

Exercise 10

Shared Memory

Problem 1

Explain why every process needs to maintain a copy of the register value in the “Majority Voting” algorithm (Either Algorithm 4.2 in the book, or ABD95 algorithm from slides).

Solution

The algorithm also needs to maintain a copy of the register value at all processes, even if we assume only one reader. Assume that some process q does not maintain a copy. Assume, furthermore, that the writer updates the value of the register: it can do so only by accessing a majority of the processes. If q is in that majority, then the writer would have stored the value in a majority of the processes minus one. It might happen that all processes in that majority, except for q , crash. But the set of remaining processes plus q also constitute a majority. A subsequent read in this majority might not return the last value written.

Problem 2

Consider a system with two processes P_1 and P_2 . Give a register execution such that each process performs at most two operations and the execution is

1. not safe
2. not regular but safe
3. not atomic but regular

Solution

1. not safe

```
P1 [ W(1) ]
P2 [ R() -> 0]
```

2. not regular but safe

```
P1 [ W(1) ] [ W(2) ]
P2 [ R() -> 0]
```

3. not atomic but regular

```
P1 [ W(1) ] [ W(2) ]
P2 [R() -> 2] [ R() -> 1]
```