

BINDURA UNIVERSITY OF SCIENCE EDUCATION

FACULTY OF SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE

BACHELOR OF SCIENCE HONORS DEGREE IN COMPUTER SCIENCE

BACHELOR OF SCIENCE HONORS DEGREE IN COMPUTER SCIENCE / SOFTWARE

ENGINEERING/ NETWORK ENGINEERING

DATABASE CONCEPTS (CS201/NWE111/CSH115) / DATABASE SYSTEMS (SWE114)

DURATION: 3 HOURS

TOTAL MARKS: 100

INSTRUCTIONS TO CANDIDATES

JUN 2025

Answer ALL questions.

The paper consists of **SECTION A** (theory) and **SECTION B** (practical)

Question 1

- a. Explain composite and foreign keys? [4]
- b. State two (2) advantages of normalisation in relational databases [2]
- c. Describe transaction recovery and transaction atomicity and explain their importance in the processing of database transactions. [6]
- d. The terms authorisation and authentication are associated with techniques that maintain database security and integrity. Explain the difference between these terms and outline the basic techniques that they use. [6]

Question 2

Consider the scenario for a book collection and answer the question that follow.

The following rules describe the scenario (Please note, the term 'book' is used to refer to a title rather than a single specific copy of a book):

- An author has a name and surname.
- An author authors one or many books.
- A book has exactly one author.

- A book has exactly one publisher.
- A book has a title and ISBN number.
- A publisher has a name.
- A publisher might exist but might not yet have published a book.
- A publisher publishes one or many books.
- A book can occur in various formats (e.g. e-book, hardback, paperback).
- A new format might be established before any books are published using it.
- A book has a price, which depends on the format.
- A category describes the type of book, e.g. children's book, crime, thriller.
- A book can fit into one or more categories.
- A category contains one or more books.

Draw the entity relationship diagram for the scenario provided using a suitable notation. Your answer must show entities with their attributes and their relationships (including cardinality and optionality). [14]

Question 3

- a. Given the dependency diagram shown in the following figure, (the primary key attributes are underlined)

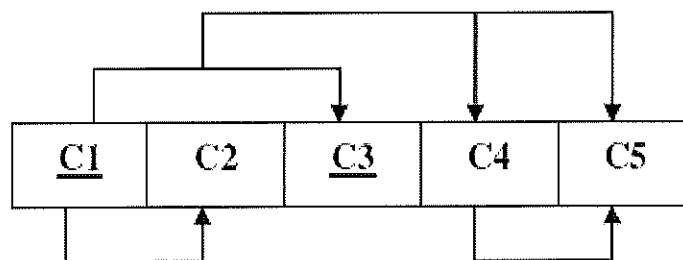


FIGURE: 3.2

- Identify and discuss each of the indicated dependencies? [4]
- Create a database whose tables are at least in 3NF, showing dependency diagram for each table? [6]

b) Consider the following entity relationship diagram for a university.

