Two hours

Question A1 and Question B1 are COMPULSORY

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Operating Systems

Date: Friday 20th January 2017

Time: 09:45 - 11:45

Please answer Questions A1 and B1
and
TWO other Questions from A2, B2 or B3

Use a SEPARATE answerbook for each SECTION

This is a CLOSED book examination

The use of electronic calculators is NOT permitted

Section A

A1. **Compulsory**

A1.a) A mobile video streaming application streams videos encoded with a bit rate of 6Mbps (Million bits per second) and a frame rate of 30 frames per second over a wireless network link with a transfer rate of 40Mbps. How long does it take to load a single frame in to the CPU's Memory? How many clock cycles of a 2GHz CPU is this equivalent to? For simplicity assume that: (1) the bit rate of the encoding is fixed so that every frame is the same size; and (2) the time taken to load memory from the network is negligible. (2 marks)

A1.b) What is the ‘address space’ of a process? Name two regions that may be contained in a process’ address space. (2 marks)

A1.c) What is the key difference between a system call and a call to an ordinary method or function. Briefly explain why this difference is important. (2 marks)

A1.d) What is the difference between a monolithic operating system and one constructed around a microkernel? (2 marks)

A1.e) What does the term *race condition* mean? Indicate one mechanism that can be used to avoid race conditions. (2 marks)

A2.a) What is preemptive process scheduling? Name two possible reasons a process may be preempted? (2 marks)

A2.b) In the context of process scheduling, what do the terms static priority and dynamic priority mean? Give an example of each form of priority. (2 marks)

A2.c) Three processes, A, B and C, have the following computational needs: Process A requires a 7 time-unit CPU burst followed by a 4 time-unit I/O burst then a 7 time-unit CPU burst before it terminates; Process B requires a 3 time-unit CPU burst followed by a 6 time-unit I/O burst then a 3 time-unit CPU burst before it terminates; Process C requires a 5 time-unit CPU burst followed by a 4 time-unit I/O burst then a 5 time-unit CPU burst before it terminates. Draw a diagram showing the states (Running/Blocked/Ready) of these processes as they are run by a non-preemptive Shortest Job First (SJF) scheduler until they all terminate, assuming that they all start ready at time-unit 0 and the time for a context switch is negligible. What is the average turnaround time (the time between entering the ready queue for the first time and terminating) for the three processes? What is the average wait time (the time a process spends in the ready queue) for the three processes? (3 marks)

A2.d) The three processes in part A2.c) have the same computational needs, but staggered arrival: Process A enters the ready queue for the first time at time-unit 0, Process B enters the ready queue for the first time at time-unit 1 and process C enters the ready queue at time-unit 2. The processes are now subjected to a preemptive Shortest Remaining Time First/Next (SRTF) scheme. In the case of two variables with the same CPU-burst time remaining the process that (re-)entered the ready queue first is selected to run. Draw a diagram showing the states of the processes from start to finish. What is the average turnaround time for the three processes? What is the average wait time for the three processes? (4 marks)

A2.e) Briefly describe a “FAT” (File Allocation Table) and an i-node (index node). (2 marks)

A2.f) In a system using *index nodes* (*i-nodes*) and blocks of 4 kilobytes, assume that each i-node contains up to 7 pointers to the first 7 successive blocks of the file on disk, then up to one pointer to a *single indirect block* on disk that contains 256 further pointers to the next 256 successive blocks of the file on disk. Any unused pointers contain a special ‘null’ value. What is the size (in kilobytes) of the largest possible file? What would be the size if we replace one of the 7 direct pointers to a pointer to a pointer to a *double indirect block* that contains 256 further pointers to single indirect blocks on disk that each contains 256 pointers to the next 256 successive blocks of the file on disk? (4 marks)

A2.g) When accessing a file in a FAT-based file system, first an algorithm provides the number for the first block of a file. Describe carefully how the file system accesses to the whole file. (3 marks)

[PTO]

Section B

B1. **Compulsory**

B1.a) In the context of converting an address generated by a program [a compiler] to the actual address; state:
- i) The names of the two memories involved; and
- ii) The unit that performs [undertakes] this translation process. (2 marks)

B1.b) Differentiate between multiprogramming and fixed partitions. (2 marks)

B1.c) Differentiate between software interrupts and hardware interrupts; in your answer give a brief description of each type of interrupt. (2 marks)

B1.d) State three distinct reasons why it is necessary for processes to share memory. (2 marks)

B1.e) In the context of permission information, e.g.: read (R), write (W), & execute (X), outline three different ways in which permissions are typically used? What can the RWX permission information control access to with respect to security and protection? (2 marks)

B2.

