Database Systems: Homework 3 Key

Due 25 October, 2013

Team: _____ Key _____

1. **(20 points)** Use the mapping algorithms to convert the EER database schema shown in Figure 8.9 to Relational form. Please use a design or drawing program, or draw neatly and legibly. Attach your solution separately.

   See attached drawing.

2. Consider the relation $R = \{A, B, C, D, E, F, G, H, I, J\}$ and the set of functional dependencies $F = \{\{A, B\} \rightarrow \{C\}, \{A\} \rightarrow \{D, E\}, \{B\} \rightarrow \{F\}, \{F\} \rightarrow \{G, H\}, \{D\} \rightarrow \{I, J\}\}$. In the prequiz you might have found that the key of this relation is $\{A, B\}$.

   (a) **(5 points)** Decompose R into 2NF.

   Splitting out attributes based on relations only partially dependent on the key gives:

R_1	=	$\{\underline{A}, D, E, I, J\}$	preserves the functional dependencies $\{\{A\} \rightarrow \{D, E\}, \{D\} \rightarrow \{I, J\}\}$
R_2	=	$\{\underline{B}, F, G, H\}$	preserves $\{\{B\} \rightarrow \{F\}, \{F\} \rightarrow \{G, H\}\}$
R_3	=	$\{\underline{A}, \underline{B}, C\}$	preserves $\{\{A, B\} \rightarrow \{C\}\}$

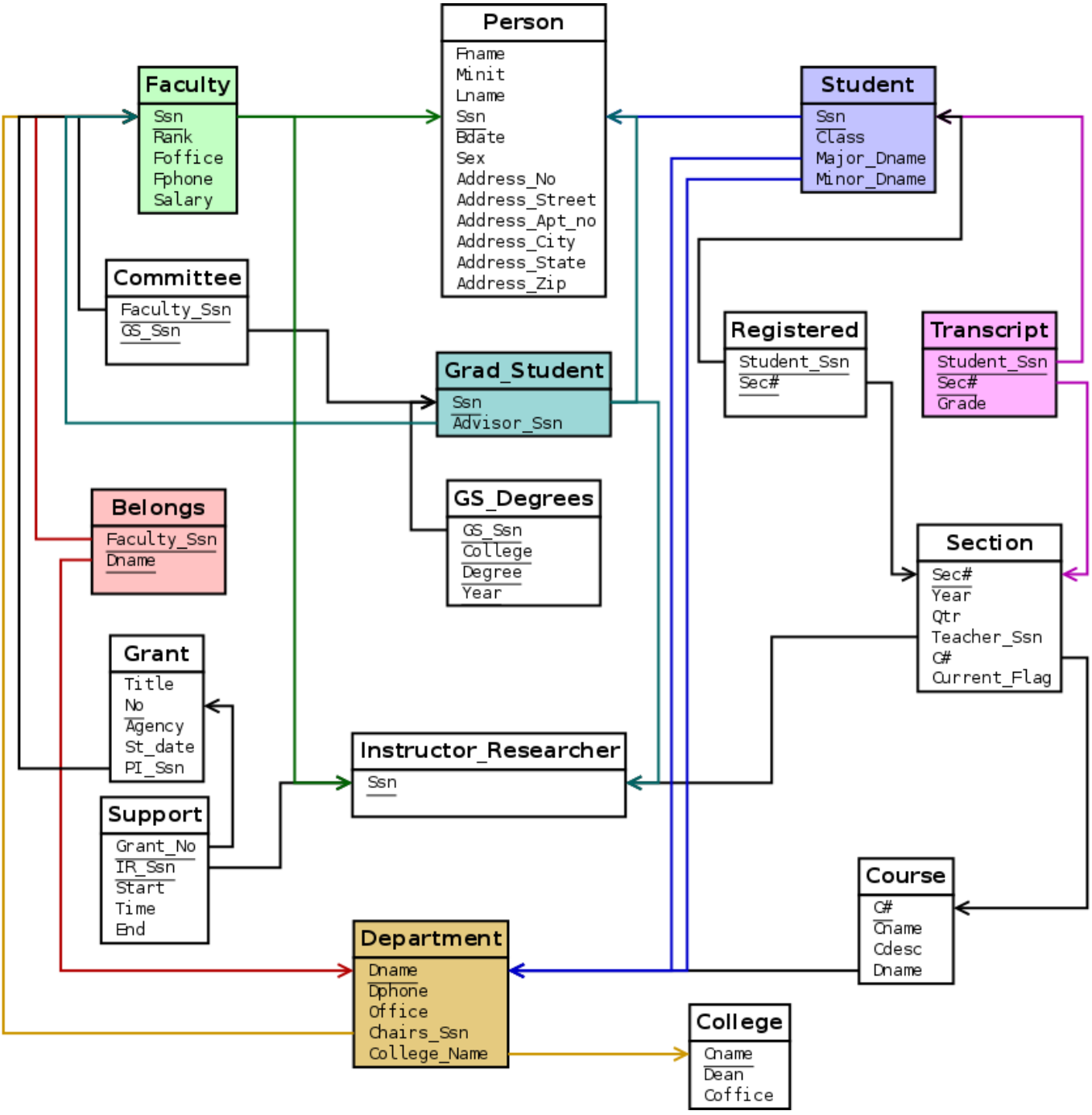
   The primary keys of these subrelations are underlined.

