Lab 6 – Exercises

Are the following histories linearizable or sequentially consistent? Explain your answers and write the equivalent linearizable/sequential consistent histories where applicable.

1. **Read/write register**
    a. Concurrent threads A, B, C, register r.

```
A: r.write(1)
C: r.read()
A: r:void
A: r.write(2)
C: r:2
C: r.read()
B: r.read()
A: r:void
C: r:1
A: r.write(1)
B: r:1
A: r:void
```

    b. Concurrent threads A, B, C, register r.

```
A: r.write(1)
B: r.read()
A: r:void
A: r.write(2)
A: r:void
A: r.write(1)
B: r:1
C: r.read()
A: r:void
C: r:2
```

2. **Stack:** we have the following operations: `push(x)` pushes element x on the stack, returns void; `pop()` retrieves an element from the stack; `empty()` returns `true` if stack is empty and `false` otherwise.
    a. Concurrent threads A, B and C, stack s.

```
C: s.empty()
A: s.push(10)
B: s.pop()
A: s:void
A: s.push(20)
B: s:10
A: s:void
C: s:true
```

b. Concurrent threads A and B, stack s.

```
A: s.push(10)
B: s.push(10)
A: s:void
A: s.pop()
B: s:void
B: s.empty()
A: s:10
B: s:true
A: s.pop()
A: s:10
```

3. **Queue:** we have the following operations: `enq(x)` inserts and element into the queue, returns void; `deq()` retrieves an element from the queue. We have three concurrent threads A, B, C and a queue q.

```
A: q.enq(x)
B: q.enq(y)
A: q:void
B: q:void
A: q.deq()
C: q.deq()
A: q:y
C: q:y
```