UNIVERSITY OF MORATUWA

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.Sc. Engineering
2009 Intake Semester 8 Examination

CS4262 DISTRIBUTED SYSTEMS

Time allowed: 2 Hours February 2014

ADDITIONAL MATERIAL: *None*

INSTRUCTIONS TO CANDIDATES:

1. This paper consists of **5** questions in **4** pages.
2. Answer any **4** questions.
3. Start answering each of the main questions on a new page.
4. The maximum attainable mark for each question is given in brackets.
5. This examination accounts for 50% of the module assessment.
6. This is a closed book examination.

   NB: It is an offence to be in possession of unauthorised material during the examination.
7. Only calculators approved by the Faculty of Engineering are permitted.
8. Assume reasonable values for any data not given in or with the examination paper. Clearly state such assumptions made on the script.
9. In case of any doubt as to the interpretation of the wording of a question, make suitable assumptions and clearly state them on the script.
10. This paper should be answered only in English.

Question 1 (25 marks)

(i) Explain why it might be a problem to detect failures in a distributed system. [2]

(ii) What is "transparency" in the context of Distributed systems? Briefly describe 3 forms of transparency. [8]

(iii) Twitter has become the social media solution for rapidly disseminating important news items during social gatherings, disasters, and political events. Twitter users are able to read news that others have never heard about. However, Twitter being owned by a single company creates many issues. For example, a government can demand Twitter to give out all Tweets related to a given event, drop selected Tweets, or they can even block access to the Twitter.com. Privacy concerned users are also worried about all their Tweets being accumulated by a single entity.

A team of volunteers is starting a project to address these issues in Twitter by developing a distributed system. They are envisioning a system where there is no single controlling entity and users will contribute to keep the system running, e.g., by contributing processing and storage resources as in peer-to-peer file sharing systems. While user-to-user Tweets are important, the volunteers have decided to first support only the public Tweets as they are the most important in disseminating news.

a) What type of an overlay network would you recommend for this solution? Explain with justification. [5]

b) Propose a suitable solution, with justification, to disseminate the Tweets among users/nodes while addressing the above mentioned concerns. [4]

c) Discuss about the scalability limitations of your proposed solution. [3]

d) "Explain failure transparency" in the context of your proposed solution. [3]

Question 2 (25 marks)

(i) Compare and contrast (i.e., identify the similarities and dissimilarities) Remote Procedure Calls (RPCs), Remote Method Invocation (RMI), and Web Services. [6]

(ii) Using a suitable diagram explain how a Remote Procedure Call (RPC) is executed. [6]

(iii) If a client and a server are placed far apart, we may see the network latency dominating the overall performance. How can we tackle this problem? [5]

(iv) Churn (number of nodes moving in and out of a system) is a norm in distributed systems such as volunteer computing, desktop grids, and peer-to-peer systems. Propose a solution that each node can use to estimate the number of nodes in such a system at any given time. While your estimate does not need to be exact, it needs to be reasonably accurate. [8]

Question 3 (25 marks)

(i) Using a suitable example explain the "event-based architecture". [4]

(ii) Nodes in a distributed system are responsible for receiving events from external sources (e.g., sensor readings, Twitter messages, or credit card transactions). Once an event is received, the receiving node processes the event to identify whether it is an important/relevant event (e.g., a Twitter message with an important hashtag). If so, it will send out a notification to all the other nodes. Given the high arrival rate of events, it is important to concurrently handle event reception, processing, and notification.

Which of the following options will you recommend to address this problem? Justify your selection.

a) Persistent vs. transient communication. [3]

b) Asynchronous vs. synchronous communication. [3]

c) Using a suitable diagram illustrate your solution based on your recommendations for questions (a) and (b). [4]

d) Message queues at the sending node vs. message queues at the receiving nodes. [3]

e) Network-level multicasting vs. application-level multicasting. [3]

(iii) What is CAP (Consistency, Availability, and Partition) theorem? Explain. [5]

Question 4 (25 marks)

(i) What are "task queues" and "work stealing" in the context of load balancing? Explain. [4]

(ii) The ACID properties are often used to define transactional semantics.

(a) Define "atomicity" as used in the ACID context. [3]

(b) Define "durability" as used in the ACID context. [3]

(ii) Define what it means for an interleaving of 2 transactions to be serially equivalent? [3]

(iii) Suppose Mr. Kamal is a customer of a multi-national bank. He is currently in the USA on a business trip. His bank maintains two replicas of "customer accounts" database, one in the USA and another in Colombo. Kamal is a customer of the Colombo branch.

(a) What concurrency issues can occur if Mr. Kamal tries to withdraw money from an ATM in the USA while his branch in Colombo tries to deposit interest to his account? Explain using an example. [6]

(b) Explain a suitable solution(s) to address these issues while satisfying ACID properties. You may recommend established techniques with justification. [6]

Question 5 (25 marks)

(i) Lamport's Timestamps are calculated based on the 2 rules given below.

Processes update their logical clocks and transmit their logical clock values in messages as follows:

LC1

$L_i := L_i + 1$, before each event is recorded at processes P_i

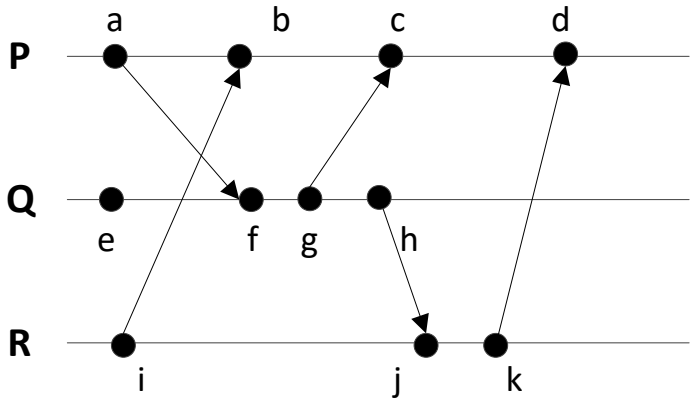
LC2

When P_i sends a message m, logical clock value $t = L_i$, is piggybacked with the message

On receiving (m, t), a process P_j computes $L_j := \max(L_j, t)$

Then computes $L_j := L_j + 1$ before logically timestamping the event *receive*(m)

a) Label the following diagram with Lamport's Timestamps.



[5]

b) Identify 2 event pairs with the same Lamport time stamp. [2]

c) How can we build a total order of events such that 2 events will not have the same time stamp? What will be the new time stamps for the pairs of events you identified in question b? [4]

(ii) What is middleware? How can it be used to provide a Platform as a Service (PaaS) in cloud computing? [5]

(iii) Discuss whether cloud computing is suitable for the following applications.

a) An online retailer that sells seasonal goods. [3]

b) Weather monitoring system that pulls data from weather stations, radars, and satellites. [3]

c) A recommendation system that recommends you what songs to listen based on your past preferences and preference of millions of others. [3]

---------- END OF THE PAPER ----------