2019 EXAMINATIONS

Lancaster University

Part II

COMPUTING AND COMMUNICATIONS [2 Hours]

SCC.211 Operating Systems

Candidates are asked to answer THREE questions from FOUR; each question is worth a total of 20 marks.

Use a <u>separate</u> answer book for <u>each</u> question.

Question 1

1.a

i. Define what is meant by the term concurrency. **[2 marks]**

ii. How does concurrency improve the 'performance' of software? **[2 marks]**

iii. Explain what is meant by the term *race condition*, the reason for its occurrence, and what synchronisation can be leveraged to minimise its effect. **[3 marks]**

1.b Within Unix, processes are spawned using fork( ), which creates an almost identical copy (child) of the calling process (parent).

i. Provide one similarity and one difference between the parent and child process. **[2 marks]**

ii. Create a small piece of code in which a single process will spawn 15 additional child processes by using fork( ). What is the equation used to calculate how many times fork( ) must be called in order to create *x* processes spawned from a single process? **[3 marks]**

1.c

i. A programmer is tasked to create a program that traverses an array of *n* strings and simultaneously writes their contents to a shared text file. The programmer is wishing to perform this as fast as possible, and has decided to do so via threading with each thread processing a segment of the array.

Write an implementation for this scenario: You are free to use Java-based or C-based pseudo-code and any concurrency mechanisms you deem fit. A concise description of the semantics of the concurrent primitives should be provided, which argues how they ensure determinism. Code should be as precise and complete as possible. **[8 marks]**

Total 20 marks

Question 2

2.a

i. Define the following concepts and provide an example for each:

Program, process, and processor. **[6 marks]**

ii. Discuss two scenarios where shared-memory communication is not able to achieve concurrency, and describe why. **[3 marks]**

iii. Define what is meant by message passing, and describe the two abstract primitives that it uses. **[3 marks]**

2.b The producer-consumer problem is a well-established multi-process synchronisation problem. One process (producer) will submit elements, *item,* into a fixed sized buffer that are removed by another process (consumer). In this scenario the order of execution of processes is non-deterministic. The code below shows a half-completed solution:

```
class producerConsumer {

   private Buffer buf = new Buffer( );


   public void produce( int item ) {

      buf.addElement( item );     //Adds element to buffer

   }

   public int consume( ) {

      buf.removeElement( item );    //Removes element from
buffer

   }
}
```

Complete this code and implement a semaphore class allowing elements to be added and removed from the buffer concurrently. You must ensure that the code supports both competitive and cooperative synchronisation. A brief description of how the semaphore class works should also be provided.

[8 marks]

Total 20 marks

Question 3

3.a Several operating systems, such as that used on the early Apple Macintosh computers, have used cooperative multitasking. Briefly outline:

- What we mean by multitasking in operating systems.
- How cooperative multitasking is achieved.
- The key assumption that underpins cooperative multitasking and the main problem seen in such systems that stems from this.

[5 marks]

3.b Most modern operating systems adopt a different approach to multitasking.

- Identify this approach.
- Describe how commonly available hardware is used to operate this scheme.
- Explain why, particularly when compared to cooperative multitasking, this approach complicates task switching.

[3 marks]

3.c The following table shows scheduling information for a set of five processes P1-P5: it shows the arrival time of each process (i.e. the time at which the process is first passed to the scheduler) and the processing or burst time (i.e. the time needed to complete the execution of the process).

Process	P1	P2	P3	P4	P5
Arrival Time (ms)	0	5	20	25	45
Processing Time (ms)	15	5	30	10	5

State what we mean by the *average waiting time* and *average turnaround time* for a set of processes and calculate these values for the above processes and the following schedulers.

i. First Come First Served

ii. Round Robin with a time quantum of 10ms **[9 marks]**

3.d The basic round-robin scheme is often extended to support different levels of priority, with higher-priority processes always being chosen in preference to lower-priority processes.

Providing clear justification for your choice, offer a scenario that illustrates why such priority schedulers might periodically pick lower-priority processes despite higher-priority processes being available to run.

[3 marks]

Total 20 marks

Question 4

4.a Two processes each open a file for writing, process 1 opens file A, and process 2 file B. These two processes are scheduled such that process 1 writes four blocks, process 2 three blocks, process 1 two blocks, and process 2 another four blocks.

- Explain what we mean by a FAT
- Draw a diagram that shows how an initially empty FAT would look after the above operations have completed.
- What information, in addition to the basic FAT, is stored in order to enable access to the above files?

[5 marks]

4.b Outline two advantages indexed approaches have over FAT based schemes, and why 'combined' index-based implementations have been popular in operating systems such as Unix.

[3 marks]

4.c A filesystem using a continuous memory allocation scheme has the state shown in the following diagram. The regions containing sizes represent free space available for allocation.

Produce a sequence of similar diagrams for **each of the three** allocation strategies: *first fit*, *best fit* and *worst fit*, that show how the above state would evolve given a sequence of requests for 420K, 450K, 450K, 180K, 600K.

For each step in the process your diagrams **must** clearly show the size of each memory region **and** whether it is free or allocated.

[8 marks]

4.d A number of modern filesystems have journals incorporated in their design. Explain what we mean by this and give two advantages that journal based systems have over more traditional designs. What are the key requirements for operations issued to the journal?

[4 marks]

Total 20 marks

--- End of Paper ---