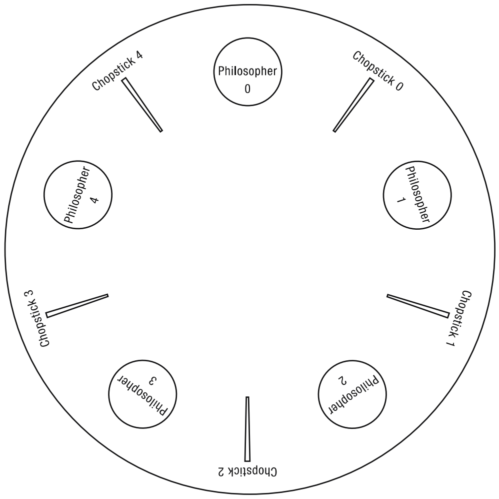
CPSC 3220 – Quiz #6 March 7, 2018 Name: ___________________

1.(2 points) Consider the variation of the Dining Philosophers problem shown to the right, where all unused chopsticks are placed in the center of the table and any philosopher can eat with any two chopsticks.

One way to prevent deadlock in this system is to provide sufficient resources. For a system with n philosophers, what is the minimum number of chopsticks that ensures deadlock freedom? Why?



n+1 suffices and is the minimum.

It suffices because in any state, if any philosopher wishes to eat, some philosopher can finish eating, returning chopsticks to the pool, and allowing any other philosopher to eat

2. (4 points) List the four conditions necessary for deadlock to occur:

Limited Resources - Resources are exclusively held

No Preemption - Resources cannot be taken away

Multiple Independent Requests - Wait while holding

Circular Waiting – Processes can wait while holding resources on processes holding other resources.

3. (2 points) Suppose you build a system using a staged architecture with some fixed number of threads operating in each stage. Assuming each stage is individually deadlock free, describe a way to guarantee that your system as a whole cannot deadlock. You only need to break one of the four necessary conditions for deadlock.

(1) no limited resources -

Make the queues infinitely large so that a stage never blocks when sending a message.

(2) preemption is allowed -

When a pipe fills up, a stage holding a lock is restated.

(3) no multiple requests -

Each stage can only request one lock.

(4) no circular waiting -

Arrange the stages into a directed graph so that there cannot be a cycle of stages waiting for each other.

Continued on Back

Banker's Algorithm Usage Table

Process	Max Needed	Allocated	Remaining Need
A	7	3	4
B	3	1	2
C	4	2	2

Total Units Available = 8

This is a safe state.

4. (1 point) Which Process(es) cannot be given any more resources?

A

5. (1 point) Which Process(es) can be given more resources?

B and C

6. (1 point) Give a sequence of requests and releases that recover all resources.

B -> C -> A   or   C -> B -> A

7. (1 point) Will the sequence of requests work?

B Request (1), C Request (1), B Request (1),

No