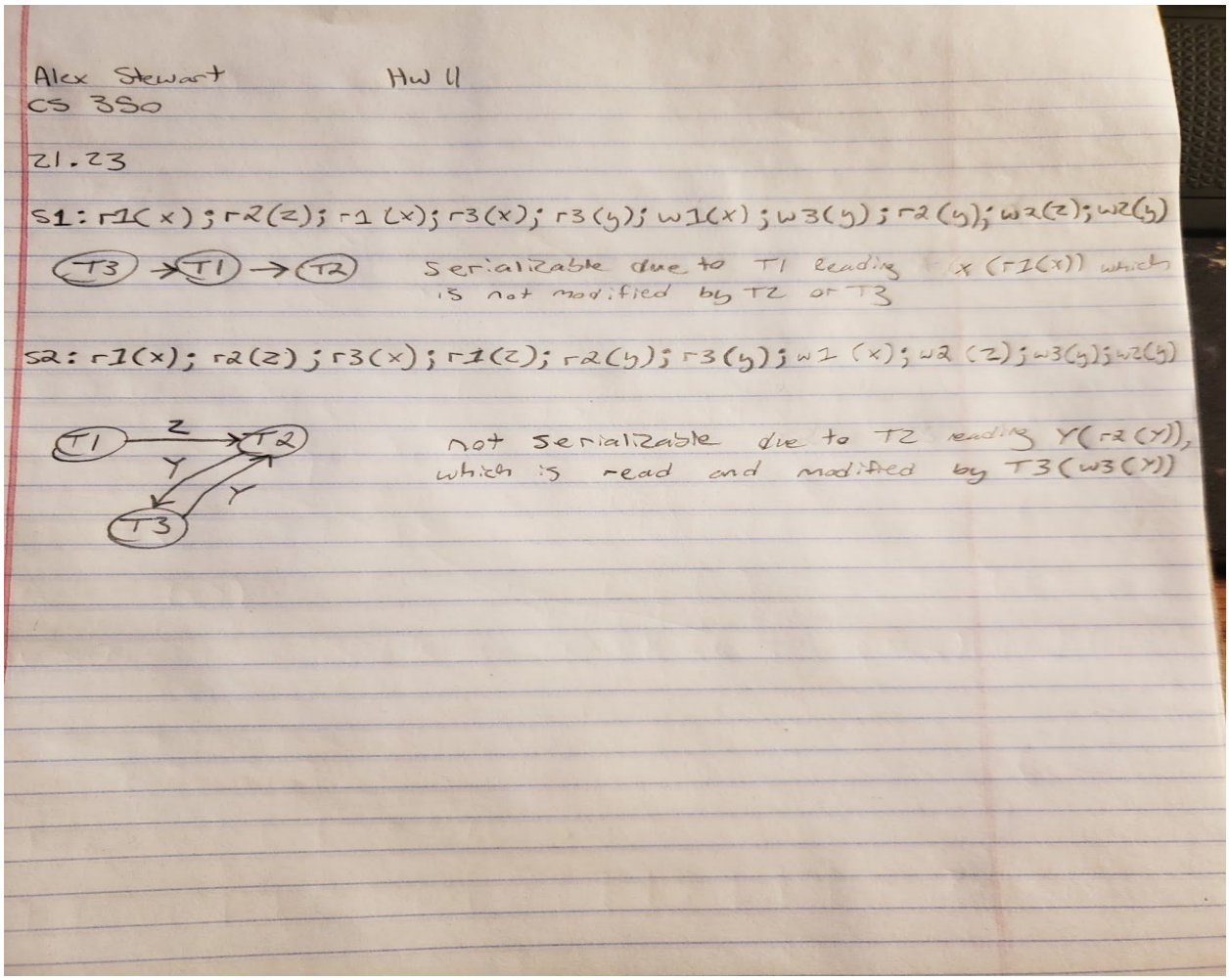
Alex Stewart

CS350 HW11

Transactions, Concurrency Control, Recovery (DUE Week 11 - Friday night at midnight)

21.23



22.25

Apply the timestamp ordering algorithm to schedules in Figure 21.8(b) and (c), and determine whether the algorithm will allow the execution of the schedules. Assume a clock has linear time points 0,1,2,3.... and the original read and write stamps of all items are 0.

read_TS(X) = read_TS(Y) = read_TS(Z) = 0

write_TS(X) = write_TS(Y) = write_TS(Z) = 0

Schedules in Figure 21.8(b) Schedule E or SE and Figure 21.8(c) Schedule F or SF.

