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 the difference between turnaround time and response time?

2. [6 pt] In almost every operating system, the interrupts are handled by kernel. Yet, we saw in our class projects that you can handle interrupts within the code. In fact, it is recommended that shell scripts perform some cleanup when an interrupt is received. Do you see a problem with what we say and what we do? Explain your answer.

3. [6+6+2+2 pt] Two periodic real-time processes p1 and p2 with period 8 and 10 and total CPU time period 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 algorithm?

   (b) How about EDF algorithm?

   (c) What can you say about a guarantee for a feasible schedule with those two algorithms?

   (d) What is the first deadline for p1 and p2?

4. [6 pt] What are the advantages of early binding compared to late binding? What are the advantages of late binding compared to early binding?

5. [6 pt] Three processes share four resource units from the same resource class that can be reserved and released only one at a time. This can be done in any order, that is, we do not have total order imposed on requests. Each process needs a maximum of two units. Can we have a deadlock in the system? Explain your answer.

6. [10 pt] Assume a system with four resource types, $C = \langle 11, 10, 6, 2 \rangle$ (this is the total number of resources in the system, and not what is currently available), and the maximum claim table shown below.

Process	R_0	R_1	R_2	R_3
p_0	1	1	1	2
p_1	5	10	3	2
p_2	1	6	6	1
p_3	2	5	3	1
p_4	5	10	6	2

   The resource allocator is considering allocating resources according to the following table:

Process	R_0	R_1	R_2	R_3
p_0	1	0	0	0
p_1	4	7	0	1
p_2	0	3	3	0
p_3	2	0	0	0
p_4	1	0	3	0

Run the safety algorithm on this system to determine if this state is safe. If it is safe, give the sequence in which processes can be run. If it is unsafe, enumerate the processes that may get involved in a deadlock. Show your steps from the algorithm.

7. [6 pt] Does a knot in a process resource graph guarantee the existence of a deadlock? Explain your answer.

8. [6 pt] Explain the difference between a heap manager and a virtual memory manager.

9. [6 pt] Why do we need the capability to relocate processes at run time?