# Solutions to Exercise 1

**Problem 1.**
**Part 1.a.**    Regular, not atomic.

**Part 1.b.**    None of the above.

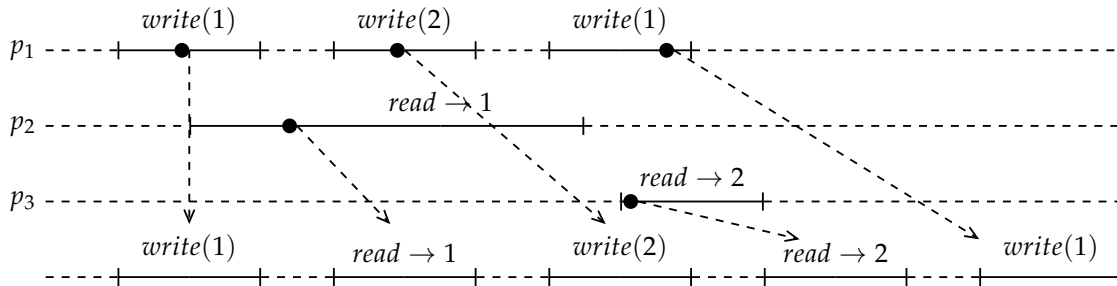**Part 1.c.**
  Atomic.



**Figure 1:** Serialization points and an equivalent sequential execution.

**Problem 2.**    Consider the transformation from (binary) SRSW safe to (binary) MRSW safe registers given in class.

**Part 2.a.**    Prove that the transformation works for multi-valued registers and regular registers.
Please see Chapter 4 of the Notes (2014) on the website, page 54.

**Part 2.b.**    Also, prove that the transformation does not work for atomic registers (by providing a counterexample that breaks atomicity).
Please see Chapter 4 of the Notes (2014) on the website, page 55.

**Problem 3.**    Consider the transformation from binary MRSW safe registers to binary MRSW regular registers, given in class.

**Part 3.a.**    Prove that the transformation does **not** generate multi-valued MRSW regular registers (by providing a counterexample that breaks regularity).
If the registers are multi-valued, then two consecutive reads on the safe register *Reg* may return arbitrary values, breaking regularity of the register implementation. Since the safe register is binary in the correct implementation (and thus limited to two values), this does not occur in the transformation given in class.

**Part 3.b.**    Also, prove that the resulting registers are not binary atomic (by providing a counterexample that breaks atomicity).

The counterexample can be easily built by scheduling two distinct reads during a write(1) operation on the register. Since the underlying register is safe, we can ensure that the first operation returns 1, while the second (non-overlapping) operation returns 0, contradicting atomicity.