1 Overview

You are to use your accounts and DB2 to implement the BIBsearch system. BIBsearch consists of two or three application programs with simple command line interfaces. The requirements for the programs are given in the following sections. A specification of the underlying schema for the database is also given and includes a schema diagram that indicates primary and foreign key constraints. Note that all submissions must use this schema. Either C or C++ must be used together with SQL and the static embedded SQL protocol to implement BIBsearch.

2 Applications Programs

2.1 Program bibauthor

This application is to print a list of publication records for each publication by an author with a name supplied as a string argument on the command line. In the case of articles, the publication identified by the pubid for the publication in which the article appears should be added to the list of requested publications so that a record for this publication is also printed. Note that a record for any publication should appear at most once in the output. Also, the publication information records should be printed to standard output.

The publications should appear according to a major sort on the year of the publication (latest first) and a minor sort on the name of the first author for the publication. In the case of publications without an author, the name of the first author should be assumed to be the zero-length string for sorting purposes.

The output format for each publication record is as follows.

Pubid: pubid
Type: { book | journal | proceedings | article }
Title: title

In addition, for each particular type of publication, the following information should also be printed.

- For books:
  Authors: author1, ..., authork
  Publisher: publisher
  Year: year

- For journals:
  Volume: volume
  Number: number
  Year: year

- For proceedings:
  Year: year
• For articles:

    Authors: author$_1$, ..., author$_k$
    In: appearsin
    Pages: startpage--endpage

2.2 Program bibcontent (for extra 20% credit)

This application is to print a list of publication records for each article appearing in a proceedings or journal identified
by a pubid supplied as a string argument on the command line. The articles should appear according to a major sort
on their starting page numbers, and should be printed with an output format matching that of bibauthor.

2.3 Program bibmaint

This application inserts new publications into the database. The input is given to the application as a list of insertion
requests on the standard input in the following format (depending on the publication type):

    author(aid#name)
    authorurl(aid#url)
    book(pubid#title#aid$_1$;...;aid$_k$#publisher#year)
    journal(pubid#title#volume#number#year)
    proceedings(pubid#title#year)
    article(pubid#title#aid$_1$;...;aid$_k$#appearsin#startpage#endpage)

NOTE: You can assume that insertions requested are originating from a reliable source, and that there is therefore no
need to check for any errors in the input, e.g., such as checking if a pubid already exists.

Sample input data for an execution of bibmaint is as follows:

    proceedings(SIGMOD02#Proc. ACM SIGMOD Conference on the Management of Data#2002)
    author(1#Peter Bumbulis)
    authorurl(2#http://db.uwaterloo.ca/itbowman)
    article(2#A Compact B-Tree#1;2#SIGMOD02#533#541)

You can assume that all string constants in the input will not contain any of the characters “#”, “;”, “)” or “)”. (HINT: read “man strtok(3)” for a way to help with parsing input commands.)

When an insertion succeeds for publications, the application must print out the newly inserted data in the same
format used in the reporting applications. Again, note that you are free to assume error checking of input data is
unnecessary.
3 Database Schema

(attribute domains and class diagram)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>SQL Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>aid</td>
<td>integer not null</td>
</tr>
<tr>
<td>name</td>
<td>char(22) not null</td>
</tr>
<tr>
<td>url</td>
<td>char(42)</td>
</tr>
<tr>
<td>pubid</td>
<td>char(10) not null</td>
</tr>
<tr>
<td>aorder</td>
<td>integer not null</td>
</tr>
<tr>
<td>title</td>
<td>char(70) not null</td>
</tr>
<tr>
<td>publisher</td>
<td>char(50) not null</td>
</tr>
<tr>
<td>year</td>
<td>integer not null</td>
</tr>
<tr>
<td>volume</td>
<td>integer not null</td>
</tr>
<tr>
<td>number</td>
<td>integer not null</td>
</tr>
<tr>
<td>appearsin</td>
<td>char(10) not null</td>
</tr>
<tr>
<td>startpage</td>
<td>integer not null</td>
</tr>
<tr>
<td>endpage</td>
<td>integer not null</td>
</tr>
</tbody>
</table>

(additional constraints)

There are a number of additional constraints that are always satisfied by a given bibliography. (Your programs may assume these constraints are always true, including bibmaint.) Let $Pu$, $B$, $J$, $Pr$ and $A$ denote the set of values for attribute “pubid” in tables PUBLICATION, BOOK, JOURNAL, PROCEEDINGS and ARTICLE, respectively, and let $In$ denote the set of values for attribute “appearsin” in table ARTICLE. Then each of the following conditions is always true:

- $Pu \subseteq (B \cup J \cup Pr \cup A)$;
- $In \subseteq (J \cup Pr)$; and
- $|Pu| = (|B| + |J| + |Pr| + |A|)$,

where $|x|$ denotes a count of the number of elements in $x$. The last constraint ensures, for example, that $B$ and $J$ are disjoint. Finally, note that “pubid” values occurring in table WROTE will always refer to publications that are either books or articles.
4 Assignment Submission

A submission of the following items should be made by the assignment due date:

1. Source listings of each of the two or three application programs comprising BIBsearch. These should be submitted online in two or three files with names beginning with bibauthor, bibmaint and optionally bibcontent, with each ending with the appropriate suffix (i.e., *.sqc).

2. A shell script named compile. Typing “./compile” should suffice to compile all submitted application programs.

3. A short design document that outlines the design of your BIBsearch system consisting of just one page. The purpose of this document is to assist graders in understanding how your system is implemented.