   (b) **(5 points)** Decompose that further into 3NF.

   Further splitting attributes with transitive dependencies on their keys gives:

   $$\begin{aligned} R_{1a} &= \{\underline{A}, D, E\} \\ R_{1b} &= \{\underline{D}, I, J\} \\ R_{2a} &= \{\underline{B}, F\} \\ R_{2b} &= \{\underline{F}, G, H\} \\ R_3 &= \{\underline{A}, \underline{B}, C\} \end{aligned}$$

Relational schema mapped from the EER schema in Figure 8.9



3. Consider the relation R, which has attributes that hold schedules of courses and sections at a university; $R = \{$Course_no, Sec_no, Offering_dept, Credit_hours, Course_level, Instructor_ssn, Semester, Year, Days_hours, Room_no, No_of_students$\}$. Suppose that the following functional dependencies hold on R:

$$\{\text{Course\_no}\} \rightarrow \{\text{Offering\_dept}, \text{Credit\_hours}, \text{Course\_level}\} \quad (1)$$

$$\begin{Bmatrix}\text{Course\_no}, & \text{Sec\_no},\\ \text{Semester}, & \text{Year}\end{Bmatrix} \rightarrow \begin{Bmatrix}\text{Days\_hours}, & \text{Room\_no},\\ \text{No\_of\_students}, & \text{Instructor\_ssn}\end{Bmatrix} \quad (2)$$

$$\begin{Bmatrix}\text{Room\_no}, & \text{Days\_hours},\\ \text{Semester}, & \text{Year}\end{Bmatrix} \rightarrow \{\text{Instructor\_ssn}, \text{Course\_no}, \text{Sec\_no}\} \quad (3)$$

(a) **(5 points)** Which sets of attributes form candidate keys of R?

The closure of $K_1 = \{$Course_no, Sec_no, Semester, Year$\}$ under the functional dependencies is the entire R. This is also the case for $K_2 = \{$Room_no, Days_hours, Semester, Year$\}$, and no other set that doesn't already contain K_1 or K_2 as subsets, so these are the two candidate keys.

(b) **(10 points)** Decompose this relation to 3NF.

Start by decomposing it to 2NF. The only FD that is partial on the left and nonkey on the right is $\{$Course_no$\} \rightarrow \{$Offering_dept, Credit_hours, Course_level$\}$, so decompose on that to:

R_1 = {Course_no, Offering_dept, Credit_hours, Course_level}

R_2 = {Course_no, Sec_no, Instructor_ssn, Semester, Year, Days_hours, Room_no, No_of_students}

Neither of these relations have a nonkey attribute transitively dependent on a key, so this is also in 3NF.

