# Midterm Grading

Name:

## Task 1: Autocommit

Expected Result: Rows 1 and 2 are inserted because of autocommit: each statement is treated as transaction. rows 3-5 are not inserted because transaction is rolled back because of duplicate key error.

Possible Points: 5

Your Points: 3

Remarks: Correctly described the table state but lacked detailed explanation of autocommit behavior and transaction rollback.

## Task 2: Transactions

### 2.1 Expected Result

Possible Points: 3

Your Points: 2

Remarks: Provided a step-by-step analysis but lacked detailed conflict explanation.

### 2.2 Strict 2PL

Possible Points: 3

Your Points: 2

Remarks: Correctly described the execution but had an incomplete explanation of lock acquisition and release.

### 2.3 MVCC Repeatable Read

Possible Points: 3

Your Points: 2

Remarks: Correctly described the execution but missed some details about versioning and commit order.

## Task 3: Waits-for-Graph

Expected Result: Transaction n+1 requesting a s-lock: Only 1 transaction can hold a x-lock on any object at any time. S-locks on objects are compatible locks. When another t requests a s-lock and gets involved in the waits-for-graph this means that one of the existing transactions holds an x-lock on the object. 🡪 the new transaction waits for exactly this one t holding the x-lock 🡪 1 more edge is added to the graph

Possible Points: 6

Your Points: 4

Remarks: Correct description but lacked explanation for some details.

## Task 4: Write Skew

Expected Result: T1 / T2: A "predicate lock" is put on all rows that match the search result of select count(\*) from test\_ssi where on\_duty = true.

Possible Points: 2

Your Points: 1

Remarks: Correctly described the behavior but missed some details.

## Total Points:

14