GRIFFITH COLLEGE DUBLIN

QUALITY AND QUALIFICATIONS IRELAND

EXAMINATION

POSTGRADUATE DIPLOMA IN SCIENCE IN COMPUTING

PARALLEL AND DISTRIBUTED PROGRAMMING
Module Code: PGDC-PDP

MASTER OF SCIENCE IN COMPUTING

PARALLEL AND DISTRIBUTED PROGRAMMING
Module code: MSCC-PDP

MASTER OF SCIENCE IN BIG DATA MANAGEMENT AND ANALYTICS

PARALLEL AND DISTRIBUTED PROGRAMMING
Module code: MSCBD-PDP

Lecturer(s):	**Osama Abushama**
External Examiner(s):	**Dr Mubashir Husain Rehmani** **Dr William Clifford**
Date: 29th May 2024	**Time: 9.45-12.45**

THIS PAPER CONSISTS OF FIVE QUESTIONS
FOUR QUESTIONS TO BE ATTEMPTED
ALL QUESTIONS CARRY EQUAL MARKS

QUESTION 1

(a) Implement a ForkJoinPool recursive class to count the negative numbers in a huge array, do not write the main class.

(10 marks)

(b) Repeat part 1 using a Callable interface. do not write the main class.

(10 marks)

(c) What is the difference between waiting() and timed_waiting() in thread states?

(5 marks)

Total (25 marks)

QUESTION 2

A Fibonacci sequence generator is a recursive algorithm has the formula:

```
F(0) = 0
F(1) = 1
F(n) = F(n-1) + F(n-2) for n > 1
```

(a) Explain why this algorithm is difficult to parallelise.

(5 marks)

(b) Write a thread class that shuffle the elements in an array. Optimizes the performance of the thread performance and ensure no deadlock.

(15 marks)

(c) Can a constructor be synchronized? Explain why.

(5 marks)

Total (25 marks)

QUESTION 3

(a) Implement a Last-In-First-Out stack, intended for use in a concurrent environment. Only methods `push` and `pop` are needed. write it using the `Lock` class so that the shared methods are safe for concurrent access.

(10 marks)

(b) Explain in OpenMP what a schedule clause means and how it can be used?

(10 marks)

(c) Can you implement the schedule at run time, show how?

(5 marks)

Total (25 marks)

QUESTION 4

(a) Write a parallel OpenMP C program to perform matrix-matrix addition of a n*m size. Assume the matrices are initialized.

(10 marks)

(b) Write a parallel MPI C program to perform matrix-matrix addition of a n*m size. Assume the matrices are initialized. Use Scatter and Gather API's.

(15 marks)

Total (25 marks)

QUESTION 5

(a) Describe data parallelism and functional parallelism.

(5 marks)

(b) Implement a CountDownLatch using ReentrantLock.

(10 marks)

(c) Discuss possible optimisations to improve the performance of your countdown latch implementation.

(5 marks)

(d) Implement a barrier in Java using a CountDown latch. Make sure that your barrier is reusable.

(5 marks)

Total (25 marks)