Two hours

Question ONE is COMPULSORY

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Operating Systems

Date: Friday 25th January 2013

Time: 14:00 - 16:00

Please answer Question ONE and any TWO other Questions
from the other THREE Questions provided

This is a CLOSED book examination

The use of electronic calculators is NOT permitted

[PTO]

1. **Compulsory**

    a) Explain what the term 'deadlock' means, and how it might occur.

    (2 marks)

    b) Computers typically operate in one of two modes, namely *user mode* or *system mode* (the latter is sometimes called privileged mode, supervisor mode or kernel mode). Explain why these two modes are needed and how they differ from each other.

    (2 marks)

    c) Briefly explain how *starvation* may occur in process scheduling.

    (2 marks)

    d) Why is the size of the time slice in pre-emptive scheduling algorithms chosen to be significantly higher than the time taken for a context switch?

    (2 marks)

    e) Explain what is meant by the term 'multiprogramming'.

    (2 marks)

    f) Explain the difference between a virtual memory address and a physical memory address.

    (2 marks)

    g) What is a 'page replacement algorithm'?

    (2 marks)

    h) What is meant by 'memory mapped I/O'?

    (2 marks)

    i) What is a 'page fault'? Describe how a page fault is handled by the Memory Management Unit and the operating system.

    (2 marks)

    j) Briefly explain why Java code that waits in a synchronized method for a condition to hold will commonly retest the condition when it has been released from its wait.

    (2 marks)

2. a) Explain briefly the difference between a process and a thread.

(2 marks)

b) Explain briefly what a CPU burst is and what an I/O burst is. What is a CPU-bound process, and what is an I/O-bound process? Why is it a good strategy in process scheduling to give higher priority to I/O-bound processes?

(4 marks)

c) Three processes A, B and C all alternate between a CPU burst of 3 time-units and an I/O burst of 3 time-units. Draw a diagram showing the states of these processes as they are run by a Round Robin scheduler for a total of 30 time-units, assuming that they all start ready at time-unit 0, the time-slice adopted by the scheduler is 2 time-units, and the time for a context switch is negligible. For what percentage of the time is the CPU executing user processes?

(4 marks)

d) A new scheduler is introduced using priority queues. There are two queues, Q1 (responsible for scheduling processes A and B) and Q2 (responsible for scheduling process C). Assuming they have ready processes, the two queues access the CPU alternately, as follows: Q1 gets 5 time-units, then Q2 gets 2 time-units, then Q1 gets 5 time-units, and so on. Processes in each queue are executed in Round Robin fashion with a time-slice of 1 time-unit. If a queue runs out of ready processes before its allocated time-units have been used, it yields access to the CPU to the other queue. Apart from this, the situation is as described for part c) above. Draw a diagram showing the states of the three processes A, B and C as they are run by the scheduler for a total of 30 time-units. For what percentage of the time is the CPU executing user processes?

(4 marks)

e) In a certain system, the execution of three threads is synchronised using three semaphores, S1, S2 and S3, as shown below. Semaphores S1 and S2 are initialised to zero, while semaphore S3 is initialised to 1. All three semaphores are used only in the sections of code shown below.

```
Thread A          Thread B          Thread C
…                 …                 …
P(S1)             P(S2)             P(S3)
P(S1)             P(S1)             V(S1)
x=3*x+4           x=x+7             x=x*5
V(S2)             V(S2)             V(S1)
V(S1)             V(S1)             V(S3)
…                 …                 …
```

Question 2e continues overleaf

Question 2e continued

      i)   If the variable x is defined as an integer shared variable, initialised to 1, and is not assigned a value in any other sections of the code apart from those shown above, what will be its value when all threads have finished executing? What will be the values in the three semaphores? Justify your answers.

<div align="right">(3 marks)</div>

      ii)   Would it be possible to guarantee the same final value for x using only two semaphores? Would it be possible to guarantee the same final value for x using only one semaphore? Justify your answers.

<div align="right">(3 marks)</div>

3. a) On a paged machine with 3 pages frames (PFs) available for it, a particular process makes accesses to the following pages in the order given:

    0, 3, 7, 1, 3, 2, 1, 3, 7

    Show the contents of the 3 page frames and the cumulative total number of page faults (PF) after each memory access assuming that an LRU page replacement algorithm is in use and that the page frames are initially empty. The type of diagram you should draw up is depicted in Figure 3a.

