UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2002

BEng Honours Degree in Computing Part II
MEng Honours Degrees in Computing Part II
BEng Honours Degree in Information Systems Engineering Part II
MEng Honours Degree in Information Systems Engineering Part II
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 C211=I2.5

OPERATING SYSTEMS II

Monday 29 April 2002, 14:00
Duration: 90 minutes
(Reading time 5 minutes)

Answer THREE questions

Paper contains 4 questions
Calculators not required

1. a. Give the design (pseudo code) for both the *synchronous send* and the *receive* procedure as used in a message passing mechanism.

   b. Explain why the kernel of a system implementing *asynchronous message passing* needs to provide *buffering*, whereas this is not needed when *synchronous* message passing is used.

   c. Give a situation in which it could be important to be able to specify a limit on the time that one is prepared to wait for a message?

   d. In the context of processes that run concurrently, when protecting access to critical regions, the following instruction can be used:

   ```
   tsl reg,lock
   ```

   What are '**`tsl`**' and '**`lock`**' here?

   Give the implementation of protection of critical regions, i.e. give assembler specifications of both '**`enter_cr`**' and '**`leave_cr`**', using **`locks`**.

The four parts carry 35%, 25%, 15% and 25% of the marks, respectively.

2. a. What are the essential differences between *threads* and *processes.*

   b. Explain in brief terms why the *server paradigm* solves the problem of mutual exclusion.

   c. What is the main rôle of the scheduler? Give the situations in which it will be invoked in an operating system that uses a message passing mechanism for communication between processes, together with a round-robin scheduling algorithm.

   d. Give the *process state diagram.* Give an extension that deals with the notion of a process being *swapped out*, i.e., the process descriptor exists, but no longer has its code in memory.

   Try to be concise for each case. 100 words per part should be the maximum.

The four parts carry, each, 25% of the marks.

3. I/O and disk organisation

   a. Briefly show in a diagram how I/O is organised in an operating system like Minix. How is this organisation fundamentally different from Unix?

   b. Explain the main difference between ordinary disk organisations and RAID disk organisation, and give at least three examples in which RAID disk organisation offers advantages over ordinary disk organisation.

   c. Suppose that a disk request queue contains requests for sectors on the following cylinders: 400, 20, 19, 74, 899. Assume that the disk arm is over cylinder 200. The operating system supports three different disk arm scheduling policies: Shortest seek time first (SSTF), SCAN and C-SCAN.

      i. For all three scheduling techniques, briefly describe the scheduling policy.

      ii. For all three scheduling techniques, show in which order the requests above will be serviced.

   d. A disk manufacturer is considering two options to increase the capacity of disks. The first option is to increase the number of sectors per track on the disk, while keeping the number of bytes per sector and the number of tracks per surface constant. The second option is to increase the number of tracks per surface while keeping the number of bytes per sector and the number of sectors per track constant. What effect will these changes have on the time required by the disk to service requests? Be specific. For each of the two possible options indicate which components in the disk's service time will increase or decrease relative to the original disk.

The four parts carry, respectively, 25%, 25%, 30% and 20%

4. a. Describe the information stored in inodes and directories. How are inodes and directories related?

   b. Briefly explain the following three file allocation methods (use a diagram if necessary) and discuss their relative overheads when accessing files:

      i. block linkage
      ii. file allocation table
      iii. inodes

   c. Consider a file system using inodes with 8 direct pointers, one single-indirect pointer and one double-indirect pointer. The block size of the file system is 512 bytes and the size of a block pointer is 4 bytes. Assume that a file of a size of 200,000 bytes has been opened. Furthermore assume that the inode has already been loaded into memory.

      i. A process requests to read bytes 9,500 to 10,300 from the file. How many blocks must the file system read from disk to satisfy the request?
      ii. How many blocks must the file system read from disk to read the entire file?

   d. Consider a file system using inodes with 8 direct pointers, one single-indirect pointer, one double-indirect pointer and one triple-indirect pointer. Assume that the size of a pointer is 4 bytes.

      i. Calculate the maximum size of files in a file system with a block size of 1024 bytes.
      ii. Estimate by approximately what factor does the maximum possible file size increase if the block size B is doubled. Your answer should be an integer number.

The four parts carry, respectively, 25%, 30%, 25% and 20%

End of Paper