

## Homework Assignment

**(Deadline: Friday, 15-Nov-2024, 11:59pm. Submit via Blackboard)**

### Question A. Integrity Constraints

**[35 marks]**

The Department of Computing wants to create a database to manage the information about its staff, students, and courses. The tables, attributes in each table, and data type of each attribute are summarized below, along with some detailed requirements for the information to be stored in the database. Write SQL statements to create these tables and specify the integrity constraints, e.g., domain constraint (including custom domains), not null, primary key, foreign key, etc.

Attribute	Data Type	Requirement
<b>Staff</b>		
ID	INTEGER	Each staff must have a unique ID.
Name	VARCHAR(50)	Each staff must have a name.
Email	VARCHAR(50)	Email must be in the form of '...@staff.comp.polyu.hk'.
Position	VARCHAR(20)	The department offers three positions: 'Assistant Professor', 'Associate Professor', 'Professor'.
Office	CHAR(5)	Office must be in the form of 'PQ...'.
Salary	INTEGER	Monthly salary must be between 80000 and 160000.
<b>Student</b>		
ID	INTEGER	Each student must have a unique ID
Name	VARCHAR(50)	Each student must have a name.
Email	VARCHAR(50)	Email must be in the form of '...@student.comp.polyu.hk'.
Type	VARCHAR(20)	Student type is either 'Domestic' or 'International'.
Major	CHAR(2)	The department offers four majors: 'CS', 'IT', 'DS', 'AI'.
AvgGrade	DECIMAL(2,1)	Grade must be between 0 and 4.3.
Advisor	INTEGER	Each student must be assigned a staff as his/her advisor; A staff cannot leave the department if he/she is the advisor of some students.
<b>Course</b>		
ID	INTEGER	Each course must have a unique ID.
Name	VARCHAR(50)	Each course must have a name.
Credits	INTEGER	Credits must be between 1 and 6.
<b>Teach</b>		
StaffID	INTEGER	In each semester ('Spring', 'Summer', or 'Fall'), a staff can teach more than one course; a course can be co-taught by multiple staff.
CourseID	INTEGER	
Semester	CHAR(6)	
Evaluation	DECIMAL(2,1)	A staff can teach the same course multiple times in different semesters; all their evaluations (between 0 and 4) must be recorded in the database. Staff can teach only the courses offered by the department; Courses can be taught only by staff from the department. When a staff leaves the department or a course is closed, all the teaching and evaluation information must be deleted from the database.

Enrolment		
StudentID	INTEGER	Each student can take more than one course; each course can be enrolled by many students.
CourseID	INTEGER	
Grade	DECIMAL(2,1)	<p>A student can take the same course multiple times, but only the last grade is recorded in the database; The grade must be between 0 and 4.3.</p> <p>Students can take only the courses offered by the department; Courses are only opened for students from the department.</p> <p>When a student leaves the department, all his/her enrolment information must be deleted from the database; A course cannot be closed if there are students enrolled in it.</p>

**Question B. Functional Dependency and Normalization****[25 marks]**

- (1) Consider relation  $R(A, B, C, D, E, F)$  and the set of functional dependencies  $\mathcal{F} = \{AB \rightarrow C, AC \rightarrow B, AD \rightarrow E, B \rightarrow D, BC \rightarrow A, E \rightarrow BCF\}$ . Find all the candidate keys for  $R$  by calculating the attribute closures.
- (2) Consider relation  $R(A, B, C, D, E, F, G, H)$  and the set of functional dependencies  $\mathcal{F} = \{A \rightarrow CD, C \rightarrow EF, G \rightarrow A, CE \rightarrow F, BG \rightarrow H\}$ . Determine the highest normal form of  $R$ , and decompose  $R$  into 2NF, 3NF and then BCNF relations. Show your steps of decomposition.

**Question C. File Organization and Indexing****[40 marks]**

Consider the relation: *Vehicle* (VID, *Maker*, *Model*, *Color*, *Price*), which contains 6400 records with each record occupying 50 bytes.

- *VID* (Integer, 4 bytes) is the primary key of the relation; the values are between 1 and 6400.
- *Maker* contains 80 distinct values; the records are evenly distributed among these values.
- *Color* contains 10 distinct values; the records are evenly distributed among these values.

The file is sorted by *VID*, and stored on a disk with the following configuration:

- Block size = 1000 bytes
- Block pointer size = 6 bytes

- (1) What is the total cost (i.e., number of block accesses) to retrieve all records with  $VID > 6000$  using linear scan? What is the total cost to retrieve these records using binary search?
- (2) A primary index is built on the *VID* field. What is the total cost to retrieve all records with  $VID > 6000$  using the primary index?
- (3) A  $B^+$  tree index is built on the *VID* field, and each tree node is 60% full on average. What is the total cost to retrieve all records with  $VID > 6000$  using the  $B^+$  tree index? How would the total cost change if the file is not sorted by *VID*?
- (4) A bitmap index is built on both *Maker* and *Color* fields. How many blocks are required to store the bitmap index (*Note*: 1 byte = 8 bits; each bitmap is stored as a fixed-length record)? Assume there are 40 records with *Maker* = 'Toyota' and *Color* = 'Black'. What is the total cost to retrieve these records using the bitmap index?

Student Name: \_\_\_\_\_

Student ID: \_\_\_\_\_

Provide your answers for Question A on this page.

Provide your answers for Question B on this page.

Provide your answers for Question C on this page.