This part for lecturer's grading use only	CS4231

NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

AY2014/2015, SEMESTER 2

FINAL ASSESSMENT FOR

CS4231: Parallel and Distributed Algorithms

27 APR 2015 **Time Allowed: 2 Hours**

INSTRUCTIONS TO CANDIDATES

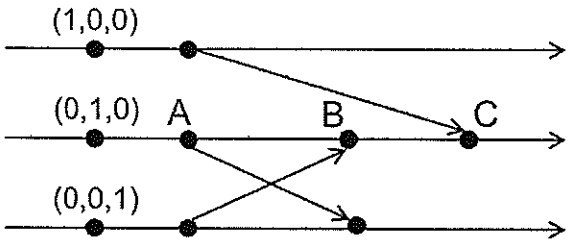
1. This assessment paper contains **ELEVEN** problems and **EIGHT** printed pages, including this page.
2. Students are required to answer **ALL** questions.
3. All questions should be answered within the space provided in this booklet.
4. This is an **Open Book** assessment. Electronic calculators are **not** allowed.
5. Read each question carefully before you answer. If you misunderstand the question and answer a different question, you get **zero credit** for that question.
6. It is your job to show the correctness of each statement you make. Otherwise points will be deducted **even if the lecturer cannot prove you are wrong**. (This is a general rule for all theory subjects – otherwise you can claim P=NP and nobody can prove you are wrong.)
7. Write your matriculation number below. Do not write your name.

MATRICULATION NO:

DO NOT WRITE BELOW THIS LINE

Problem 1 and 2	
Problem 3, 4, and 5	
Problem 6, 7, and 8	
Problem 9	
Problem 10	
Problem 11	
Total	

Problem 1. (6 marks)
Give the vector clock values of events A, B, and C in the following execution, as generated by the standard vector clock protocol. The vector clock value of the first event on each process has already been indicated in the figure.

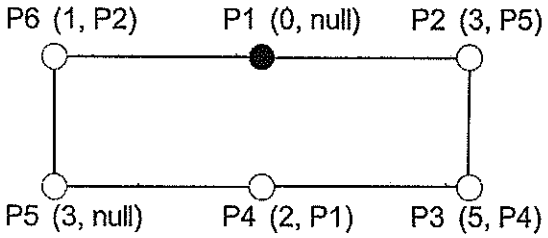


The vector clock value of event A is: _____

The vector clock value of event B is: _____

The vector clock value of event C is: _____

Problem 2. (8 marks)
Consider the self-stabilizing spanning tree algorithm. Assume that the system starts from the following state, where P1 is the root:



In the figure, the tuple beside each node indicates the current values of the `dist` variable and the `parent` variable on that node.

Starting from the above state in the figure, what will the tuple for P3 be after P3 takes an action (i.e., executes its code once), assuming that other nodes do not take actions?

P3's tuple will become: _____

Starting from the above state in the figure, what will the tuple for P5 be after P5 takes an action (i.e., executes its code once), assuming that other nodes do not take actions?

P5's tuple will become: _____

After the system eventually stabilizes (i.e., enters into a legal state), what will the tuple for P3 be?

P3's tuple will eventually be: _____

After the system eventually stabilizes (i.e., enters into a legal state), what will the tuple for P5 be?

P5's tuple will eventually be: _____

Problem 3. (4 marks)
For mutual exclusion algorithms, briefly explain the difference between the Progress property and No-Starvation property, and briefly explain why Progress does not necessarily imply No-Starvation.

Problem 4. (6 marks)
Using Java monitor, Alice designs the following protocol to ensure that no more than one process can access a certain file simultaneously. Every process is supposed to use this protocol for accessing the file.

```
// locked is a shared Boolean object initialized to be false;
while (locked == true) { synchronize (locked) { locked.wait(); } }
locked = true;
// allowed to access file here
locked = false;
synchronize (locked) { locked.notifyAll(); }
```

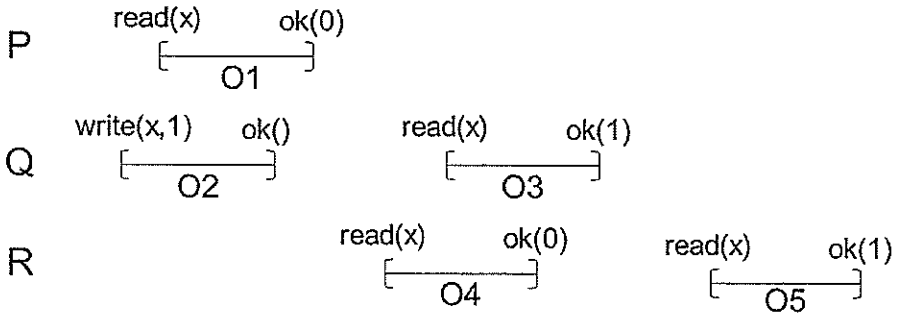
What is wrong with this code? Explain.