SE = r2(Z); r2(Y); w2(Y); r3(Z); r1(X); w1(X); w3(Y); w3(Z); r2(X); r1(Y); w1(Y); w2(X);

1 2 3 4 5 6 7 8 9 10 11 12 13

SF = r3(Y); r3(Z); r1(X); w1(X); w3(Y); w3(Z); r2(Z); r1(Y); w1(Y); r2(Y); w2(Y); r2(X);

1 2 3 4 5 6 7 8 9 10 11 12 13

Assume that each operation takes one time unit. The numbers under the operations indicated the time when each operation has occurred. We should also assume that each transaction timestamp corresponds to the time of its first operations in each schedule. The transaction timestamps are as follows

values don't change

TS(T1) = 6 TS(T1) = 3

TS(T2) = 1 TS(T2) = 7

TS(T3) = 4 TS(T3) = 1

(a) Applying timestamp ordering to Schedule E

Initial values =

read_TS(X)=0,read_TS(Y)=0,read_TS(Z)=0,write_TS(X)=0,write_TS(Y)=0,write_TS(Z)=0

TS(T1)=6, TS(T2)=1, TS(T3)=4 **no change**

T2 = read_item(Z)

TS(T2) > write_TS(Z)

Execute read_item(Z)

Set read_TS(Z) <- max(read_TS(Z),TS(T2)) = 1

read_TS(X)=0,read_TS(Y)=0,read_TS(Z)=1,write_TS(X)=0,write_TS(Y)=0,write_TS(Z)=0

T2 read_item(Y)

TS(T2) > write_TS(Y)

Execute read_item(Y)

Set read_TS(Y) <- max(read_TS(Y),TS(T2)) = 1

read_TS(X)=0,read_TS(Y)=1,read_TS(Z)=1,write_TS(X)=0,write_TS(Y)=0,write_TS(Z)=

0

T2: write_item(Y)

TS(T2) = read_TS(Y) and TS(T2) > write_TS(Y)

Execute write_item(Y)

write_TS(Y) <- max(write_TS(Y),TS(T2)) = 1

read_TS(X)=0,read_TS(Y)=1,read_TS(Z)=1,write_TS(X)=0,write_TS(Y)=1,write_TS(Z)=

0

T3: read_item(Y)

TS(T3) > write_TS(Y)

Execute read_item(Y)

read_TS(Y) <- max(read_TS(Y),TS(T3)) = 4

read_TS(X)=0,read_TS(Y)=4,read_TS(Z)=1,write_TS(X)=0,write_TS(Y)=1,write_TS(Z)=

0

T3: read_item(Z)

TS(T3) > write_TS(Z)

Execute read_item(Z)

read_TS(Z) <- max(read_TS(Z),TS(T3)) = 4

read_TS(X)=0,read_TS(Y)=4,read_TS(Z)=1,write_TS(X)=0,write_TS(Y)=1,write_TS(Z)=

0

T1: read_item(X)

TS(T1) > write_TS(X)

Execute read_item(X)

read_TS(X) <- max(read_TS(X),TS(T1)) = 6

read_TS(X)=6,read_TS(Y)=4,read_TS(Z)=1,write_TS(X)=0,write_TS(Y)=1,write_TS(Z)=

0

T1: write_item(X)

TS(T1) = read_TS(X) and TS(T1) > write_TS(X)

Execute write_item(X)

write_TS(X) <- max(write_TS(X),TS(T1)) = 6

read_TS(X)=6,read_TS(Y)=4,read_TS(Z)=1,write_TS(X)=6,write_TS(Y)=1,write_TS(Z)=

0

T3: write_item(Y)

TS(T3) = read_TS(Y) and TS(T3) > write_TS(Y)

Execute write_item(Y)

write_TS(Y) <- max(write_TS(Y),TS(T3)) = 4

read_TS(X)=6,read_TS(Y)=4,read_TS(Z)=1,write_TS(X)=6,write_TS(Y)=4,write_TS(Z)=

0

T3: write_item(Z)

TS(T3) > read_TS(Z) and TS(T3) > write_TS(Z)

Execute write_item(Z)

write_TS(Z) <- max(write_TS(Z),TS(T3)) = 4

read_TS(X)=6,read_TS(Y)=4,read_TS(Z)=1,write_TS(X)=6,write_TS(Y)=4,write_TS(Z)=4

T2: read_item(X)

TS(T2) < write_TS(X)

Abort and Rollback Y2, Reject read_item(X)

Result

Only T1 would finish because of cascading rollback in T2 and T3

(b) Applying timestamp ordering to Schedule F

Initial values (new values are shown after each operation):

read_TS(X)=0,read_TS(Y)=0,read_TS(Z)=0,write_TS(X)=0,write_TS(Y)=0,write_TS(Z)=0

TS(T1)=3, TS(T2)=7, TS(T3)=1 (These do not change)