B2.a) The following question address segmentation, which you were introduced to in your lecture series.

B2.a.1. In the context of memory management, what is a *segment*?

B2.a.2. What advantages are there in using segmentation?
Hint: What do segments *ensure* and *prevent*?

B2.a.3. Explain how logical addresses are translated into physical addresses in a system which has support for segmentation.

(6 marks)

B2.b) In the context of page-replacement algorithms, explain the Not Recently Used (NRU) page replacement algorithm. (4 marks)

B2.c) Consider a process which accesses pages in the order:

0,1,0,2,3,1,0,1,4,0,3,1.

Suppose that it has room in memory for exactly 3 pages, and that none of the pages mentioned above are loaded at the start of the sequence, what would be the number of page faults if Least Recently Used page-replacement algorithm is used? Draw up a 3 page frame memory table, as depicted in figure B2.c, to enable the calculation to be undertaken. (5 marks)

Requested Order

	0	1	0	2	3	1	0	1	4	0	3	1	
3 page frames													Most recent
													Second most
													Third most
PF													

M=Miss and H=Hit

Question B2.c. Diagram of a 3 page frame memory.

[PTO]

B2.d) A processor has an address bus width of 36 bits. The processor uses this address bus to generate the logical address for a virtual memory system. The computer system in which the processor resides has 1GB of main memory and uses a paged virtual memory system that has a page size of 64KB.

In each case below, please show full working in your answer:

(i) How many pages are there in the paged virtual memory system? (1 mark)

(ii) How many page frames are there in main memory? (1 mark)

(iii) The logical address generated by the processor is made up of two bit fields, a 'page number' (or tag) and an 'offset.' How many bits in the logical address are allocated to the tag (or page number) and how many to the offset? (3 marks)

B3.

B3.a) An Input/Output (I/O) scheme in a computer system may be controlled by a polled I/O scheme or by an interrupt-driven I/O scheme. In the context of a simple input device (a keyboard) carefully explain how polled and interrupt-driven I/O schemes work. Which would you expect to give the best performance? (6 marks)

B3.b) Direct memory access (DMA) is interrupt driven. Given that a processor writes to a disk, utilizing DMA, describe the four-step DMA process for writing data. (4 marks)

B3.c) What is a deadlock? Describe how it can occur and show an example. (3 marks)

B3.d) In a certain system, the execution of three threads is synchronised using three semaphores, S1, S2 and S3, as shown below. All semaphores are initialised to zero. All three semaphores are used only in the sections of code shown below:

	Thread A	Thread B	Thread C
	…	…	…
1:	P(S1)	P(S2)	V(S2)
2:	y=x+1	P(S2)	V(S1)
3:	V(S3)	y=x*y	P(S3)
4:	V(S2)	V(S3)	P(S3)
5:	V(S1)		x=y/2
6:			V(S2)
	…	…	…

i) If the variables x and y are defined as integer shared variables, initialised to 10 and 50 respectively, and are not assigned a value in any other sections of the code apart from those shown above, what will be their value when all three threads have finished executing? What will be the values of the three semaphores S1, S2 and S3? Justify your answers. (4 marks)

ii) We want to simplify the code and remove one P(S2) in Thread B (line 2) and one P(S3) in Thread C (line 4). Indicate which other changes would be necessary (if any) to ensure that the outcome does not change, i.e. the value of the shared variables x and y when all three threads have finished executing is the same. Assume that the initial value of the semaphores is still 0 in all cases. Justify your answer. (3 marks)

END OF EXAMINATION