    (4 marks)

```
Access:          0   3   7   1   3   2   1   3   7

Most recent:     X   X   X   X   X   X   X   X   X   )   Contents of
Second most :    X   X   X   X   X   X   X   X   X   )   page frames
Third most:      X   X   X   X   X   X   X   X   X   )   in memory

Total PFs:       X   X   X   X   X   X   X   X   X
```
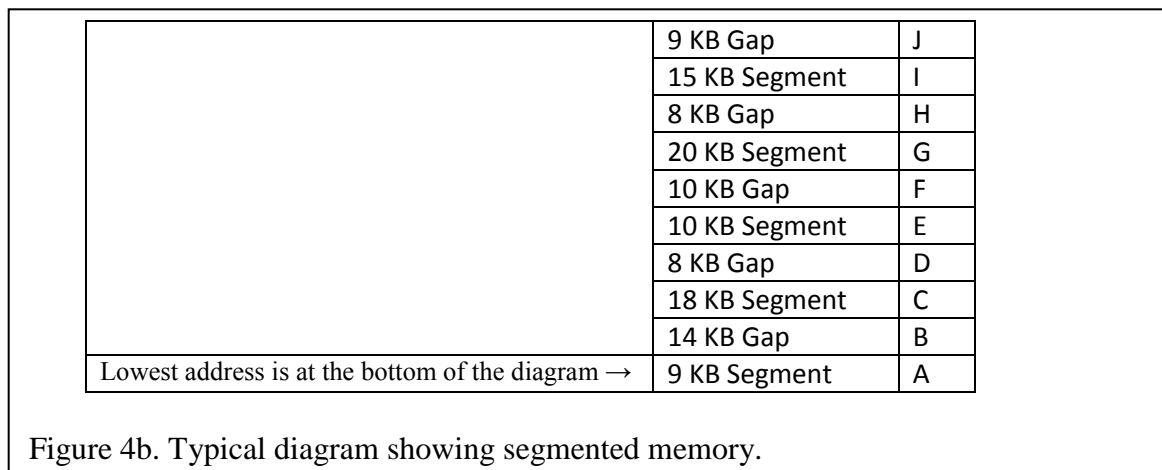
Figure 3a. Typical diagram showing 3 page frames and the cumulative total number of page faults.

b) A 16-bit virtual address is divided into an 8-bit page number and an 8-bit page offset; whereas the associated physical address has a 5-bit page frame with the appropriate page offset address sizing. Show, with the aid of a fully labelled diagram, the table structure necessary to convert this virtual address into a real physical address.

(6 marks)

c) With the aid of a diagram, describe the structure of a simple base and limit system and explain how it achieves relocation. **NOTE**: For full marks your answer must contain a detailed concise diagram and explanation; e.g. a fully labelled diagram and a concise explicit description of a base and limits system and how it achieves relocation.

(6 marks)

d) Given a 1G address space and associated 128K page size, calculate the number of pages in the virtual address space. **NOTE**: To gain full marks you must show full working.

(2 marks)

e) Given a physical address space of 2G and associated 64K block size, calculate the number of page frames in the physical address space. **NOTE**: To gain full marks you must show full working.

(2 marks)

4. a) Explain how a peripheral communicates with the CPU using interrupts.

(3 marks)

b) Segmented memory is an alternative view to that adopted by paged memory; the following questions relate to segmented memory.

i) Explain the difference between a *page* and a *segment* in a virtual memory system.

(3 marks)

ii) A computer system uses segmented virtual memory only (no pages). The state of the memory at a given time is shown in Figure 4b. Indicate what happens when a new segment requiring 9KB of memory space is loaded using the following algorithms:
    1) Best Fit; and
    2) First Fit.
State where the 9KB will be placed given 1) and 2), and, due to the different algorithms, if any issues arise. Note that it is assumed that the lowest address is at the bottom of the diagram, and the third column indexes all the segments A, …, J.

(4 marks)

	9 KB Gap	J
	15 KB Segment	I
	8 KB Gap	H
	20 KB Segment	G
	10 KB Gap	F
	10 KB Segment	E
	8 KB Gap	D
	18 KB Segment	C
	14 KB Gap	B
Lowest address is at the bottom of the diagram →	9 KB Segment	A

Figure 4b. Typical diagram showing segmented memory.

c) A *pathname* in a hierarchical file system defines how a file is to be located on the disk.

i) Explain the distinction between an *absolute pathname* and a *relative pathname*.

(2 marks)

Question 4c continued

    ii) Describe in detail the algorithm used to locate a file given an *absolute pathname* or given a *relative pathname*.

                                                       (3 marks)

    iii) In a system using *index nodes* (*i-nodes*), the above algorithm will first find the identifier of the i-node that describes the target file. Explain how the identified i-node is found on disk.

                                                       (2 marks)

    iv) Assume that each i-node contains up to 8 pointers to the first 8 successive blocks of the file on disk, then up to one pointer to a *single indirect block* on disk that contains 512 further pointers to the next 512 successive blocks of the file on disk, and then up to one pointer to a *double indirect block* that contains 512 further pointers to single indirect blocks on disk that each contain 512 pointers to the next 512 successive blocks of the file on disk. Any unused pointers contain a special 'null' value. What is the size (in blocks) of the largest possible file?

                                                       (3 marks)

END OF EXAMINATION