COMP 3311
DATABASE MANAGEMENT SYSTEMS

LECTURE 23 EXERCISES
RECOVERY SYSTEM

EXERCISE 1

Consider the log corresponding to a schedule of three *serially executing* transactions T_1, T_2 and T_3 where a system failure occurs at the point indicated. Assume that we use the *immediate update protocol*.

(a) Which transactions are undone?

Undo: T_3

(b) Which transactions are redone?

Redo: T_1 T_2

Log File
<T_1 start>
<T_1, B, 15, 12>
<T_1 commit>
<T_2 start>
<T_2, B, 12, 18>
<T_2, D, 25, 26>
<T_2 commit>
<T_3 start>
<T_3, D, 20, 25>

← **system failure**

Scan the log and
undo **if** <T_i start> **but no** <T_i commit>
redo **if** <T_i start> **and** <T_i commit>

EXERCISE 2

Consider the log corresponding to a schedule of three *concurrently executing* transactions T_1, T_2 and T_3 where a system crash occurs at the point indicated. Assume that we use the *immediate update protocol* with checkpointing.

(a) Which transactions are undone?

Undo: T_2

(b) Which transactions are redone?

Redo: T_3 T_1

Log File	
<T_1 start>	
<T_1, A, 2, 5>	
<T_1, B, 1, 2>	
<T_2 start>	
<T_1, C, 2, 8>	
<T_2, B, 2, 5>	
<checkpoint, {T_1, T_2}>	← add T_2 to undo-list
<T_1 commit>	← add to redo-list
<T_2, B, 5, 6>	
<T_3 start>	√ on redo-list
<T_3, A, 5, 3>	
<T_3 commit>	← add to redo-list
<T_2, C, 8, 2>	
<T_2, A, 3, 5>	
	← **system failure**

```
Scan the log backward and
if find <$T_i$ commit> add to redo-list
if find <$T_i$ start> add to undo-list if not in redo-list
for every $T_i$ in <checkpoint, {$T_i$, …}>
    if $T_i$ not in redo-list add to undo-list
```

EXERCISE 3

Consider the log corresponding to a schedule of four *serially executing* transactions T_1, T_2, T_3 and T_4 where a system crash occurs at the point indicated. Assume that we use the *immediate update protocol* with checkpointing.

(a) Which transactions are undone?

Undo: T_4

(b) Which transactions are redone?

Redo: T_1 T_2 T_3

Log File
<T_1 start>
<T_1, A, 1, 2>
<T_1, B, 5, 12>
<checkpoint>
<T_1 commit>
<T_2 start>
<T_2, B, 12, 18>
<T_2 commit>
<T_3 start>
<T_3, D, 25, 26>
<T_3 commit>
<T_4 start>
<T_4, A, 20, 25>

← **system failure**

Scan log backward to first <$T_i$start> *before* the checkpoint.
Scan log forward and
undo **if** <T_i start> **but no** <T_i commit>
redo **if** <T_i start> **and** <T_i commit>

EXERCISE 4

Consider the log corresponding to a schedule of three *concurrently executing* transactions T_1, T_2 and T_3 where a system crash occurs at the point indicated. Assume that we use the *immediate update protocol* with checkpointing.

(a) Which transactions are undone?

Undo: T_1 T_3

(b) Which transactions are redone?

Redo: T_2

Log File	
<T_3 start>	
<T_3, B, 15, 12>	
<T_2 start>	
<T_2, B, 12, 18>	
<checkpoint {T_2, T_3}>	← add T_3 to undo-list
<T_2 commit>	← add to redo-list
<T_1 start>	← add to redo-list
<T_1, D, 20, 25>	
<T_1, D, 25, 26>	
	← **system failure**

```
Scan the log backward and
if find <$T_i$ commit> add to redo-list
if find <$T_i$ start> add to undo-list if not in redo-list
for every $T_i$ in <checkpoint, {$T_i$, …}>
    if $T_i$ not in redo-list add to undo-list
```