

Two hours

**UNIVERSITY OF MANCHESTER  
SCHOOL OF COMPUTER SCIENCE**

Distributed Computing

Date: Thursday 17th May 2012

Time: 14:00 - 16:00

---

**Please answer Question ONE and any two from Questions TWO, THREE and FOUR**

---

This is a CLOSED book examination

The use of electronic calculators is NOT permitted

**[PTO]**

1. **Compulsory**

- a) What is a Java *servlet*? (2 marks)
- b) What is the main assumption on which *Cristian's clock synchronisation algorithm* is based? (2 marks)
- c) Explain the difference between a *name server* and a *directory server*. (2 marks)
- d) Explain briefly what is meant by the term *middleware*. (2 marks)
- e) Why is it practically impossible to achieve *strict consistency* in a distributed system? (2 marks)
- f) Traditional RPC mechanisms cannot handle pointers. What is the problem? (2 marks)
- g) What is meant by *parameter unmarshalling*? (2 marks)
- h) What is the key difference between *caching* and *replication*? (2 marks)
- i) Explain briefly why somebody would like to use *Cloud Computing*. (2 marks)
- j) In the context of lab exercise 2, what would you do to launch a *denial of service* attack against the server? (2 marks)

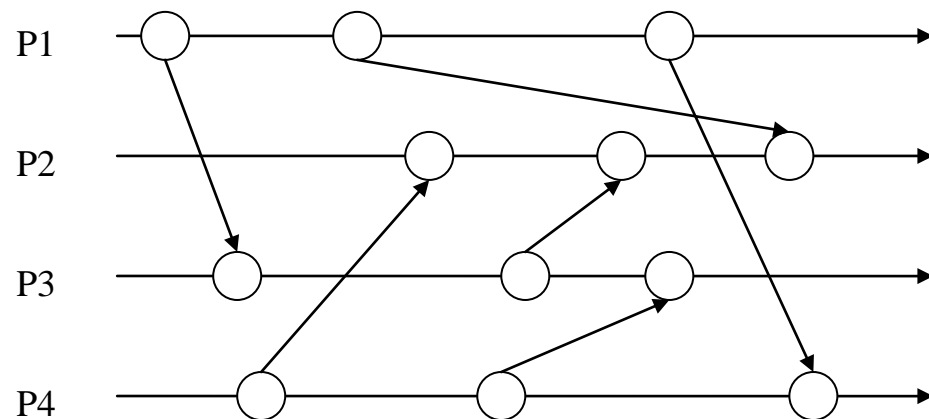
2. a) Explain briefly what is wrong with the assumption “latency is zero” in the context of distributed computing. Why is it considered a common fallacy? (3 marks)
- b) Explain briefly what the four properties commonly denoted by the acronym ACID are when referring to transactions. (3 marks)
- c) Outline the Byzantine Generals problem, and illustrate how one of three being a traitor makes a solution impossible, whereas with one of four it is achievable. (6 marks)
- d) The following four processes access a shared variable  $x$ . Each process accesses a different replica of the store used to hold this variable. Before any process starts executing, the value of  $x$  is 0 in all the replicas.

<u>Process 1</u>	<u>Process 2</u>	<u>Process 3</u>	<u>Process 4</u>
$x=1;$	$x=2;$	$y=0;$	$z=2;$
		$if(x==1)$	$if(x==2)$
		$y=y+1;$	$z=z+2;$
		$if(x==2)$	$if(x==1)$
		$y=y+2;$	$z=z+1;$

- (i) When all four processes have completed executing the statements given, are 3 and 5 possible values of  $y$  and  $z$  respectively, if the replication uses the sequential consistency model? Justify your answer. (4 marks)
- (ii) When all four processes have completed executing the statements given, are 3 and 5 possible values of  $y$  and  $z$  respectively, if the replication uses the causal consistency model? Justify your answer. (4 marks)

[PTO]

3. a) Explain briefly what is the role of a client stub and a server stub in RPC. (3 marks)
- b) Describe in detail how the two-phase commit protocol can implement distributed transactions. (5 marks)
- c) Consider the figure below, which shows 4 processes and a number of communication events taking place over a period of time.



