# ATHLONE INSTITUTE OF TECHNOLOGY

# SCHOOL OF ENGINEERING

# SEMESTER 1 EXAMINATIONS 2015

# December Session

**BACHELOR OF SCIENCE IN SOFTWARE DESIGN (CLOUD/GAME DEVELOPMENT)**

**YEAR 3**

**OPERATING SYSTEMS & CONCURRENCY**

| | |
|---|---|
| **External Examiner(s):** | **Dr Chris Exton**<br>**Mr Jerh O'Connor** |
| **Internal Examiner(s):** | **Dr Sheila Fallon** |

**Instructions to candidates:**
Read all questions carefully.
All questions carry equal marks.
Answer **Three** out of **Four** questions.

**Time Allowed: 2 Hrs**

**No. of pages including cover sheet: 3**

Q.1. (a) Define the following terms: process, volatile environment, and context switching.

(6 marks)

(b) Explain the Fetch-Execute cycle. Your answer should include explanations of the following terms: PC and CIR

(6 marks)

(c) Briefly explain the difference between process scheduling and dispatching. Give two classifications of processes and explain how a Processing Scheduling algorithm should treat the two types of process.

(8 marks)

**[20 marks]**

Q.2. (a) Using short sections of java code, illustrate how to create and start a number of named Threads. The threads should loop 20 times printing out their name each time.

(5 marks)

(b) Draw a diagram showing all possible sequences of states through which a thread may pass. With reference to the thread you wrote in part (a) above explain why each state transition may occur

(8 marks)

(c) What is a Critical Section and what is mutual exclusion? Give an example of hardware support provided by processors to enable the implementation of mutual exclusion. Explain why this support is so useful

(7 marks)

**[20 marks]**

Q.3. (a) Explain the Producer Consumer design pattern. Describe **four** benefits of this design pattern

(5 marks)

(b) Using short sections of java code illustrate

(i) how a producer thread creates items and inserts them into an instance of java.util.concurrent.BlockingQueue.
(ii) how a consumer thread removes items from the BlockingQueue
(iii) a main() method that creates and starts a producer thread **and** a consumer thread.

(10 marks)

(c) With regard to a concurrent application explain the term liveness. Briefly explain three common liveness issues for a concurrent application.

(5 marks)

**[20 marks]**

Q.4.(a) Explain how the java.util.concurrent package separates task submission and task execution. In particular what are the relevances of the interfaces Executor and ExecutorService?

(6 marks)

(b) How do you obtain ExecutorService objects that implement the following ThreadPool strategies

- FixedThreadPool
- CachedThreadPool

What strategies do these ThreadPools implement?

(6 marks)

(c) Explain how “Callable”s and “Future”s are used to execute tasks when we want future access to the results of the task.

(8 marks)

**[20 marks]**