Course: CS3402 Database Systems
Term: 2020-21 Semester B
Assignment #2 (100 marks)
Submission deadline: March 29, 2021

Name: _____

Student ID: _____

[40 marks] Question 1. Examine the Course table shown below.

- CourseID is the primary key
- FD: LeaderID → LeaderName

(a) Describe whether this table is in 1NF? If this table is not in 1NF, describe how you can normalize it to 1NF. [10 marks]

(b) Describe whether this table is in 2NF? If this table is not in 2NF, describe how you can normalize it to 2NF. [10 marks]

(c) Describe whether this table is in 3NF? If this table is not in 3NF, describe how you can normalize it to 3NF. [20 marks]

CourseID	CourseName	LeaderID	Semester	LeaderName
CS3023	Operating Systems	3453	A	Peter Chan
CS1134	C++ Programming	5644	B	Bob Mak
CS5645	Internet Security	6635	A	Sally Wong
CS3444	Data Structures	4322	A	John Tam

Answers:

(a) This table is in 1NF. [5 marks]

This table has no multivalued attributes and nested relations. [5 marks]

(b) This table is in 2NF. [5 marks]

Every nonprime attribute in this table is fully functionally dependent on the primary key. (or there is only attribute in the primary key) [5 marks]

(c) [10 marks] There exists a non-key attribute transitively dependent on the key, i.e., LeaderName depends on LeaderID and LeaderID depends on CourseID, i.e., CourseID => LeaderID and LeaderID => LeaderName

[10 marks] To normalize this table to 3NF, create another relation which specifically captures the dependency LeaderID => LeaderName

Course(CourseID, CourseName, LeaderID, Semester)
CourseLeader(LeaderID, LeaderName)

Name: _____

Student ID: _____

[30 marks] Question 2. Examine the table shown below and the set of functional dependency on its attributes:

- CourseRmAlloc (CourseId, CourseName, Year, Lecturer, Enrollment, RoomId, RoomCapacity, Day, Time)
- FDs:
  CourseId → CourseName
  CourseName → CourseId
  {CourseId, Year} → Lecturer
  {CourseId, Year} → Enrollment
  RoomId → RoomCapacity
  {RoomId, Year, Day, Time} → CourseId
  {CourseId, Year, Day, Time} → RoomId

(a) Is {CourseId, Year} a candidate key? Show the details of determining the closure of {CourseId, Year} for your answer. [15 marks]

(b) Is {Year, Day, Time, CourseId} a candidate key? Show the details of determining the closure of { Year, Day, Time, CourseId } for your answer. [15 marks]

Answers:

(a) [15 marks]

X = {CourseId, Year}

Iteration	**oldX^+ (Before)**	**FD**	**X^+ (After)**	**X^+ = oldX^+?**
1	{CourseId, Year}	CourseId → CourseName {CourseId, Year} → Lecturer {CourseId, Year} → Enrollment	{CourseId, CourseName, Lecturer, Enrollment}	False
2	{CourseId, CourseName, Lecturer, Enrollment}	CourseName → CourseId	{CourseId, CourseName, Lecturer, Enrollment}	True

X^+ = {CourseId, CourseName, Lecturer, Enrollment} (Since X^+ does not include all the attributes in R, X = {CourseId, Year} is not a candidate key.)

Name: _____

Student ID: _____

(c) [15 marks]

X = {Year, Day, Time, CourseId}

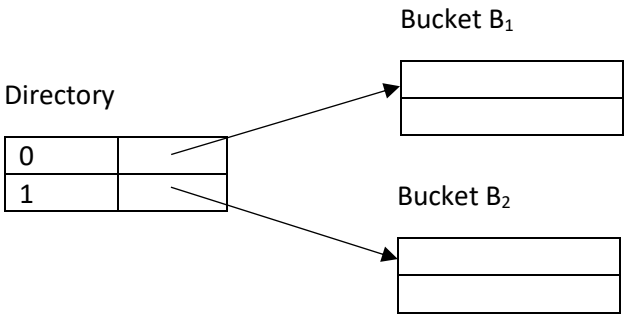
Iteration	**oldX^+ (Before)**	**FD**	**X^+ (After)**	**X^+ = oldX^+?**
1	{Year, Day, Time, CourseId}	CourseId → CourseName {CourseId, Year} → Lecturer {CourseId, Year} → Enrollment {CourseId, Year, Day, Time} → RoomId	{Year, Day, Time, CourseId, CourseName, Lecturer, Enrollment, RoodId}	False
2	{Year, Day, Time, CourseId, CourseName, Lecturer, Enrollment, RoodId}	CourseName → CourseId RoomId → RoomCapacity {RoomId, Year, Day, Time} → CourseId	{Year, Day, Time, CourseId, CourseName, Lecturer, Enrollment, RoomId, RoomCapacity}	False
3	{Year, Day, Time, CourseId, CourseName, Lecturer, Enrollment, RoodId, RoodCapacity}		{Year, Day, Time, CourseId, CourseName, Lecturer, Enrollment, RoomId, RoomCapacity}	True

X^+ = {Year, Day, Time, CourseId, CourseName, Lecturer, Enrollment, RoomId, RoomCapacity}
(Since X^+ includes all the attributes in R, X = {Year, Day, Time, CourseId} is a candidate key.)

Name: _____

Student ID: _____

[30 marks] Question 3. Suppose we use the extensible hashing method to store items with 4-bit hash values and each bucket can store two hash values. Initially, the global depth is 1; the directory and the 2 empty data file buckets B_1 and B_2 are depicted below:



(a) What are (i) the index of the directory (the initial index of the directory is 0 and 1) and (ii) the content stored in buckets B_1 and B_2 after inserting 4 items with hash values 0001, 0111, 1100, and 1010? (Please describe your answer without drawing any diagram.) [10 marks]

(b) What are (i) the new global depth, (ii) the index of the directory (including the index and the address of its corresponding bucket) and (iii) the content stored in each bucket after further inserting 1 item with hash value 1111 to your answer from part (a)? (Please describe your answer without drawing any diagram.) [20 marks]

Answers:

(a)

(i) The index of the directory is still 0 and 1. [2 marks]
(ii) Bucket B1 stores 0001 [2 marks] and 0111 [2 marks] and bucket B2 stores 1100 [2 marks] and 1010 [2 marks].

(b)
(i) The global depth is increased from 1 to 2. [2 marks]
(ii) The index of the directory is 00, 01, 10 and 11. [2 marks]
The entries of 00 and 01 store the address of B1. [4 marks]
The entry of 10 stores the address of B2. [2 marks]
The entry of 11 stores the address of a new bucket B3. [4 marks]
(iii) B1 stores 0001 and 0111 [2 marks], B2 stores 1010 [2 marks], and B3 stores 1100 and 1111. [2 marks]