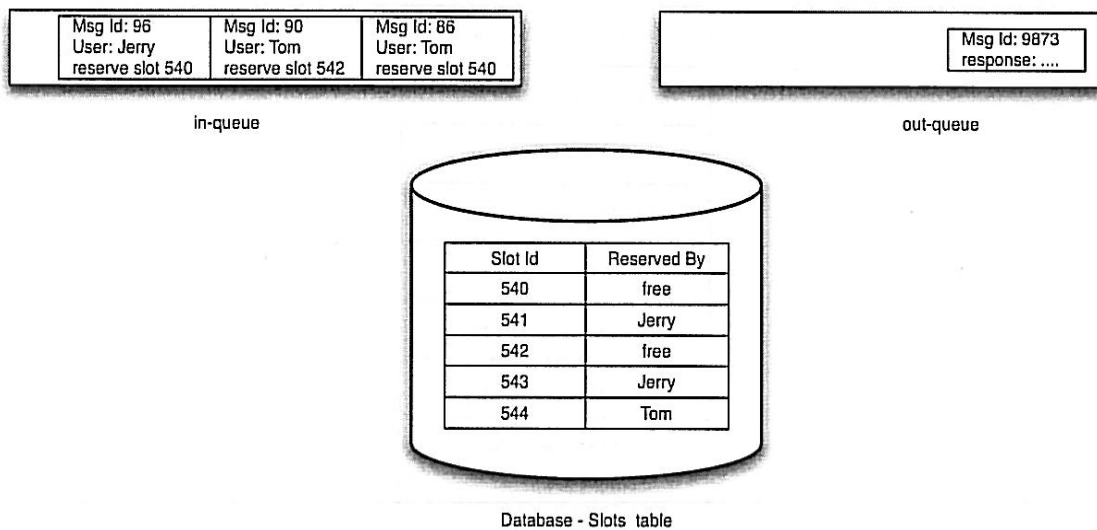
Calculate the value of Lamport clocks and vector clocks for each of the 12 events shown above. You can assume that all logical clocks start initially with zeros. (6 marks)

- d) In a system containing 7 computers, identified by the integers 1-7, the coordinator is chosen by the Bully algorithm to be the live one with the highest identifier. Assume for this part that all messages are delivered promptly, and that the computers and the network are entirely reliable.

At a certain point in time the coordinator (computer 7) and the computer with the second-highest identifier (computer 6) crash. How many messages in total are sent if the computer with identifier 1 is the computer discovering the crash and triggering an election? You need to count all three types of messages that the algorithm sends. You can assume that computers with identifiers 6 and 7 remain crashed during the election.

(6 marks)

4. This question is based on your understanding of transactions and distributed transactions, in the context of lab exercise 2. The server that processes all requests (reservation, cancellation, availability and bookings retrieval) hosts a database for storing reservations and some message-oriented middleware, which provides an in-queue (to queue requests) and an out-queue (to queue responses to messages). Suppose there are three messages in the in-queue: two from Tom and one from Jerry. Suppose also that Tom already holds a reservation for slot 544 and Jerry holds a reservation for slots 541 and 543 (assume a total of 1000 slots). The state of the database and queues is shown below.



The pseudo-code that processes a reservation request is shown below. It assumes that all messages are delivered exactly once by the underlying messaging infrastructure. Also, assume that the maximum permitted number of reservations a student can hold at any one time is two, that is, `max_res=2`.

```
begin_tx;    // this is an ACID transaction to reserve a slot
            dequeue a request from the in-queue

    if requested slot is 'free' and
        number of reservations by student < max_res then
        reserve slot for the student;
        send 'slot reserved' response message;
    else if number of reservations by student >= max_res then
        send 'fail: too many reservations' response message;
    else if slot is not 'free' then
        send 'fail: slot is not free' response message;
    endif
commit;      // transaction commits
```

(Question 4 continues on the following page)

(Question 4 continues from the previous page)

- a) Show what the state of the slots table in the database would look like after all in-queue messages are executed (assuming that messages are executed left-to-right, that is, the first message executed is the request from Jerry). Your answer should indicate clearly the response messages generated in each case. (4 marks)
- b) Show via an execution sequence (an order in which statements are executed), how Tom can hold more than the maximum permitted number of reservations if two threads are concurrently processing messages from the in-queue and if the code is not enclosed in an ACID transaction (4 marks)
- c) After which statements in the pseudo-code for processing a reservation request can the server crash? Describe the state of the database, in-queue, and out-queue after the crash and recovery. (4 marks)
- d) Suppose that four threads, running on different CPUs, are concurrently processing messages from the in-queue. Every incoming message requires processing time of 40 milliseconds and, on average, there are about 10000 messages per hour sent to the server. How long do you expect to have messages waiting in the in-queue before their processing starts? (4 marks)
- e) Propose a strategy that the server may use to adapt on-the-fly the number of threads so that the length of the in-queue is minimal and threads are kept as busy as possible. State any assumptions you make. (4 marks)

**END OF EXAMINATION**