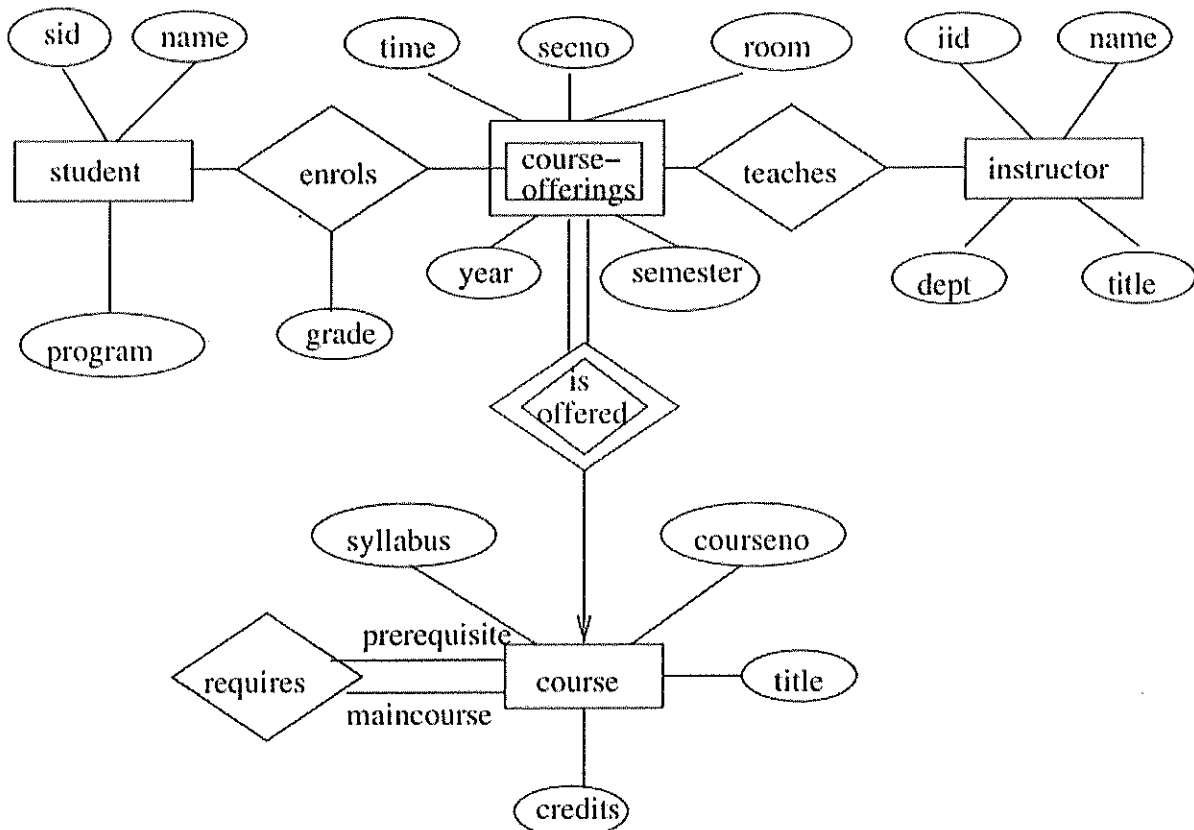


Figure: 3.1 E-R diagram for a university

Map the ERD depicted above (Figure 3.1) to relational schema

[8]

SECTION B (PRACTICAL)

CREATE A FOLDER ON THE DESKTOP AND NAME IT USING YOUR REGISTRATION NUMBER. SAVE ALL YOUR WORK IN THIS FOLDER

Question 4

The following database schema is used to store information about ballet dancers, shows, and companies.

Company (name, city, country)

Dancer (did, name, birthyear, country)

Show (sid, title, choreographer, composer, year)

Role (did, sid, role, company)

The tables contain the following information:

- *Company* stores information about dance companies. The attributes *name*, *city*, and *country* are all strings; we assume for this exam that all companies have unique names.
- *Dancer* stores information about individual dancers. *Did* is a unique integer id for each dancer. *Name* is a string with the dancer's name, *birthyear* is an integer, and the dancer's native *country* is a string.
- *Show* stores information about ballet shows (dances). Each show has a unique integer id *sid*, string attributes for the show *title*, *choreographer*, and *composer*, and an integer *year* in which the show was created.
- *Role* stores information about which dancers have been in which shows, the name of the role (part) they danced, and the company where they danced that part in that particular show. The dancer and show id's are integers, the *role* and *company* names are strings. A dancer may have danced multiple roles in the same show at the same company, or danced the same role in the same show for different companies, and so forth.

Several attributes in *Role* are foreign keys: *did* references *did* in *Dancer*, *sid* references *sid* in *Show*, and *company* references *name* in *Company*. For this exam, assume that all data values are not null. The next page contains some sample data for each of these tables.

A snapshot of the tables constituting the database is shown below:

Company

name	city	country
Imperial	St. Petersburg	Russia
Bolshoi	Moscow	Russia
Ballet Russe	Paris	France
NYCB	New York	USA
PNB	Seattle	USA

Dancer

did	name	birthyear	country
101	Pavlova	1881	Russia
102	Legnani	1863	Italy
103	Gerdt	1884	Russia
104	Ulanova	1910	Russia
105	Duncan	1877	USA
106	Dumas Ang	1994	USA
107	Boal	1965	USA
108	Korbes	1981	Brazil

Show

sid	title	choreographer	composer	year
201	The Swan	Fokine	Saint-Seans	1905
202	Cinderella	Ivanov	Filinhoff	1893
203	Cinderella	Zakharov	Prokofiev	1940
204	Apollo	Balanchine	Stravinsky	1928
205	Swan Lake	Petipa	Tchaikovsky	1895
206	Nutcracker	Balanchine	Tchaikovsky	1954
207	Nutcracker	Stowell	Tchaikovsky	1983

Role

did	sid	role	company
108	204	Terpsichore	NYCB
106	207	Warrior Mouse	PNB
107	204	Apollo	NYCB
101	201	Swan	Ballet Russe
102	202	Cinderella	Imperial
103	202	Prince	Imperial
108	205	White Swan	PNB
108	205	Black Swan	PNB
104	203	Cinderella	Bolshoi

- a. Using SQL, create and populate the database shown above and save it as dance_db, make sure you enforce all the required constraints. [30]
- b. Using SQL, create the following reports and save the reports in the database.
- i. For every dancer who has performed the role 'Black Swan' in the show 'Swan Lake' for one or more companies, create a view which list the name of the dancer and the company name(s), sorted by dancer name. If the dancer has performed that role for more than one company, there should be one line of output for each dancer, company pair. The companies can be listed in any order. [5]
 - ii. List the dancer ids (did) and names of all dancers who have danced in a show choreographed by 'Fosse' but have not danced in a show choreographed by 'Robbins'. Each did/name pair should only appear once in the output. [5]
 - iii. List the dancer ids (did) and names of all dancers born on or before 1950 and who have danced in at least three different shows. If a dancer has danced different roles in the same show, it still only counts once in the total number of shows. Each dancer/did pair should only be listed once. [5]
 - iv. For every dancer who has danced for one or more companies in a different country than where they were born, list the name of the dancer and the names of those companies. [5]

*****END OF PAPER*****