Transaction Processing
Exercise 6

Assignment 1: Explain the multiversion concurrency control (MVCC) protocol based on the following example:

<table>
<thead>
<tr>
<th>T1</th>
<th>T2</th>
</tr>
</thead>
<tbody>
<tr>
<td>r₁(x)</td>
<td>r₂(x)</td>
</tr>
<tr>
<td>w₁(x)</td>
<td>w₂(y)</td>
</tr>
<tr>
<td>r₁(y)</td>
<td></td>
</tr>
<tr>
<td>w₁(z)</td>
<td></td>
</tr>
<tr>
<td>commit;</td>
<td>commit;</td>
</tr>
</tbody>
</table>

Which problem does the MVCC protocol not solve?

Assignment 2: Given is the following schedule:

T₁
---
r(x) w(x)
---

T₂
---
r(x)
---

T₃
---
r(x)
---

T₄
---
r(x) w(x)
---

T₅
---
r(x)
---

T₆
---
r(x) w(x)
---

T₇
---
r(x)
---

Determine for each transaction which version of x is read and when the different versions can be released.

Assignment 3: Which problems occur in the following schedule?

- How can these problems be avoided?
- Which further isolation levels do you know?
- Explain which problems of multi-user operation can be avoided with the discussed isolation levels.

<table>
<thead>
<tr>
<th>T1</th>
<th>T2</th>
</tr>
</thead>
<tbody>
<tr>
<td>r(K)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>r(K)</td>
</tr>
<tr>
<td>K:=K-1</td>
<td>w(K)</td>
</tr>
<tr>
<td>B:=K-0,5</td>
<td>w(B)</td>
</tr>
</tbody>
</table>

Assignment 4: Explain the concept of serializibility checking based on permutations based on the following schedules $s_1$ bis $s_6$:

- $s_1 := w_2(x)w_2(y)r_1(x)r_1(y)w_1(y)r_2(y)$
- $s_2 := r_1(x)w_1(x)r_2(y)w_2(y)r_1(y)r_2(x)$
- $s_3 := w_1(y)w_2(y)r_2(y)r_1(x)w_3(z)$
- $s_4 := r_1(x)w_1(x)r_2(y)r_3(y)w_2(x)w_3(x)$
- $s_5 := w_2(x)w_1(x)w_1(y)w_2(y)w_1(y)w_3(z)$
- $s_6 := w_2(x)w_2(y)r_1(x)r_1(y)w_1(y)w_3(z)$

Which of the schedules are serializable.

Exercise 6 2/2 Good Luck!