Candidate Number

G6059

THE UNIVERSITY OF SUSSEX

BSc and MComp SECOND YEAR EXAMINATION
May/June 2017 (A2)

Operating Systems

Assessment Period: May/June 2017 (A2)

DO NOT TURN OVER UNTIL INSTRUCTED TO BY THE LEAD INVIGILATOR

Candidates should answer TWO questions out of THREE. If all three questions are attempted only the first two answers will be marked.

The time allowed is TWO hours.

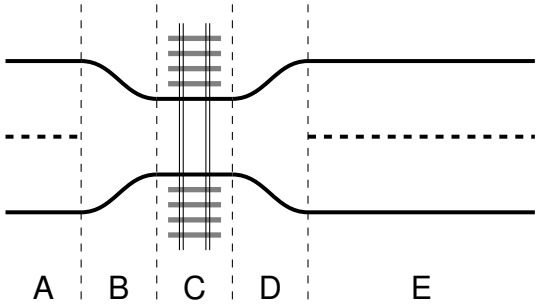
Each question is worth 50 marks.

At the end of the examination the question paper and any answer books/answer sheets, used or unused, will be collected from you before you leave the examination room.

1. Process synchronisation.

   (a) Name three different process synchronisation mechanisms, explain how they work, and discuss their advantages and disadvantages. [15 marks]

   (b) Implement a simulation of the following road/railway traffic scenario. [35 marks]



Explanations:

- The road has five sections: A, B, C, D and E.
- A car driving from left to right is simulated by an instance of a thread `CarLeftRight`. A car driving from right to left is simulated by an instance of a thread `CarRightLeft`. You are given the methods `void A()`, `void B()`, `void C()`, `void D()`, `void E()` that simulate a car driving through the corresponding section of the road.
- A train is simulated by an instance of a thread `Train`. You are given the methods `void approach()`, `void cross()`, `void continue()` that simulate a train approaching the road, crossing it, and continuing its way after the crossing, respectively.
- Your implementation must enforce the following rules:
  - There must not be any collisions between cars and trains.
  - Section A can hold one car in each direction.
  - Section E can hold `K` cars in each direction.
  - Sections B, C and D can only hold a single car (regardless of the direction).
  - Only a single train can cross the road at a time.
  - Trains have priority over cars.
  - A car must not block the railway or deadlock while trying to pass.
  - You should maximise the number of cars driving on the road and the number of trains crossing the road.

Consider the code template on the next page. You are expected to implement four code snippets (10–15 lines each). These code snippets replace the comments `// my code` n in the template ($n \in \{1, 2, 3, 4\}$). You are not required to copy the template, but clearly label the start of each of your code snippets with the corresponding number n.

```
1  public class TrafficSimulation {
2
3    static final int K=50;
4    // initialize semaphores
5    // my code 1
6
7    class CarLeftRight implements Runnable {
8      void run() {
9        // my code 2
10     }
11   }
12
13   class CarRightLeft implements Runnable {
14     void run() {
15       // my code 3
16     }
17   }
18
19   class Train implements Runnable {
20     void run() {
21       // my code 4
22     }
23   }
24
25   public static void main(String[] args) {
26     // creates and starts threads...
27     // (no implementation required)
28   }
29 }
```


The following class might be useful:

Class Semaphore
`Semaphore(int permits, boolean fair)` Creates a Semaphore with the given number of permits and the given fairness setting (fair=true).
`void acquire()` Acquires a permit from this semaphore, blocking until one is available.
`void release()` Releases a permit, returning it to the semaphore.

2. Scheduling, processes and threads.

   (a) Name five goals pursued by scheduling algorithms, explain them and give application examples for each of them to illustrate their importance. [15 marks]

   (b) Name three events that can cause a process switch. Give a concrete example for each of these events. [6 marks]

   (c) Which operations are done by an operating system to perform a process switch? [10 marks]

   (d) What is a microkernel? Which services does it provide? Discuss why you would (not) use a microkernel. [5 marks]

   (e) Give an example of an application scenario where the use of threads is preferred to the use of processes. Give an example of an application scenario where the use of processes is preferred to the use of threads. In both cases, justify your choice. [10 marks]

   (f) Consider the following piece of C code:


```
1 int main(int argc, char** argv) {
2   fork();
3   fork();
4   return 0;
5 }
```


   How many child processes are created upon execution of this program? Justify your answer. [4 marks]

3. Memory management, file systems and I/O.

   (a) We will compare the number of page faults for the three page replacement strategies OPT, FIFO and LRU. For each of the three strategies fill in a table such as the one given as a template below.

	A	B	C	D	B	E	D	F	C	E	B	F	A	F	E	D	F	A
0																		
1																		
2																		
3																		
PF																		

   There are six pages: A, B, C, D, E, F. The header row contains the accesses to pages in the order from left to right. There are four page frames (the four rows 0, 1, 2, 3). The row PF is to mark a page fault. Discuss why the result confirms your expectations or why it does not. [30 marks]

   (b) What is a block in a file system? Name three techniques for block allocation in a file system, and explain advantages and drawbacks. [16 marks]

   (c) What is DMA? What is its advantage? [4 marks]

End of Paper