Øving 18

# Oppgave 1: Serialiserbarhet

1. Utførelse 1 (T1 – T2 – T3):

(4 \* 3), men settes så til 2

Utførelse 2 (T1 – T3 – T2):

Økes med 4, settes til 2, og ganges med 3 🡪 6

Utførelse 3 (T2 – T1 – T3):

0\*3, økes med 4, settes til 2, 🡪 2

Utførelse 4 (T2 – T3 – T1):

0\*3, settes til 2, og økes med 4, 🡪 6

Utførelse 4 (T3 – T1 – T2):

Settes til 2, økes med 4, ganges med 3, 🡪 18

Utførelse 6(T3 – T2 – T1):

Settes til 2, ganges med 3, økes med 4, 🡪 10

1. Utførelse 1 (T1 – T2 – T3):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tid | Transaksjon T1 | Verdi som T1 ser | Transaksjon T2 | Verdi som T2 ser | Transaksjon T3 | Verdi som T3 ser |
| 1 | select beløp into variabel1 | 0 |  |  |  |  |
| 2 |  |  | select beløp into variabel2 | 0 |  |  |
| 3 |  |  |  |  | select beløp into variabel3 | 0 |
| 4 | * variabel1 økes med 4 * update beløp med verdien i variabel1   (0 + 4) | 4 |  |  |  |  |
| 5 |  |  | * variabel2 ganges med 3 * update beløp med verdien i variabel2   (0\*3) | 0 |  |  |
| 6 |  |  |  |  | * variabel3 settes lik 2 * update beløp med verdien i variabel3   (=2) | 2 |
| 7 | commit |  |  |  |  |  |
| 8 |  |  | commit |  |  |  |
| 9 |  |  |  |  | commit |  |

Endelig svar på denne utførelsen blir 2

Utførelse 2 (T2 – T3 – T1):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tid | Transaksjon T1 | Verdi som T1 ser | Transaksjon T2 | Verdi som T2 ser | Transaksjon T3 | Verdi som T3 ser |
| 1 |  |  | select beløp into variabel2 | 0 |  |  |
| 2 |  |  |  |  | select beløp into variabel3 | 0 |
| 3 | select beløp into variabel1 | 0 |  |  |  |  |
| 4 |  |  | * variabel2 ganges med 3 * update beløp med verdien i variabel2   (0\*3) | 0 |  |  |
| 5 |  |  |  |  | * variabel3 settes lik 2 * update beløp med verdien i variabel3   (=2) | 2 |
| 6 | * variabel1 økes med 4 * update beløp med verdien i variabel1   (0 + 4) | 4 |  |  |  |  |
| 7 |  |  | commit |  |  |  |
| 8 |  |  |  |  | commit |  |
| 9 | commit |  |  |  |  |  |

Endelig svar blir 4

Utførelse 3 (T3 – T1 – T2):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tid | Transaksjon T1 | Verdi som T1 ser | Transaksjon T2 | Verdi som T2 ser | Transaksjon T3 | Verdi som T3 ser |
| 1 |  |  |  |  | select beløp into variabel3 | 0 |
| 2 | select beløp into variabel1 | 0 |  |  |  |  |
| 3 |  |  | select beløp into variabel2 | 0 |  |  |
| 4 |  |  |  |  | * variabel3 settes lik 2 * update beløp med verdien i variabel3   (=2) | 2 |
| 5 | * variabel1 økes med 4 * update beløp med verdien i variabel1   (0 + 4) | 4 |  |  |  |  |
| 6 |  |  | * variabel2 ganges med 3 * update beløp med verdien i variabel2   (0\*3) | 0 |  |  |
| 7 |  |  |  |  | commit |  |
| 8 | commit |  |  |  |  |  |
| 9 |  |  | commit |  |  |  |

Det endelige svaret blir 0.

1. Utførelsen S1-S3-U1-U3-S2-U2 er seriell fordi de utføres etter hverandre. U3 er satt til 2 og det har ingen ting å si hva som er før.
2. Fordi det er flere som utføres etter hverandre slik at den ene må egentlig være ferdiggjort før den neste startes på. U3 kan ikke oppdatere beløp når U1 har satt eksklusiv lås på beløp.

# Oppgave 2: Isolasjonsnivå og låsing i ulike databasesystem

1. Får ingen resultat, men det skal egentig bli en rad i a med 0, og en i b med 1.
2. Det blir endret uten noe problem.
3. Vi får «deadlock», men resultatet blir skrevet ut.