DEPARTMENT OF COMPUTER SCIENCE

Prof. Dr. Nikolaus Augsten
Jakob-Haringer-Str. 2
5020 Salzburg, Austria
Phone: +43 662 8044 6347
E-Mail: nikolaus.augsten@plus.ac.at

PARIS LODRON UNIVERSITÄT SALZBURG

Non-Standard Database Systems — Exam
Summer Semester 2024 — 25.06.2024

Name: ______________________ **Student ID:** ______________________

Hints

- Check whether you received all pages of the exam (9 pages).
- Write your name or your student ID on each sheet of the exam and hand in all pages.
- All answers are expected to be written on the exam sheets.
- Clearly highlight and enumerate additional pages that are used for longer answers. Match your text with the according exercise.
- Only use pencils that are permanent and non-red colored.
- Use the notation and techniques discussed in the lecture.
- Exercises with more than one solution are not graded.
- You are allowed to use one A4 sheet with your personal notes (both sides, hand written or printed).
- Exam duration: 90 minutes

Signature ______________________

Grading — Filled by the examiner

Exercise	1	2	3	4	5	6	7	8	Sum
Total points	2	2	2	2	2	2	2	2	16
Points reached									

Exercise 1 - *Throughput and Response Time.* 2 Points

A database system must process transactions t_1, t_2, t_3, and t_4. For each transaction, Table 1 shows the start time (when the transaction enters the system) and the execution time (the runtime of the transaction in the system when it is not interrupted).

Compute the average response time and throughput for the following task scheduling strategies where shorter transactions are given priority over longer transactions (i.e., the waiting queue is ordered by execution time):

1. No running transaction is aborted.
2. A running transaction is aborted and put back to the waiting queue for re-execution if a shorter transaction (i.e., smaller execution time) enters the system.
3. A running transaction is aborted and put back to the waiting queue for re-execution if a much shorter transaction enters the system; a transaction is much shorter if it requires at most half of the execution time.

Transaction	Start Time	Execution Time
t_1	0	3
t_2	2	2
t_3	6	5
t_4	8	2

Table 1: Start and execution time of transactions.

Exercise 2 - *Network Architectures.* 2 Points

Pick which of the following interconnection network architectures with n system components satisfy the following statements.

Grading: Positive marking rewards each correct answer with 0.5p. Each incorrect marking is penalized with 0.25p. Questions without any marking are considered unanswered and will not be graded.

1. All n system components are directly connected to each other.
   ○ Bus ○ Mesh (no wrap around) ○ Hypercube
2. Every node can reach the same number of nodes in one hop.
   ○ Bus ○ Mesh (no wrap around) ○ Hypercube
3. The number of hops to send a message to a node is at most proportinal to $\sqrt{n}$.
   ○ Bus ○ Mesh (no wrap around) ○ Hypercube
4. The number of communication links grows with the number of components.
   ○ Bus ○ Mesh (no wrap around) ○ Hypercube

Exercise 3 - *Partitioning Vector.* 2 Points

The histogram in Table 2 shows the value distribution of a partitioning attribute of type integer. Use the histogram to define a partitioning vector for four disks that avoids data skew. Assume a uniform distribution within each value range.

Value Range	Number of Values
$0 \dots 3$	20
$4 \dots 8$	10
$9 \dots 16$	40
$17 \dots 21$	50
$22 \dots 23$	10
$24 \dots 29$	20

Table 2: Histogram for partitioning vector.

Exercise 4 - *2-Phase-Commit (2PC)*. 2 Points

Transaction T is initiated at site S_i with coordinator C_i, $1 \leq i \leq n$, and is executed at sites S_k, $1 \leq k \leq n$. Discuss the following situations:

1. Site S_k $(k \neq i)$ violates the 2PC protocol and sends the `ready T` message before writing all log records to stable storage.
2. Site S_k $(k \neq i)$ violates the 2PC protocol and releases the locks before the end of Phase 2.

Exercise 5 - *Availability*. 2 Points

Answer the following questions regarding availability in distributed systems.

1. Explain why network partitioning and site failures are generally indistinguishable in distributed systems.
2. Explain how a system can achieve availability even if a (minor) network partition occurs?

Exercise 6 - *Distributed Lock Manager.* 2 Points

A database system with full replication is distributed over five data centers across the globe with the processing nodes p_j distributed over the data centers d_i as follows: $d_0 = \{p_0, p_1, p_2\}$, $d_1 = \{p_3, p_4, p_5\}$, $d_2 = \{p_6\}$, $d_3 = \{p_7, p_8, p_9\}$, $d_4 = \{p_{10}, p_{11}\}$. The latency within a data center or between any two adjacent data centers d_i and $d_{(i+1) \bmod 4}$ is low, the latency to a non-adjacent data center is high. Configure an efficient quorum consensus based technique for reading and writing replicas that guarantees strong consistency.

Exercise 7 - *Deadlocks*. 2 Points

Explain how false cycles can lead to an unnecessary rollback based on an example.

Exercise 8 - *Vector clocks.* 2 Points

Assume a single data item Q that is replicated on sites S_1, S_2, and S_3. A site S_i can do (i) a local write on Q, $W(Q)$, which changes the value of the local copy Q_i, or (ii) copy the value from a different site S_j, $C(S_j)$, $j \neq i$, which copies the value of Q_j to Q_i. All vectors are initialized with the zero vector.

1. Show the vector clocks resulting from the schedule in Figure 1.
2. Identify the first operation that leads to a branching history.

Note: Local reads, which will typically precede a local write in a real schedule, are not relevant for conflict detection and omitted from the schedule.

S_1	S_2	S_3
$W(Q)$		
	$W(Q)$	
		$C(Q_2)$
		$C(Q_1)$
		$W(Q)$
$C(Q_3)$		
	$C(Q_1)$	
$W(Q)$		
$W(Q)$		
		$W(Q)$

Figure 1: Schedule on replicated data item Q.