INF3100: Databasesystemer – Oppgavesett 9

Oppgave 18.8.1: Below are several sequences of events, including start events, where st_i means that transaction T_i starts. These sequences represent real time, and the timestamp scheduler will allocate timestamps to transactions in the order of their starts. Tell what happens as each executes.

a) $st_1; r_1(A); st_2; w_2(B); r_2(A); w_1(B);$

b) $st_1; st_2; r_1(A); r_2(B); w_2(A); w_1(B);$

Oppgave 18.x.1: For hvert av tilfellene (a), (b) og (c): Beskriv hva som skjer hvis vi forsøker å eksekvere dem med SI-protokollen FUW (First Updater Wins). Sett selv inn operasjoner av formen

- $l_i(x)$ – T_i ber om eksklusiv lås på element x
- $u_i(x)$ – T_i frigir låsen på x
- c_i – commit
- a_i – abort

(a) $r_1(a); r_2(a); r_1(b); w_2(a); w_1(a); r_2(b); w_2(b);$

(b) $r_1(a); r_2(b); w_2(b); r_1(b); w_1(a); w_1(b);$

(c) $r_1(a); r_2(a); r_3(a); w_1(b); r_3(b); w_2(a); w_3(a); w_2(b);$

Oppgave 19.1.2: Suppose that each of the sequences of actions below is followed by an abort action from transaction T_1. Tell which transaction needs to be rolled back.

a) $r_1(A); w_1(B); r_3(B); w_3(C); r_2(C); w_2(D);$

b) $r_3(A); r_2(A); r_1(A); w_1(B); r_3(B); r_2(B); w_3(C); r_2(C);$

c) $r_3(A); r_2(A); r_1(A); w_1(B); r_2(B); w_3(C); r_2(C);$

d) $r_1(A); r_3(B); w_1(B); w_3(C); r_2(B); r_2(C); w_2(D);$

Oppgave 19.2.1: For each of the sequences of actions below, assume that shared locks are requested immediately before each read action, and exclusive locks are requested immediately before every write action. Also, unlocks occur immediately after the final action that a transaction executes. Tell what actions are denied, and whether deadlock occurs. Also tell how the waits-for graph evolves during the execution of the actions. If there are deadlocks, pick a transaction to abort, and show how the sequence of actions continues.

a) $r_1(A); r_3(B); r_2(C); w_1(B); w_3(C); w_2(D);$

d) $r_1(A); r_3(B); w_1(C); r_2(D); r_4(E); w_2(B); w_3(C); w_4(A); w_1(D);$