

**Purdue University**  
**Computer Science Department**  
**CS44800 Homework 5**  
**Fall 2024**

Total Points: **3 points**  
Submit via **Brightspace**

Provide concise but to the point explanation to support your answers. If possible, use bullets to organize the ideas in your answer.

**Question 1. (1.5 point)**

Consider the following schedules, where R, W, C stands for ‘Read’, ‘Write’ and ‘Commit’ respectively and subscripts refers to transactions (i.e.,  $R_1(X)$  means transaction  $T_1$  read X):

**Schedule 1:**  $R_1(X), W_1(X), R_3(X), W_3(X), W_2(X), R_1(Y), W_1(Y), C_1, W_3(Y), C_3, R_2(Y), W_2(Y), C_2$

**Schedule 2:**  $R_2(X), W_2(X), R_3(Y), W_3(Y), R_3(Z), W_3(Z), C_3, R_2(Z), W_2(Z), C_2, R_1(X), W_1(X), C_1$

**Schedule 3:**  $W_1(A), W_2(B), W_3(C), R_1(X), R_2(X), R_1(Y), W_1(X), C_1, W_2(Y), C_2, W_3(Y), C_3$

For each schedule answer the following question:

- (a) Is this schedule conflict serializable? If yes, provide an equivalent serial schedule of transactions.
- (b) Does the schedule avoid cascading aborts? Explain your answer.
- (c) Assume 2PL protocol where operations are skipped if the transaction cannot acquire a lock and deadlocks are handled using a wait-for graph, like Project 4. Furthermore, once a transaction commits, all the locks held are released and the operations that were on hold are retried – according to the priority in the schedule. Is the schedule possible under this 2PL protocol (i.e. all transactions can be committed)? Explain your answer. If the 2PL protocol leads to a deadlock, give the wait-for graph. Note: operations that were skipped will be re-tried later.

**Question 2. (1.5 point)**

- (a) Discuss the different types of transaction failures. Explain each failure.
- (b) What is the system log used for?
- (c) What are checkpoints and why are they important?