

UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2002

BEng Honours Degree in Computing Part III
BEng Honours Degree in Information Systems Engineering Part III
MEng Honours Degree in Information Systems Engineering Part III
for Internal Students of the Imperial College of Science, Technology and Medicine

*This paper is also taken for the relevant examinations for the
Associateship of the City and Guilds of London Institute*

PAPER C335=I3.14

DISTRIBUTED SYSTEMS

Wednesday 1 May 2002, 10:00
Duration: 120 minutes

Answer THREE questions

Paper contains 4 questions
Calculators not required

- 1a The following message passing primitives are supported by a set of library calls:

send (dest, msg) – an *asynchronous* send message primitive, where dest is the name of the process to which the message msg is to be sent.

receive (source, msg) – this causes the receiving process to block waiting for a message from the process with name source. msg is a buffer into which the incoming message is copied.

receiveany (source, msg) – the process is blocked waiting for a message from any source. The name of the sender is received in source and the incoming message is received in msg.

- i) Explain what is meant by an *asynchronous* send message primitive and why it may lead to buffer exhaustion at the receiver.
- ii) Why is there a need for two types of receive primitives?

- b Using the above message primitives, design a simple printer service for a distributed system with multiple printers, each controlled by a process called *printer*. When a *user* process wants to print a document it sends a message containing its process type (i.e. user) to a single *coordinator* process which allocates free printers. The coordinator replies with the name of a free printer when one is available. When a printer is free it sends a message containing its process type (i.e. printer) to the coordinator to indicate it is now available for printing.

Give pseudocode outlines for the *user*, *coordinator* and *printer* processes, using the above message primitives. Your solution should describe any datastructures needed by the coordinator process.

Assume the printer process has sufficient buffer space for a single message containing one page to print, and that communication is reliable so timeouts and retransmissions can be ignored.

The two parts carry, respectively, 20% and 80% of the marks.

- 2a Explain the terms *first party binding* and *third party binding*. Discuss the advantages/disadvantages of each.
- b A debugging tool provides users with the capability of locating all objects in a distributed system from a directory, determining the interface class of an object from the directory, binding to the object, and then permitting the user to input suitable parameters for the various invocations which can be invoked on the object to test it. Explain how Java and CORBA each permit the binding of the debugging tool to any arbitrary object.
- c A distributed programming system uses a name service which maintains replicated information in a number of servers connected to a large network. An update to the information base can be initiated at any site in the network. Explain what would be required to maintain *strong consistency* amongst the replicated information and discuss why a *weak consistency* strategy is better.

The three parts carry, respectively, 35%, 30 and 35% of the marks.

- 3a Explain each of the following security terms:
- Authentication
 - Authorisation Policy
 - Confidentiality
 - Integrity
 - Non-repudiation
- b Discuss the properties of a message digest. Explain how a cryptographic (asymmetric) public-key system can be used, together with a suitable hash function, to generate a message digest to detect modification and guarantee authenticity of a message.
- c The Department has decided to implement an electronic token payment system which allows staff and students to print to any print server, by sending a file, plus required tokens to the server. Tokens are generated by a central token server. The print server acknowledges receipt and returns any change in the form of tokens. Tokens come in a range of values and cannot be forged or reused. They are issued to individual members of the Department, they cannot be used by anyone else and they expire at the end of the academic year. Assume an asymmetric public key system.

Describe the format of a token justifying each of the fields. Explain what actions the print server must perform in order to check the validity of the token. Describe what information the print servers maintain in order to process print requests and what must be held by the central token server, discussing how the information should be organised to improve performance.

The three parts carry, respectively, 25%, 30 and 45% of the marks.

- 4a Briefly discuss the main characteristics of a peer-to-peer network.
- b You have to design a peer-to-peer network to be used on a world-wide basis to reliably store cooking recipes.
- i) Describe a technique for generating globally unique message identifiers. Explain why the identifier is unique.
 - ii) Describe a technique for discovering to which nodes you should set up connections as 'neighbours'. Explain how your technique prevents overloading specific nodes with network connections.
 - iii) Describe how to perform a search for a chocolate cake recipe. Your solution must indicate how the search terminates and prevents looping.

The four parts carry, respectively, 25%, 20%, 15% and 40% of the marks.