Problem 1

Assume the following transactions: $T_1, T_2$ and $T_3$:

<table>
<thead>
<tr>
<th>Schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td>time</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
</tbody>
</table>

Assume that the transactions begin at the following timestamps,

$$T_1 : 2, T_2 : 4, T_3 : 1.$$ 

All transactions will commit at the same time. Will any transactions in this schedule abort if we are using the time-stamp protocol? Which one(s)?

Problem 2

Consider a database with the following initial values, and the attached command log:

$A = 50, B = 48, C = 0, D = 47$

LOG:

$< T_0, \text{start} >$
$< T_0, A, 50, 75 >$
$< T_1, \text{start} >$
$< T_1, B, 48, 92 >$
$< T_2, \text{start} >$
$< T_2, C, 0, 33 >$
$< T_1, B, 92, 108 >$
$< \text{checkpoint} : T_0, T_1, T_2 >$
$< T_3, \text{start} >$
$< T_0, A, 75, 100 >$
$< T_2, \text{commit} >$
$< T_3, D, 47, 52 >$
$< T_3, \text{commit} >$

Assume that the system crashes before the remaining transactions can commit. Use the recovery protocol for concurrent transactions (which persists all in-memory dirty pages and transaction log entries at each checkpoint) to answer the following questions.
1. List any transactions that will need to be undone or redone in the recovery process.

2. List, in order, the set of logged operations to be performed to undo or redo the transactions. (i.e. "Set $A$ to 7", "Set $B$ to 39", etc.)

3. Give the final values for $A$, $B$, $C$, and $D$. 