CS 455: Principles of Database Systems

Review Guide 1: Relational Algebra (Soln)

Problems

1. Suppose we're given two non-empty relations R and S, where $|R| = n$ and $|S| = m$ tuples, respectively. Assume c could be any boolean condition, and L is non-empty list of attributes of R. You can further assume that R and S are compatible, so that the set operators involved are applicable. Where it applies, note that $\emptyset \times A = \emptyset$.

   For each relational algebra expression, give the maximum number of tuples that results as a formula in terms of m and n. I've gotten one started for you.

	Maximum Size of Result
$\sigma_c(S)$	m
$R \cup S$	$m + n$
$R \cap S$	$n (= m)$
$\pi_L(R) \setminus S$	n
$\pi_L(R) \cap \sigma_c(R)$	$min(m, n)$
$R \bowtie S$	mn
$\sigma_c(R) \times S$	mn

books

BNO	Title	Author	Date	Edition
231	The Soul of a New Machine	Tracy Kidder	1981	1
77	Programming Pearls	Jon Bentley	2000	2
23	Programming Pearls	Jon Bentley	1981	1
2	Tess of the d'Urbervilles	Thomas Hardy	1850	1

publishers

PNO	Publisher	City	Web Site
1	Back Bay Books	Boston	backbay.com
2	Addison Wesley	New York	addisonwesley.com
3	Modern Library	London	randomhouse.com
4	Penguin	New York	penguin.com

publishes

PNO	BNO	Pages	Copyright
1	231	293	1981
2	77	235	2001
2	23	200	1980
3	2	565	2001
4	2	540	1990

Assumptions:

- *BNO* are unique book identifiers.
- *PNO* are unique publisher identifiers.
- The date in the *books* relation represents the year the book was originally issued. The date in the *publishes* table represents the copyright date when that particular publisher issued it.

2. Consider the relational database given on the previous page.

   (a) Why would it be a good idea to not permit the following query?

   $$publishes \leftarrow publishes \cup \{(4, 21, 212, 1995)\}$$

   (b) List *all* superkeys for the **publishes** relation, and circle the candidate key(s).

   (c) ** The schema of the **publishers** relation is not a great design. What problem(s) do you see? What would you do to fix it?

   The issue is that this relation suffers from redundancies in real life. Think about how you'd store multiple websites and multiple locations.

   (d) Suppose we wish to include a new entity, *Editors*, into our books database. Each editor has a unique editor number, name, and the year they joined the group. Each publisher can hire multiple editors, but each editor can only work for one publisher. Each editor can edit many books. Design the necessary relation(s) that minimizes redundancy.

3. Give the **relational algebra** expressions for each of the following queries. Your queries should be general and work for any instance of the relations.

   (a) Retrieve all book titles and their authors that copyrighted on or before 1990.

   $\pi_{Title,Author}(\sigma_{Copyright \leq 1990}(publishes) \bowtie books)$

   (b) For each publisher, find the average number of pages in the books it has published.

   $({}_{PNO}\,\mathcal{G}_{average(Pages)}(publishes)) \bowtie publishers$

   (c) Find the total number of pages published by Back Bay Books.

   $\mathcal{G}_{sum(Pages)}(\sigma_{Publisher='BackBayBooks'}(publishers) \bowtie publishes)$

   (d) ** Find the books with the most and least number of pages. (Caveat: Books may share the same number of pages.)

   $maxmin \leftarrow \mathcal{G}_{max(Pages)}(publishes) \cup \mathcal{G}_{min(Pages)}(publishes)$
   $\sigma_{Pages \in maxmin}(publishes) \bowtie books$

   (e) ** Suppose I renamed the "Copyright" attribute in the **publishes** relation to "Date." Find the book title and page number for anything written by Thomas Hardy.

   $\sigma_{Author='ThomasHardy'}(books) \bowtie_{books.BNO=publishes.BNO} publishes$

   This one was tricky. You cannot use a $\bowtie$ here because it will try to enforce equality on *both* BNO and Date. Instead, I used a θ join to enforce only one of the equalities.

   (f) ** Retrieve the city (or cities) with the highest number of publishers.

   $Tmp \leftarrow \rho_{(city,numpubs)}({}_{city}\mathcal{G}_{count(*)}(publishers))$
   $max \leftarrow \mathcal{G}_{max(numpubs)}(Tmp)$
   $\pi_{city}(\sigma_{numpubs \in max}(Tmp))$