4. **(20 points)** Write a program for accessing a `LIBRARY` database with the schema shown in Figure 4.6, that allows a user to search for books having a particular title, author, or both. Attach the source code.

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

import MySQLdb

# There are many ways to do this. Here's one.

host = 'example.com'
user = 'hrothgar'
passwd = 'ih8grendL'
db = 'library'

try:
    conn = MySQLdb.connect(host, user, passwd, db)
    cursor = conn.cursor()

    title = raw_input('Title to search for (leave blank if "any"): ')
    author = raw_input('Author to search for (leave blank if "any"): ')
    both = len(title) > 0  and  len(author) > 0

    query = 'select Book_id, Title, Publisher_name from Book natural join Book_Authors
        where '
```

```
    if len(title) > 0:
        query += "Title = '%s'" % title
    if len(author) > 0:
        if both:
            query += ' and '
        query += "Author_name = '%s'" % author

    cursor.execute(query)
    rows = cursor.fetchall()
    print 'Found %d books matching that search:' % len(rows)
    for row in rows:
        print '[%5d] %s, published by %s' % row

except MySQLdb.Error, e:
    print "database error %d: %s" % (e.args[0], e.args[1])
```

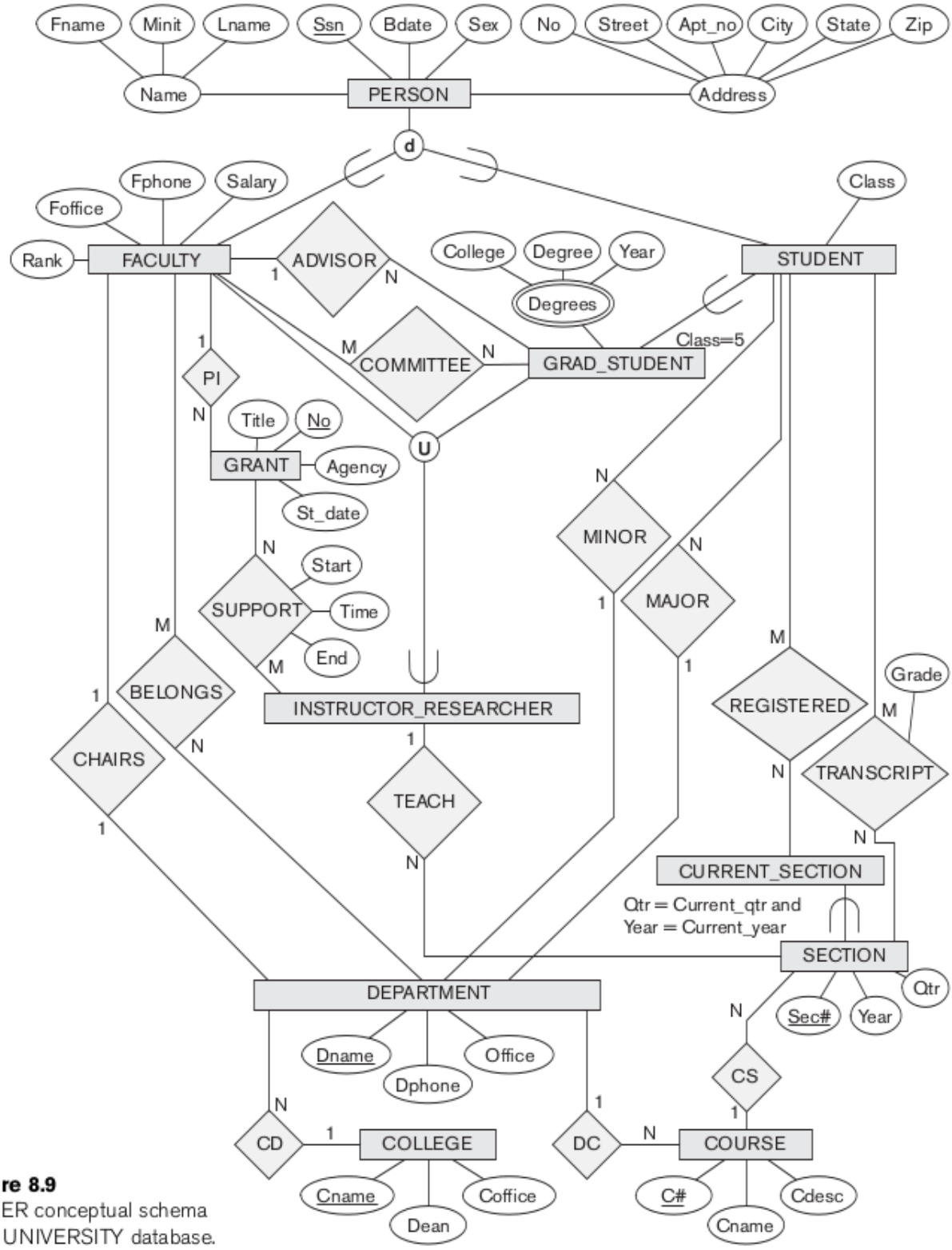


Figure 8.9
An EER conceptual schema for a UNIVERSITY database.

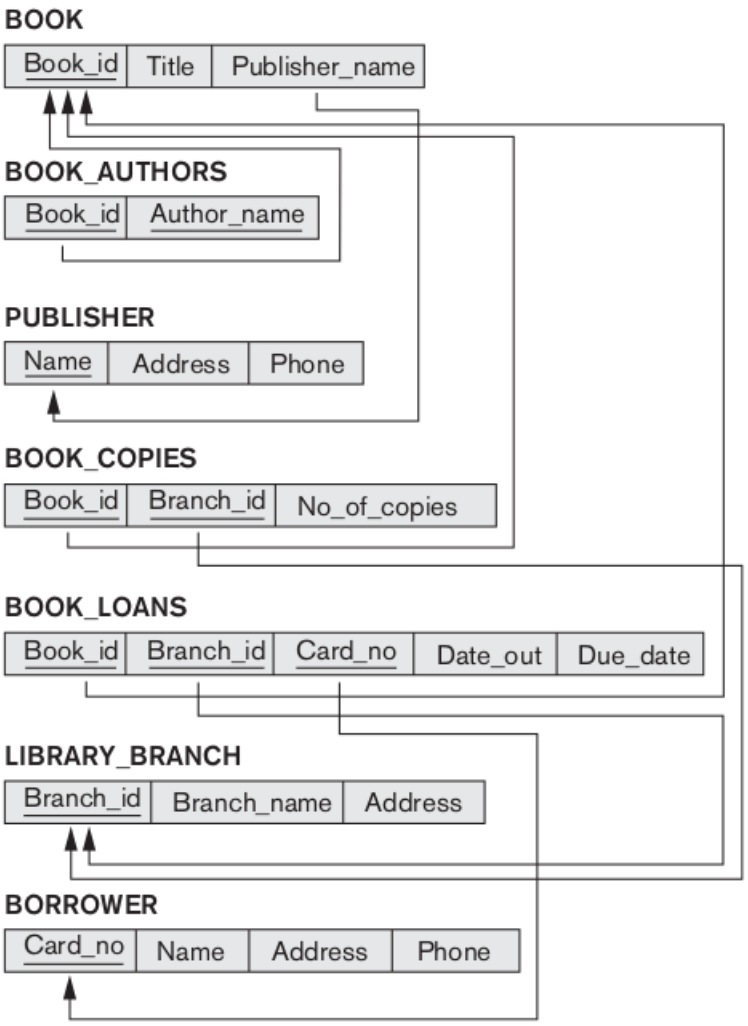


Figure 4.6
A relational database schema for a LIBRARY database.