Immediate Snapshot Algorithm

(a) Prove that, in the Algorithm of Slide 24, if at most x processes invoke REC_WRITE-SNAPSHOT$(x, -)$, then at most $x-1$ invoke REC_WRITE-SNAPSHOT$(x-1,\text{-})$ at Line 12 and at least one returns a view at Line 10.

(b) Prove that the Algorithm of Slide 24 fulfills the termination, self-inclusion, containment and immediacy properties.

(c) Compute the complexity of the Algorithm of Slide 24 in terms of number of shared memory accesses.

Consensus Algorithm for 2 Processes

Suppose that 2 processes run the following algorithm:

```
function PROPOSE(v)
    est ← v
    r ← 1
    while true do
        view ← IS[r].WRITE-SNAPSHOT(s)
        if |view| = 2 then
            est ← v_j such that (j, v_j) ∈ view and j ≠ id
            r ← r + 1
        else
            return est
```

(a) Show that this algorithm doesn't solve consensus.

(b) Suppose that in any execution, there is a round such that the invocations of WRITE-SNAPSHOT by the two processes are not set-linearized together. Prove that this algorithm then solves consensus between the two processes.

(c) Draw the subdivided segment representing the possible states of the system after a few rounds. Then, tag the possible state of the processes with their decision values.

k-Set Agreement with Less than k crashes

Find and prove an algorithm solving the k-set agreement among $n > k$ processes in presence of at most $k - 1$ crashes.