

## Übungsblatt 11

**Übung 11.1** Zeigen Sie, dass *Lost Update* und *Nonrepeatable Read* immer gemeinsam auftreten, d.h. dass bei einem Schedule mit einem *Lost Update* auch ein *Nonrepeatable Read* auftreten kann und umgekehrt.

**Übung 11.2** In der Vorlesung wurde plausibel gemacht, dass *Konflikt-Serialisierbarkeit* (keine Konflikt-Operationen vertauscht) hinreichend für *Serialisierbarkeit* (identische Auswirkungen auf die Datenbank) ist.

Geben Sie ein Beispiel dafür an, dass *Konflikt-Serialisierbarkeit* *nicht notwendig* für *Serialisierbarkeit* ist, d.h. ein Beispiel, das serialisierbar, aber nicht konflikt-serialisierbar ist.

**Übung 11.3** Betrachten Sie die beiden folgenden Transaktionen (dargestellt sind nur *read* und *write* Operationen):

$$T_1 : r_1(X); w_1(X); r_1(Y); w_1(Y)$$
$$T_2 : r_2(Y); w_2(Y); r_2(X); w_2(X)$$

- Wieviele mögliche Ausführungspläne gibt es?
- Wie viele Ausführungspläne sind konfliktäquivalent zur seriellen Ausführung  $(T_1, T_2)$ ?
- Geben Sie einen nicht serialisierbaren Ausführungsplan an und prüfen Sie, welche Auswirkung das Zwei-Phasen Sperrprotokoll auf diesen Plan hätte.

**Übung 11.4** Betrachten Sie die folgenden Schedules von drei nebenläufigen Transaktionen:

$$S1 : r_1(X); r_2(X); r_3(Y); w_1(X); r_2(Z); r_2(Y); w_2(Y); w_1(Z)$$
$$S2 : w_3(X); r_1(X); w_1(Y); r_2(Y); w_2(Z); r_3(Z)$$

- Geben Sie die zugehörigen Präzedenzgraphen an.
- Sind die Schedules serialisierbar? Wenn ja, geben Sie die äquivalenten seriellen Schedules an.