The University of British Columbia
Computer Science 304

Midterm Examination
October 29, 2013

Time: 80 minutes
Instructor: Rachel Pottinger

Name_________________________________________Student No___________________
(PRINT) (Last) (First)

Signature________________________________________

This examination has 4 doublesided pages.

Check that you have a complete paper.

This is a closed book, closed notes exam. No books or other material may be used.

Answer all the questions on this paper.

Give very short but precise answers.

State any assumptions you make

Work fast and do the easy questions first. Leave some time to review your exam at the end.

Good Luck

<table>
<thead>
<tr>
<th>Question</th>
<th>Mark</th>
<th>Out of</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.a</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1.b</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1.c</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1.d</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2.a</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2.b</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>Out of 30</td>
</tr>
</tbody>
</table>
All queries for this exam use the same schema as in some of the SQL tutorials:

- authors( au_id, au_lname, au_fname, phone, address, city, state, zip)
- titleauthors( au_id, title_id, au_ord, royaltyshare)
- sales( sonum, stor_id, ponum, sdate)
- salesdetails( sonum, qty_ordered, qty_shipped, title_id, date_shipped)
- editors( ed_id, ed_lname, ed_fname, ed_pos, phone, address, city, state, zip, ed_boss)
- titleditors( ed_id, title_id, ed_ord)
- titles( title_id, title, type, pub_id, price, advance, ytd_sales, contract, notes, pubdate)
- publishers( pub_id, pub_name, address, city, state)

The schema will be repeated on following pages for easy reference.

Foreign Keys are shown in the following diagram, where the referring attribute is marked by a + and the referencing attribute is marked by a  (e.g., au_id in titleauthors references au_id in authors)
The schema again:
authors(au_id, au_lname, au_fname, phone, address, city, state, zip)
titleauthors(au_id, title_id, au_ord, royaltyshare)
sales(sonum, stor_id, ponum, sdate)
salesdetails(sonum, qty_ordered, qty_shipped, title_id, date_shipped)
editors(ed_id, ed_lname, ed_fname, ed_pos, phone, address, city, state, zip)
titleditors(ed_id, title_id, ed_ord)
titles(title_id, title, type, pub_id, price, advance, ytd_sales, contract, notes, pubdate)
publishers(pub_id, pub_name, address, city, state)

1. a. SQL: Find the first name of all authors who are not editors. Remove duplicates and alphabetize.
The schema again:

```sql
authors( au_id, au_lname, au_fname, phone, address, city, state, zip)
titleauthors( au_id, title_id, au_ord, royaltyshare)
sales( sonum, stor_id, ponum, sdate)
salesdetails( sonum, qty_ordered, qty_shipped, title_id, date_shipped)
editors( ed_id, ed_lname, ed_fname, ed_pos, phone, address, city, state, zip)
titleditors( ed_id, title_id, ed_ord)
titles( title_id, title, type, pub_id, price, advance, ytd_sales, contract, notes, pubdate)
publishers( pub_id, pub_name, address, city, state)
```

b. **SQL:** “List the last names of all authors who have a letter 'k' in their last name?” If a last name occurs more than once, only list it once
The schema again:

authors(`au_id, au_lname, au_fname, phone, address, city, state, zip`)
titleauthors(`au_id, title_id, au_ord, royaltyshare`)
sales(`sonum, stor_id, ponum, sdate`)
salesdetails(`sonum, qty_ordered, qty_shipped, title_id, date_shipped`)
editors(`ed_id, ed_lname, ed_fname, ed_pos, phone, address, city, state, zip`)
titleditors(`ed_id, title_id, ed_ord`)
titles(`title_id, title, type, pub_id, price, advance, ytd_sales, contract, notes, pubdate`)
publishers(`pub_id, pub_name, address, city, state`)

**c: SQL.** For each editor who has edited more than two books, return the last name of the editor and how many books she/he has edited.
• The schema again:
  authors( au_id, au_lname, au_fname, phone, address, city, state, zip)
  titleauthors( au_id, title_id, au_ord, royaltyshare)
  sales( sonum, stor_id, ponum, sdate)
  salesdetails( sonum, qty_ordered, qty_shipped, title_id, date_shipped)
  editors ( ed_id, ed_lname, ed_fname, ed_pos, phone, address, city, state, zip)
  titleditors( ed_id, title_id, ed_ord)
  titles( title_id, title, type, pub_id, price, advance, ytd_sales, contract, notes, pubdate)
  publishers( pub_id, pub_name, address, city, state)

d. Find the last name of the first author(s) of the book(s) that has had the most number of copies ordered. Remove duplicates.
The schema again:

- authors( au_id, au_lname, au_fname, phone, address, city, state, zip)
- titleauthors( au_id, title_id, au_ord, royaltyshare)
- sales( sonum, stor_id, ponum, sdate)
- salesdetails( sonum, qty_ordered, qty_shipped, title_id, date_shipped)
- editors( ed_id, ed_lname, ed_fname, ed_pos, phone, address, city, state, zip)
- titleeditors( ed_id, title_id, ed_ord)
- titles( title_id, title, type, pub_id, price, advance, ytd_sales, contract, notes, pubdate)
- publishers( pub_id, pub_name, address, city, state)

2. THIS QUESTION IS FOR DATALOG

a. **DATALOG**: Find the first name of all people who have been either editors or authors

b. **DATALOG**: Find the titles of all books by an author with the last name ‘Smith’ where the author’s zip is greater than 15232