UNIVERSITY OF LONDON

IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1996

BEng Honours Degree in Computing Part I
MEng Honours Degrees in Computing Part I
BEng Honours Degree in Information Systems Engineering Part I
MEng Honours Degree in Information Systems Engineering Part I
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 1.5 / I1.5

OPERATING SYSTEMS I

Tuesday, May 7th 1996, 2.00 - 3.30

Answer THREE questions

For admin. only: paper contains
4 questions
2 pages (excluding cover page)

1. Input and Output

1a. Explain why it is necessary for operating systems to control all I/O operations and how their control of I/O can be made reliable. Discuss how the efficiency of I/O management by the Operating System influences overall efficiency of a multi-processed computer system.

1b. Explain why software buffers are used for I/O devices. Describe an organised scheme for the management of software buffers for I/O and show how these buffers may be used for spooling.

1c. Show an I/O input procedure in assembler language which handles single data items without interrupts. You may use any easily interpretable assembly language. The code should be complete and terms which are not defined by the code should be described by words.

The three parts carry , respectively 30%, 30%, 40% of the marks.

2. Memory Management

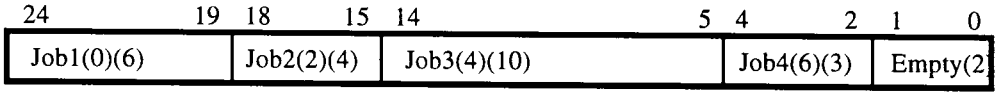
2a. Discuss two ways by which the Operating System Loader can load an object program at a different starting address from which it was compiled and still produce a correct executable program.

2b. Discuss why memory protection is essential in all modern computer systems. Describe how memory protection is achieved in two different systems; one has a base and limit register, the other hardware paging.

2c. An Operating System allocates 25 Kbytes for very small jobs. The jobs arrive at 2 seconds intervals and they all complete their execution in 8 seconds. The O.S. does not preempt jobs; i.e.; the jobs are loaded into memory in the order they arrive. The jobs and their sizes from t=0 to t=14 seconds are shown below:

Job number -->	1	2	3	4	5	6	7	8	
Arrival time -->	0	2	4	6	8	10	12	14	seconds
Size -->	6	4	10	3	2	10	8	4	

At time 6 seconds the following is the memory map where the 25 Kbyte blocks are numbered from 0 to 24.



Using the best fit memory allocation method show how the jobs are loaded until all jobs have been executed. Calculate the average waiting time for the eight jobs.

The three parts carry , respectively 30%, 30%, 40% of the marks.

3. Processor Management and Hardware Paging

3a. Discuss what a process is and what information the Process Control Block (PCB) holds. Explain how an Operating System manages the execution of loaded processes in a multiprocessed computer environment if the computer system has only one processor.

3b. Write a general procedure in easily interpretable assembler language which fills a software buffer of 512 words capacity and is called by the following assembler language calling sequence:

```
MOVE    #BUFFAD, A0     ; The buffer base address.
MOVE    DATAWORD, D1    ; The next data to be placed in the buffer.
JSR     FILLBUF         ; Call the fill buffer procedure.
```

The first word in the buffer is the data pointer with values ranging from 0 (empty) to 512 (full). The routine returns the number of data words in the buffer in register **D0** after the data was placed in the buffer, or it returns the value -1 if the buffer was full when the routine was called.

Since this routine can be called by interrupt service routines, you have to provide mutual exclusion. The computer has a **TSET** (Test and Set) instruction which you may use for this purpose.

3c. Describe the following terms and their role in computer systems which have paged memory hardware.

(i) Demand Paging
(ii) Working Set
(iii) Thrashing

The three parts carry , respectively 30%, 40%, 30% of the marks.

4. General Principles and Process Synchronisation

4a. A computer system receives jobs and sends back results through a communication link. Discuss how the system's performance seen by the users and the utilisation of computer resources can be influenced by the number of multiprocessed jobs (from a single running job to a large number of concurrently running jobs).

4b. Describe how binary and counting semaphores are used for process synchronisation. Discuss how the performance of a multi-processed computer system is influenced by providing synchronisation through the Operating System as opposed to semaphores used by application programs. Demonstrate the use of a binary semaphore by showing appropriate assembler language codes for the *wait()* and *signal()* routines.

4c. What are the necessary conditions for deadlock to occur? Discuss means by which deadlock can be prevented or avoided. Describe the simplest situation when two processes become deadlocked.

The three parts carry , respectively 30%, 40%, 30% of the marks.

End of paper