PAPER CODE NO.	EXAMINER : Boris Konev
COMP212	DEPARTMENT : Computer Science Tel. No. 0151 794 6799

THE UNIVERSITY
of LIVERPOOL

May 2006 EXAMINATIONS

Bachelor of Arts: Year 2
Bachelor of Science : Year 2
No qualification aimed for: Year 1

Distributed Systems

TIME ALLOWED : Two Hours

INSTRUCTIONS TO CANDIDATES

Answer any **four** questions only.
Each question is worth 25 marks.

If you attempt to answer more questions than the required number of questions (in any section), the marks awarded for the excess questions will be discarded (starting with your lowest mark).

THE UNIVERSITY
of LIVERPOOL

Question 1

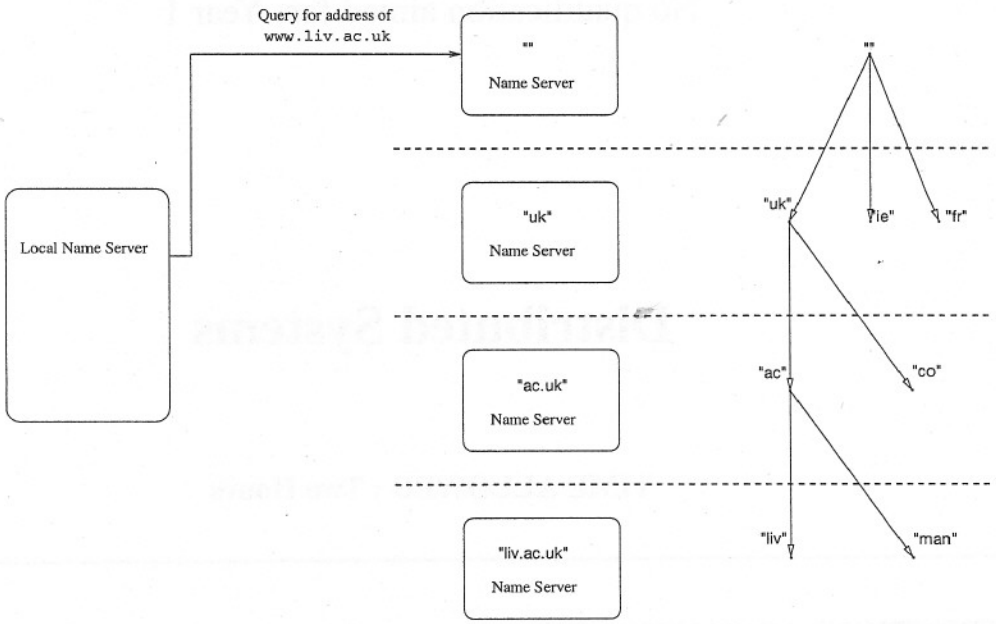
A. Give four examples of resources to be shared in a distributed system. **2 marks**

B. Explain what is meant by (distribution) transparency. **4 marks**

C. What is the difference between a distributed operating system and a network operating system? **4 marks**

D. What is a three-tiered client-server architecture? **5 marks**

E. A local name server performs a **recursive** query for the address of `www.csc.liv.ac.uk` starting from a root node. The name hierarchy and name servers are given in the picture.



Assuming that no data can be found in server caches, represent graphically and describe the communication between the name servers for this query. **7 marks**

F. Communication in Jini is based on which of the following:

a) message queueing,

b) Java RMI, or

c) remote procedure calls? **3 mark**

Question 2

A. Classify the following protocols: IP, TCP, HTTP according to the ISO OSI classification. **3 marks**

B. Describe precisely what is meant by a scalable system. **3 marks**

C. Why are transport-level communication services often inappropriate for building distributed applications? **3 marks**

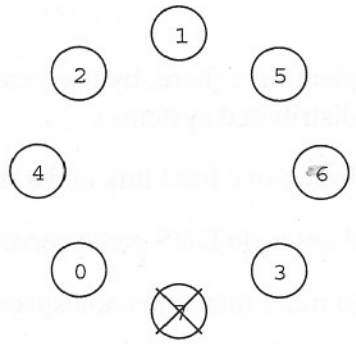
D. Processes in distributed systems are often divided into two groups: clients and servers.

