

Homework Exercise #8
Due: 11:30 a.m., November 19, 2014

1. Consider an asynchronous shared-memory system in which any number of processes can experience crash failures.

Define a `BLOCKVECTOR` object type which stores a vector of m components. Each component stores a natural number. The `BLOCKVECTOR` object provides two operations:

- `BLOCKWRITE`(i, j, v), where $1 \leq i \leq j \leq m$ and v is a natural number, changes all of the components $i, i + 1, \dots, j$ of the vector to v and returns `ACK`, and
- `SCAN` returns the contents of the entire vector without changing the object's state.

You may assume that $m \geq 10$.

- (a) Suppose you have a wait-free consensus algorithm that uses `BLOCKVECTOR` objects. Suppose C is a multivalent configuration such that every successor of C is univalent. Prove that each process's next step after C must update a component that is not updated by any other process's next step.
- (b) Determine the largest number k such that wait-free consensus can be solved in a system of k processes using `BLOCKVECTOR` objects. (Your value of k may or may not depend on m .) Prove your answer is correct.

Hint: in designing a consensus algorithm, it is sometimes helpful to divide the processes into two groups, so that processes can first agree on a value within each group, and then decide which of the two groups' chosen values will be the common output of all processes.