T3: read_item(Y)

TS(T3) > write_TS(Y)

Execute read_item(Y)

Set read_TS(Y) <- max(read_TS(Y),TS(T3)) = 1

read_TS(X)=0,read_TS(Y)=1,read_TS(Z)=0,write_TS(X)=0,write_TS(Y)=0,write_TS(Z)=0

T3: read_item(Z)

TS(T3) > write_TS(Z)

Execute read_item(Z)

Set read_TS(Z) <- max(read_TS(Z),TS(T3)) = 1

read_TS(X)=0,read_TS(Y)=1,read_TS(Z)=1,write_TS(X)=0,write_TS(Y)=0,write_TS(Z)=

0

T1: read_item(X)

TS(T1) > write_TS(X)

Execute read_item(X)

read_TS(X) <- max(read_TS(X),TS(T1)) = 3

read_TS(X)=3,read_TS(Y)=1,read_TS(Z)=1,write_TS(X)=0,write_TS(Y)=0,write_TS(Z)=

0

T1: write_item(X)

TS(T1) = read_TS(X) and TS(T1) > write_TS(X)

Execute write_item(X)

write_TS(X) <- max(write_TS(X),TS(T1)) = 3

read_TS(X)=3,read_TS(Y)=1,read_TS(Z)=1,write_TS(X)=3,write_TS(Y)=0,write_TS(Z)=

0

T3: write_item(Y)

TS(T3) = read_TS(Y) and TS(T3) > write_TS(Y)

Execute write_item(Y)

write_TS(Y) <- max(write_TS(Y),TS(T3)) = 1

read_TS(X)=3,read_TS(Y)=1,read_TS(Z)=1,write_TS(X)=3,write_TS(Y)=1,write_TS(Z)=

0

T3: write_item(Z)

TS(T3) = read_TS(Z) and TS(T3) > write_TS(Z)

Execute write_item(Z)

write_TS(Z) <- max(write_TS(Z),TS(T3)) = 1

read_TS(X)=3,read_TS(Y)=1,read_TS(Z)=1,write_TS(X)=3,write_TS(Y)=1,write_TS(Z)=

1

T2: read_item(Z)

TS(T2) > write_TS(Z)

Execute read_item(Z)

Set read_TS(Z) <- max(read_TS(Z),TS(T2)) = 7

read_TS(X)=3,read_TS(Y)=1,read_TS(Z)=7,write_TS(X)=3,write_TS(Y)=1,write_TS(Z)=

1

T1: read_item(Y)

TS(T1) > write_TS(Y)

Execute read_item(Y)

Set read_TS(Y) <- max(read_TS(Y),TS(T1)) = 3

read_TS(X)=3,read_TS(Y)=3,read_TS(Z)=7,write_TS(X)=3,write_TS(Y)=1,write_TS(Z)=

1

T1: write_item(Y)

TS(T1) = read_TS(Y) and TS(T1) > write_TS(Y)

Execute write_item(Y)

write_TS(Y) <- max(read_TS(Y),TS(T1)) = 3

read_TS(X)=3,read_TS(Y)=3,read_TS(Z)=7,write_TS(X)=3,write_TS(Y)=3,write_TS(Z)=

1

T2: read_item(Y)

TS(T2) > write_TS(Y)

Execute read_item(Y)

Set read_TS(Y) <- max(read_TS(Y),TS(T2)) = 7

read_TS(X)=3,read_TS(Y)=7,read_TS(Z)=7,write_TS(X)=3,write_TS(Y)=3,write_TS(Z)=

1

T2: write_item(Y)

TS(T2) = read_TS(Y) and TS(T2) > write_TS(Y)

Execute write_item(Y)

write_TS(Y) <- max(write_TS(Y),TS(T2)) = 7

read_TS(X)=3,read_TS(Y)=7,read_TS(Z)=7,write_TS(X)=3,write_TS(Y)=7,write_TS(Z)=

1

T2: read_item(X)

TS(T2) > write_TS(X)

Execute read_item(X)

Set read_TS(X) <- max(read_TS(X),TS(T2)) = 7

read_TS(X)=7,read_TS(Y)=7,read_TS(Z)=7,write_TS(X)=3,write_TS(Y)=3,write_TS(Z)=

1

T2: write_item(X)

TS(T2) = read_TS(X) and TS(T2) > write_TS(X)

Execute write_item(X)

write_TS(X) <- max(write_TS(X),TS(T2)) = 7

read_TS(X)=7,read_TS(Y)=7,read_TS(Z)=7,write_TS(X)=7,write_TS(Y)=7,write_TS(Z)=1

Result

F will execute