CSE 451/851 Assignment #4

Notes:

- There are **Five** questions in this homework.
- Submit your response to Canvas.
- Latex or Word-based submissions are required for written questions. **Scanned handwritten submissions will not accepted.**
- Please answer each question in detail.

1. **[10 points]** Suppose the following two processes, foo and bar are executed concurrently and share the semaphore variables S and R (each initialized to 1) and the integer variable x (initialized to 0).

```
void foo( ) {
   do {
     semWait(S);
     semWait(R);
     x++;
     semSignal(S);
     SemSignal(R);
   } while (1);
}
```

```
void bar( ) {
do {
    semWait(R);
    semWait(S);
    x--;
    semSignal(S);
    SemSignal(R);
} while (1);
}
```

Can the concurrent execution of these two processes result in one or both being blocked forever? If yes, give an execution sequence in which one or both are blocked forever. If no, explain.

2. **[30 points]** Consider the readers/writers problem, where (1) any number of readers may simultaneously read the file, (2) only one writer at a time may write to the file, and (3) if a writer is writing to the file, no reader may read. Furthermore, let us assume that readers have priority, where as long as there are readers, a writer should wait, (2) as long as one writer is accessing the shared the file, no other writer or reader may access it. We also require that if a writer is writing and there exists multiple readers and writers waiting, then priority is given to readers.
You are to develop a solution to the problem using a monitor. Use the following procedures and condition variables. You are free to use additional integer variables, where you see fit.

Monitor: reader_writer;
cond canRead, canWrite;

Declaration:

startread():

endread():

startwrite():

endwrite():

Initialization:

3. **[30 points]** Show the schedule using FCFS, non-preemptive priority (a smaller priority number implies higher priority) and round robin with quantum 30ms for the given workload.

Process	Burst Time	Priority	Arrival Time
P1	50 ms	4	0 ms
P2	20 ms	1	20 ms
P3	100 ms	3	40 ms
P4	40 ms	2	60 ms

a) Draw Gantt chart for all three scheduling policies.
b) What is the average waiting time of the above scheduling policies?
b) What is the average turnaround time of the above scheduling policies?

4. **[10 points]** Consider a variant of the RR scheduling algorithm where the entries in the ready queue are pointers to the PCBs.

a. What would be the effect of putting two pointers to the same process in the ready queue?
b. What would be the major advantage of this scheme?
c. How could you modify the basic RR algorithm to achieve the same effect without the duplicate pointers?

5. **[40 points]** Given the following state for the Banker's Algorithm.

5 processes P0 through P4
4 resource types: A (12 instances); B (12 instances); C (8 instances); D (10 instances)
Snapshot at time T0:

	Allocation	*Max*	*Available*
	A B C D	*A B C D*	*A B C D*
P_0	2 0 0 1	4 2 1 2	
P_1	3 1 2 1	5 2 5 2	
P_2	2 1 0 3	2 3 1 6	
P_3	1 3 1 2	1 4 2 4	
P_4	1 4 3 2	3 6 6 5	

a) Calculate the available array.
b) Calculate the Need matrix.
c) Illustrate that the system is in a safe state by demonstrating an order in which the processes may complete.
d) If a request from process P1 arrives for (1, 1, 0, 0), can the request be granted immediately?
e) If a request from process P4 arrives for (0, 0, 2, 0), can the request be granted immediately?