Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Operating Systems (CS550)**

**Exam 2**

**Part I (20 points).**

Circle either true or false to indicate that the following statements are true or false. Evaluate “truth” in terms of the material covered in this course. The statements below are intended to have an unambiguous answer.

1. T F There are exactly three conditions necessary for deadlock.

2. T F In CPU scheduling, the scheduler’s main purpose is to shift processes

between the blocked and ready states.

3. T F Mutual exclustion allows for two processes to access a shared

resource while all others wait.

4. T F To solve the dining philosophers problem, you must both disallow

circular wait and disallow hold and wait.

5. T F In dynamic scheduling process and thread priorities never change.

6. T F MPI\_Irecv is a blocking function.

7. T F Round robin scheduling is a non-preemptive policy that makes use of

 time slices that are often longer than the average CPU burst time.

8. T F A monitor only provides mutual exclusion at the level of an

executable statement.

9. T F When performing atomic operations, one should never use

transaction logging and checkpoint restart.

10. T F The enqueuer is used to select the next process to enter the run state.

**Part II (80 points).**

Read instructions and questions carefully.

1. The CPU scheduler has three main functions and three parts to carry out these functions. List the three parts, describing the function of each (6 points).
2. Provide pseudocode for the reader priority variant of the readers writers problem. Assume readers and writers are represented by threads, and both are forked (started) within a driver class (10 points).
3. Explain the difference between a semaphore and a monitor (4 points).
4. Given 5 processes that start at the same time, having CPU burst times of 7, 10, 20, 68, and 32 milliseconds, draw a Gantt chart for those processes using the First Come, First Served scheduling policy (5 points).
5. Assuming the same processes as in problem 4 and sixth process arriving at time 30ms, draw a Gantt chart for those processes assuming a Round Robin scheduling policy with a time slice of 20ms (10 points).
6. Compute the average process wait time for problems 4 and 5. Which of the two policies provides better wait time performance based upon the problems worked? Which policy would be more appropriate for an interactive system? Why? (10 points)
7. Explain the difference between a blocking and non-blocking receive in MPI. That is, explain the difference between MPI\_Recv and MPI\_Irecv (5 points).
8. Apply the Banker’s algorithm to the following situation. There are four resource types with the total number of system resources expressed as the tuple (2, 3, 2, 1). There are three processes with current allocations of (1, 0, 0, 1), (1, 1, 0, 0), and (0, 0, 2, 0). The maximum claim of each process is (2, 0, 0, 1), (1, 3, 0, 0), and (1, 3, 2, 1). Assume each process can complete its work once it receives its maximum claim. Determine if a safe state can be achieved, being sure to show all work (10 points).
9. Explain how deadlock detection and recovery differs from deadlock avoidance (10 points).
10. Describe the difference between binary and counting semaphores, and explain why both are needed in the Producer-Consumer problem (10 points).