RHODES UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

EXAMINATION: JUNE 2023

COMPUTER SCIENCE HONOURS
PAPER 4
DISTRIBUTED AND PARALLEL PROCESSING

Internal Examiner: Prof GC Wells
External Examiner: Prof I Sanders

MARKS: 120 marks
DURATION: 2 hours

GENERAL INSTRUCTIONS TO CANDIDATES

1. This paper consists of **3 pages** and **9 questions**. **Please ensure that you have a complete paper.**
2. Answer ALL questions.
3. The use of calculators is permitted in the examination, however, make sure that you show all workings.
4. The Oxford Concise English Dictionary **may** be used during this examination.

DISTRIBUTED AND PARALLEL PROCESSING [120 MARKS]

QUESTION 1: Hardware [14 marks]

During the course we discussed five different hardware approaches used in parallel and distributed processing: *multicore processors*, *symmetric multiprocessors*, *heterogeneous processors*, *clusters*, and *supercomputers*. Describe and discuss any two of these five approaches. In terms of the Candidate Type Architecture (CTA), what is the critical performance factor that distinguishes these systems?

QUESTION 2: Patterns [12 marks]

During the course we discussed a number of parallel programming patterns, including the *Divide and Conquer Pattern*, the *Pipeline Pattern* and the *Replicated-Worker Pattern*. Describe and discuss any **two** of these three patterns.

QUESTION 3: Theory [10 marks]

Message passing has been described as the "assembly language" of parallel programming. Discuss this view and state, with reasons, whether it is justified in your opinion.

QUESTION 4: Performance [12 marks]

A scientific research centre is considering purchasing a new cluster system. They have narrowed down their options to two systems:

1. A 20-processor system.
2. A 30-processor system, where each processor is 20% slower than those in 1.

If the cluster is to be used for applications where 10% of the application must be executed sequentially, what would be the expected speedup of the two designs compared to a single processor system, using the same processor as cluster 1? (Show your working).

QUESTION 5: Java Concurrency [15 marks]

EITHER:

The Java Concurrency Utilities package (`java.util.concurrent`) contains several *synchronizers*, including *semaphores*, *barriers* and *latches*. Discuss how each of these mechanisms provides synchronisation, giving a brief example of how they might be used.

OR:

Discuss the *executor* services that are provided by the Java Concurrency Utilities package (`java.util.concurrent`). What is the primary purpose of these services? You should include a discussion of the concept of a "*future*", and its implementation in the Concurrency Utilities package.

QUESTION 6: Multiprocessing [16 marks]

Describe how you would use the UNIX System V IPC *semaphore* and *shared memory* facilities in order to provide synchronous communication of simple messages between two processes. Your

solution should allow a sender to place a message in a shared memory segment and then wait until the receiver has read it before proceeding. You need not give accurate code for your answer, but should mention the important steps in creating and using the IPC facilities that you need to use.

QUESTION 7: CSP/JCSP **[4+10 = 14 marks]**

a) What does the following CSP code do?

```
* [  x : integer ; A ? x  →  C ! x * x
   □ x : integer ; B ? x  →  C ! x * x ]
```

b) Convert this code into JCSP (you do not need to give a syntactically perfect answer). Explain any significant differences between the CSP version and the JCSP approach.

QUESTION 8: RMI **[15 marks]**

Explain in detail how you would use Java's RMI in order to implement a simple location-based service. A client program should send its current location (represented as a string) and a radius of interest (an integer) to the service, which should respond with a list of local services (represented as strings). Define the acronym RMI, and be sure to explain all the concepts that are involved, such as marshalling/unmarshalling, stub, skeleton, registry, etc.

QUESTION 9: CSP Metalanguage **[6+6 = 12 marks]**

Each customer of a bank first opens an account. He or she then makes any number of deposits and withdrawals, and finally terminates his/her account. Let us initially ignore the amount of each deposit or withdrawal, and not worry whether the account is in credit or debit. The alphabet of the account is therefore:

α ACC = {open, deposit, withdraw, terminate}

a) Construct the process ACC.

b) The bank shuts at 3.00p.m. every day, and does not reopen again until 9.00a.m. the following morning. Only deposits are possible during the interval when the bank is closed. Introduce two new events {shut, reopen}, and write a process that, when added in parallel with ACC, prevents any other events from happening between these two. The alphabet of the new process should be: α ACC $\cup$ {shut, reopen}

END OF THE EXAMINATION