Problem 5. (4 marks)
Does causal order message delivery necessarily implies FIFO order message delivery? Briefly justify your answer.

Problem 6. (4 marks)
Briefly explain the key difference between the guarantees of logical clocks and the guarantees of vector clocks.

Problem 7. (4 marks)
Briefly explain the notion of deadlocks in synchronization code and how deadlocks can be avoided.

Problem 8. (6 marks)
Consider the following concurrent history, where the shared integer variable x has an initial value 0. Enumerate 3 different legal sequential histories that preserve process order and are equivalent to this parallel history.



Sequential history #1 (please write as a sequence of the 5 operations): _____

Sequential history #2 (please write as a sequence of the 5 operations): _____

Sequential history #3 (please write as a sequence of the 5 operations): _____

Problem 9. (8 marks)
Consider the leader election problem on a ring with N nodes, where N is known and $N \geq 3$. The timing model is synchronous. The nodes on the ring all have ids. Among these N nodes, $N-1$ nodes have unique and distinct ids. The last node has an id that is the same as the id of one of the $N-1$ nodes. (Intuitively, there are always two "twins" on the ring and we don't know beforehand which two nodes are the twins.) Design a deterministic leader election protocol for such a setting. Your answer should have the following **3 parts**: i) **provide high-level intuition** of your protocol, ii) **provide an unambiguous description** of your protocol, and iii) **prove rigorously** that your protocol will elect one and only one leader.

Problem 10. (6 marks)
Consider the distributed consensus problem where the timing model is **asynchronous**, the communication channels are reliable, and the nodes may experience crash failures. There are total N nodes and the maximum number of nodes that may experience crash failures is 1. Every node can directly communicate with every other node. Every node has a binary initial value. Furthermore, there exists some value k such that among these N nodes, it is impossible for exactly k nodes to have initial values of 0 and the remaining N-k nodes to have initial values of 1. Both N and k are known to you, and k satisfies $1 \leq k \leq N-1$. Do you feel it is possible to solve distributed consensus under this setting? If yes, **indicate "Yes"** and then **give a protocol together with a formal proof** showing that your protocol satisfies termination, validity, and agreement. If no, **indicate "No"** and then **give a formal impossibility proof.**

First, indicate "Yes"/"No": _____

Next, please provide a detailed answer according to the above instructions:

Problem 11. (4 marks)
Consider the distributed consensus problem where the timing model is **synchronous**, the communication channels are reliable, and nodes may experience crash failures. There are total N nodes and the maximum number of nodes that may experience crash failures is $N-1$. Every node can directly communicate with every other node. Under this setting, in class, we learned the following protocol to achieve consensus in N rounds: In each round, every node sends to all other nodes all the initial values it has seen so far. (A node does not send message to itself.) At the end of round N, every node decides on the maximum value seen.

Alice proposes to improve the protocol in the following way, to allow nodes to decide as early as possible. For all i where $1 \leq i \leq N$, at the end of the ith round, every node X checks whether it receives at least $N-i$ messages from the other nodes in that round. If yes, then node X immediately decides on the maximum value it has seen. Once decided, node X will not decide again in future rounds, but it will continue running the protocol (i.e., sending and receiving values) until round N, to help other nodes to decide.

Alice claims that her protocol satisfies the following properties:
-- Early decision: For all f where $0 \leq f \leq N-1$, if there are actually only f failures during the protocol's execution (note that f can be much smaller than $N-1$ and that f is not known to the protocol beforehand), then every node that has not crashed by the end of round $f+1$ must decide by the end of round $f+1$.
-- Validity: If all nodes have the same initial value, then that initial value should be the only possible decision value.
-- Agreement: Every node that decides (regardless of whether the node crashes later) must decide on the same value.

Do you agree with Alice's claim? If you agree with Alice, **indicate "Yes"** and then **formally prove** that Alice's protocol satisfies all the properties she claims. If you do not agree with Alice, **indicate "No"** and give a clear **counter-example** explaining that some property can be violated. Only indicating "Yes"/"No" will not earn you any mark.

First, do you agree with Alice? (Yes/No) _____

Next, please provide a detailed answer according to the above instructions:

(more space is available on the next page for you to write your solution...)

(The space below is for you to continue your solution for Problem 11. Do NOT use the space to write the solution for any other problem.)

END-OF-PAPER