CS 4760	Operating Systems	Test 2
Name:	Spring 2021	Max Pts: 68

Important: This is an open book test. You can use any books, notes, or paper. If there is a syntax error in any program segment, just write it down and you will get full credit for the problem.

1. [6 pt] What is a critical performance requirement in an interactive OS?
2. [6 pt] Consider a variant of the round-robin scheduling algorithm where the entries in the ready queue are pointers to PCBs. What would be the effect of putting two pointers to the same process in the ready queue? Is there any downside?
3. [3+3+3+2 pt] Processes p_1 and p_2 with period 8 and 10 and total CPU time as 3 and 6, respectively, arrive at time 0.
   (a) Which process will have control of CPU at time 0, 5, 10, 15, 20, and 25 using RM algorithm?
   (b) How about EDF algorithm at the same times?
   (c) What can you say about a guarantee for a feasible schedule with each of those two algorithms?
   (d) What is the first deadline for p_1 and p_2?
4. [6 pt] Why is spin lock not useful on a Linux OS running on a uniprocessor?
5. [6 pt] What is a good way to prevent circular wait condition to handle deadlocks? Can your method still lead to a deadlock?
6. [10 pt] Consider a system with the following set of processes and states:

   $$P = \{p_0, p_1, p_2\},\ S = \{s_0, s_1, s_2, s_3\}$$

   State changes due to processes are:

   $$\begin{array}{lll} p_0(s_0) = \{s_0, s_1, s_2\} & p_1(s_0) = \emptyset & p_2(s_0) = \emptyset \\ p_0(s_1) = \emptyset & p_1(s_1) = \{s_1\} & p_2(s_1) = \{s_0, s_1\} \\ p_0(s_2) = \emptyset & p_1(s_2) = \{s_0\} & p_2(s_2) = \{s_2\} \\ p_0(s_3) = \{s_3\} & p_1(s_3) = \{s_0\} & p_2(s_3) = \{s_0, s_1\} \end{array}$$

   Is the system safe? Is it deadlocked? Is there a knot in the system? You may have to draw the state transition diagram to answer the question but you don't need to submit the diagram.
7. [6 pt] Why do we need the capability to relocate processes?
8. [6 pt] What is the difference between compile-time binding and load-time binding?