Table of Contents

What are Schedules ?
Types of schedule
Example of serial schedule
Example of non serial schedule
Schedule recoverability
Example of Schedule recoverability

# What are Schedules ?

Schedules are sequences that indicate the chronological order in which instructions of concurrent transactions are executed.
A schedule must preserve the order in which the instructions appear in each individual transaction.

# Types of schedule

1. Serial schedule
2. Non serial schedule

## Example of serial schedule

Transaction2 starts after completion of transaction1.

| Transaction1 | Transaction2 |
|---|---|
| Read(X); | |
| X=X-1000; | |
| Write(X); | |

| | |
|---|---|
| Read(Y); | |
| Y=Y-1000; | |
| Write(Y); | |
| | Read(X); |
| | X=X+1000; |
| | Write(X) |

## Example of non serial schedule

Transaction2 start while transaction1 was in process.

| Transaction1 | Transaction2 |
|---|---|
| Read(X); | |
| X=X-1000; | |
| | Read(X); |
| | X=X+1000; |
| Write(X); | |
| Read(Y); | |
| | Write(X); |
| Y=Y-1000; | |
| Write(Y) | |

## Schedule recoverability

Sometimes a transaction may be incomplete due to a software issue, system crash or hardware failure.

Than failed transaction needed to be rollback.

A transaction needed to be saved at each point.

In SQL we use commit to save the transactions state.

This example uses commit at proper point, which maked it recoverable schedule. In case of fail it can be roll back to commit point.

### Example of Schedule recoverability

| Transaction1 | Transaction1Buffer | Transaction2 | Transaction2Buffer | Database |
|---|---|---|---|---|
| | | | | X = 1000 |
| Read(X); | X = 10000 | | | X = 10000 |
| X=X-1000; | X = 9000 | | | X = 10000 |
| Write(X); | X = 9000 | | | X = 9000 |
| Commit; | | | | |
| | | Read(X); | X = 9000 | X = 9000 |
| | | X=X+1000; | X = 10000 | X = 9000 |
| | | Write(X); | X = 10000 | X = 10000 |

| | | | | |
|---|---|---|---|---|
| | | Commit; | | |

Check out this video lecture, for more information on schedule.