Exercise 1. ***Consistency Models***

(a) The history is sequentially consistent. q.enq(y) can happen first in thread B, then q.end(x) in Thread A. Finally, Thread A will dequeue the value y, because it has been enqueued first (in Thread B).

(b) The history is not linearizable. In any case, q.enq(y) happens after q.enq(x). This implies that x has to be dequeued first.

(c) Yes, all linearizable histories are sequentially consistent. Assume we have a linearizable history with some linearization point. Without adjusting anything, this history must also be sequentially consistent.

(d)
- Both p and q viewed in isolation are sequentially consistent as can be seen by either moving thread A or B.
- The history as a whole is not sequentially consistent and can thus also not be linearizable.
- The history is not sequentially consistent, because there are circular dependencies when trying to reorder the items.