Past Paper

Intro

-
  - process vs thread

Automata composition

Safety and liveness

-

Safety

mutex

-
  - recursive

Semaphore

-
  - bug finding

MRSW

-
  - liveness

CCR, Monitors, ProgL

-
  - monitor, condition variable

-
  - Monitor
    - implicit Mutual exclusion via mutex acquired when entering
    - explicit Conditional synchronisation occurs via `signal()` and `wait()`
    - vs CCRs

-
  - Signaling semantics of condition variables

Liveness and Deadlock

-

- deadlock and livelock, conditions
    - Mutex, Crash recovery
  - y2014p5q8 (c)
    - Deadlock, partial order, and their implications

  - y2021p5q7 (b)
    - Banker's algorithm

  - y2020p5q7 (c,d)
    - priority inversion

Without shared data

Message Passing

- y2022p5q4(c)
  - shared, global variable behaviour emulation

Transactions

Database Concurrency Control

- y2021p5q7(c,d)
  - [Non]-Strict Isolation
  - Atomic operation

- y2011p5q7
  - conflicting, cascading aborts
  - TSO

- y2012p5q8
- y2016p5q8
  - history graphs, serial