, s2.sid
FROM student s1, student s2
WHERE s1.sid <> s2.sid AND
NOT EXISTS(SELECT 1
           FROM Buys t, Student s1, Student s2
           WHERE t.sid = s1.sid AND
           NOT EXISTS (SELECT t2.sid, t1.bookno, t2.bookno, s1.sid
                       FROM Buys t2, Buys t1, Student S1, Student S2
                       WHERE t2.sid = s2.sid AND t1.bookno <> t2.bookno));
```

iv. Now that we've pushed down the upper query's relations into deeper queries, we can translate the deepest query as:

$$\varepsilon = \pi_{t2.sid,t1.bookno,t2.bookno,s1.sid}(\sigma_{t2.sid=s2.sid \land t1.bookno \neq t2.bookno}(T1 \times T2 \times S1 \times S2))$$

v. Our middle query then can be translated as:

$$\tau = \pi_{s1.sid,s2.sid,t.bookno}(\sigma_{t.sid=s1.sid}(T \times S1 \times S2) \overline{\ltimes} \varepsilon)$$

vi. Finally, we can semijoin the top level query in an expression like so:

$$\pi_{s1.sid,s2.sid}(\sigma_{s1.sid \neq s2.sid}(S1 \times S2) \overline{\ltimes} \tau)$$

2.