RollNo: ----------------------------

Thapar Institute of Engineering & Technology, Patiala
Computer Science and Engineering Department
Auxiliary Examination

Course Code: UCT402	Course Name: Database Management Systems
August 28, 2023	Sunday, 5.30 PM – 8.30PM
Time: 3 Hours, M. Marks: 50	Name of Faculty: Dr. Rajendra Kumar Roul

(Note: Answer all the questions with valid points only, which are most appropriate for your answers. Unnecessary answers or answers written in pencil will not be evaluated. Assume the missing information (if any) suitably. All the symbols used here have their usual meaning.)

Q1. a) Consider a relation R (A, B, C, D, E, H). Find the canonical cover of the following functional dependencies holds in R. **4M**

A→B, ABCD→E, EF→G, EF→H, ACDF→EG

b) Discuss different normal forms with example. **6M**

Q2. Consider two relations **Student** (rollno, name, class_no, mark) and **Course** (course_id, rollno, course_name). In Course table, rollno is a foreign key with respect to the primary key column 'rollno' of Student table. Write the sql statements for the following queries. **(5 x 2 = 10M)**

a) Display those student details who got the same marks as the minimum marks of each class.
b) Display student names and their course_id even for a course there is no student.
c) Display the student details of each class who got minimum marks.
d) Increment the marks by 10 for those students whose name's third character is 'a' and fifth character is 's', and exactly eight characters in length.
e) To display all those class_no that have an average mark less than that of class_no 10.

Q3. Consider the following two transactions:

```
T1: read(A)
    read(B)
    if A=0 then B: =B+1; end if;
    write(B)
```

```
T2: read(B)
    read(A)
    if A=0 then B: =B+2; end if;
    write(B)
```

Add lock and unlock instructions to T1 and T2 so that they observe the following locking protocols.
i) Two-phase locking protocol (2PL) **2M**
ii) Strict 2PL **2M**
iii)Rigorous 2PL **2M**
iv)Does the execution of these two transactions result in a deadlock? Explain with an example. **2M**

Q4. Consider the following schedule S with four transactions T1, T2, T3, and T4.

T1	T2	T3	T4
			Read(x)
			Read(z)
	Read(x)		
		Read(x)	
Write(y)			
	Write(x)		
		Read(y)	
		Write(y)	
		Commit	
	Write(y)		
	Commit		
Write(x)			
Commit			

i) State whether S is conflict serializable or not by using precedence graph. **2M**
ii) If S is conflict serializable then determine all the possible serializable schedules. **2M**
iii) State whether S is cascadeless or not with proper reason. **2M**
iv) State whether S is strict schedule or not with proper reason. **2M**
v) State whether S is recoverable or not with proper reason. **2M**
vi) List all the conflicting operations in S. **2M**

Q5. Consider the relations **EMP** (id, name, salary, bonus) and **Overtime** (id, no-of-extra-hours).
Initially the bonus field is initially NULL for all the employees. Write a **stored function** which will update the salary of each employee by adding his/her bonus based on the following conditions:

   i)   If the no-of-extra-hours is between 1 and 2: bonus = 10% of salary
   ii)   If the no-of-extra-hours is between 3 and 4: bonus = 20% of salary
   iii)   If the no-of-extra-hours is between 5 and 6: bonus = 30% of salary
   iv)   If the no-of-extra-hours is >=7:   bonus = 40% of salary
   v)   If the no-of-extra-hours is <1:   No bonus

The function should finally return the total salaries of the '**n' highest paid employees** using **parameterized cursor**. The value of '**n**' is passed to the cursor as a parameter from the function. **10M**

****