(a) Describe the client-server model. **3 marks**

(b) Give a graphical representation of the request-reply interaction between a client and a server. **2 marks**

(c) Describe a 3-tier architecture. **3 marks**

E. Consider a group of eight processors numbered 0 to 7. Previously process 7 was the coordinator, but it has crashed.



Process 5 is the first one to notice this, so it initiates a new election by the ***bully*** algorithm. Apply the algorithm to the described situation, show messages being sent between the processes, and find who will be the new coordinator. **8 marks**

Question 3

A. What do client and server stubs do for remote procedure call (RPC)? **4 marks**

B. (a) What is the purpose of a registry for object-oriented remote method invocation (RMI)? **2 marks**

(b) If a client invokes the same remote method again, does it necessarily use the registry? **2 marks**

C. Why are threads (or processes) so important for distributed systems? **4 marks**

D. Give an example of a stateless protocol. **2 marks**

E. Is it possible to connect across a network two components of a distributed system written in different languages (e.g., COBOL and Java) using a *distributed object-based system*? Is it possible to connect such components by means of Java RMI? Give brief justifications for your answers. **6 marks**

F. Are processes using Jini for coordination required to coexist at the same time? Explain why/why not. **5 marks**

Question 4

A. Give two examples of identifiers (here, by the term "identifiers" we mean those identifiers that refer to entities in distributed systems). **2 marks**

B. Explain the difference between a hard link and a soft link in UNIX filesystems. **4 marks**

C. Which name server addresses do DNS name servers hold by default, and why? **4 marks**

D. Can one name belong to more than one namespace? **3 marks**

E. The distributed file system Coda allows a client to continue working with a shared file even if there is no network connection between the client and the server. Explain how this is made possible. **4 marks**

F. Consider a network consisting of 5 computers, A (coordinator), B, C, D, and E. At $14:35$ the coordinator decides to synchronise the clock of all computers in the network. At that moment, the clock of every computer in the network shows the following.

Computer	Clock
A	14:30
B	14:37
C	14:35
D	14:28
E	14:25

Apply the Berkley clock synchronisation algorithm to this situation, show the stages of computation, and write what will be the outcome of the synchronisation. The time needed for computation and for network communication is negligible. **8 marks**

Question 5

A. What is the purpose of Lamport's Timestamps? **5 marks**

B. A clock is reading 10:15:35.0 (hr:min:sec) when it is discovered to be 10 sec fast. Are there any drawbacks in setting it back to the correct time at that point? **2 marks**

C. Define what a *critical section* is. **4 marks**

D. What is a proxy server? **4 marks**

E. Consider the following modification of Ricart and Agrawala's distributed mutual exclusion algorithm considered in the lectures (the modified part is highlighted in *italics*).

- A process wanting to enter critical sections (CS) sends a message with (`cs name, process id, current time`) to all processes (including itself).
- When a process receives a CS request from another process, it reacts based on its current state with respect to the CS requested as follows:
  - If the receiver is <u>not</u> in the CS and it does not want to enter the CS, it sends an OK message to the sender.
  - If the receiver is in the CS, it does not reply and queues the request.
  - If the receiver wants to enter the CS but has not yet, it compares the timestamp of the incoming message with the timestamp of its message sent to everyone. The *highest* timestamp wins.
    - If the incoming timestamp is *higher*, the receiver sends an OK message to the sender.
    - If its own timestamp is *higher*, the receiver queues the request and sends nothing.
  - After sending out requests asking permission to enter a critical section, a process waits until everyone else has given permission. As soon as all the permissions are in, it may enter the critical section. When it exits the section, it sends OK to all the processes on its queue and deletes them all from the queue.

Will this modification satisfy the Safety and Liveness properties? Explain why/why not for **both** properties. **8 marks**

F. Is it possible to describe *semantics* of objects and services in CORBA IDL? **2 marks**

Question 6

A. Is agreement possible in a system with unreliable communication? **2 marks**

B. What purpose is achieved by means of physical triple modular redundancy? **3 marks**

C. Can the model of triple modular redundancy handle Byzantine failures? **3 marks**

D. Consider two communication services for use in asynchronous distributed systems. In service A, messages may be lost, corrupted, duplicated or delayed. In service B, messages may be delayed or delivered too fast but messages arrive ordered and with the correct contents.

   (a) Describe the classes of failure exhibited by each service. **5 marks**

   (b) Can service B be described as a reliable communication service? **3 marks**

E. If Alice wants to send secret information to Bob, should she know Bob's public or Bob's private key? **2 marks**

F. Compare secret key encryption with public key encryption. Give two advantages of secret key over public key encryption and two advantages of public key over secret key encryption. Can a protocol use both secret and public key encryptions to have the advantages of both? **7 marks**