

## COMPUTER SCIENCE TRIPOS Part IB – 2022 – Paper 5

### 4 Concurrent and Distributed Systems (djg11)

- (a) A shared resource is updated by one writer at a time but must support concurrent readers. One or more locks are used to manage exclusion.
- (i) What design choices exist regarding fairness? [2 marks]
  - (ii) Compare having two simple locks, respectively used for read and write exclusion, with having a single, specialised locking primitive. What additional state transition(s) could the specialised lock support? [4 marks]
  - (iii) If deadlock is to be avoided using a locking order, is a complete or partial order needed? [2 marks]
  - (iv) How should the various transitions possible with the specialised lock mentioned above be integrated into a lock ordering policy? [3 marks]
- (b) A mutex is acquired at the start of the body of a function.
- (i) What can go wrong if the function is recursive? [1 mark]
  - (ii) To avoid the problem, a friend suggests that threading systems should allow lock acquisitions to proceed if the lock is already held by the same piece of code. Could something like this work and is it worthwhile? [3 marks]
- (c) The behaviours provided by a shared, global variable are to be emulated within a message-passing system.
- (i) Give pseudocode that provides the emulation. [2 marks]
  - (ii) Discuss whether emulation of locking primitives, such as semaphores, is possible or would be worthwhile? [3 marks]