EE 4710 Exam 1. Name ________________

4 pages. Open Book. Open Notes. Closed Internet. Cheating will result in a score of 0.

1. (3 pts) What is a real-time system? A system whose correctness depends on both the logical results of computations and the time that those results are produced.

2. (2 pts) The author of the text uses the term "hard real-time" to describe (circle one)
   (a) tasks whose usefulness drops precipitously after their deadlines have passed.
   [✓ (b)] tasks that require rigorous validation of their timing constraints.
   (c) tasks that are periodic and have a maximum execution time.
   (d) tasks that contain minerals, such as potassium or calcium carbonate.

3. (2 pts) Jobs have 2 different types of dependencies that lead to precedence constraints. What are they? (circle two)
   [✓ (a)] Data dependencies
   (b) Resource dependencies
   (c) Temporal dependencies
   [✓ (d)] Control dependencies

4. (5 pt) In class, we listed 6 properties of a valid schedule. Name five.
   1. At most 1 job per processor at a time
   2. At most 1 processor per job at a time
   3. No job scheduled before its release time
   4. The amount of time allocated to a job equals its execution time
   5. All precedence constraints are met
   6. No preemption of non-preemptable jobs.

5. (2 pts) A job's completion time minus its release time is called response time.

6. (2 pt) ([✓ true]/false) If the sum of all job execution times is less than the smallest relative deadline, then a simple scheduling loop will produce a feasible schedule.

7. (2 pt) What conditions must be met for EDF to be optimal? (circle all that apply)
   [✓ (a)] tasks must be preemptable.
   (b) the relative deadline of each task must equal the task's period.
   [✓ (c)] the tasks must be scheduled on a single processor.
   (d) the scheduler must be clock-driven.

8. (7 pts) Give the real-time model parameter for each of the descriptions, below.

T_i task *i*
r_i release time for job or task *i*
d_i deadline for job or task *i*
e_i (maximum) execution time for job or task *i*
J_i job *i*
p_i period of task *i*
c_i completion time for job or task *i*
D_i relative deadline for task *i*

EE 4710 Exam 2. Name (7 pts)______________________

4 pages. Open Book. Open Notes. Closed Internet. Cheating will result in a score of 0.

Part 1. Short Answer.

1. (3 pts) What is priority inversion? Where a lower Priority task runs while a higher priority task waits

2. (4 pts) Which of the following approaches risks uncontrolled priority inversion? (Circle all that apply)
   (a) Using the Nonpreemptive Critical Section protocol (NPCS)
   [✓ (b)] Resource locking with semaphores
   (c) Using the Priority Inheritance Protocol
   (d) Using the Priority Ceiling Protocol

3. (3 pts) What is deadlock? A condition where 2 or more tasks mutually wait on each other

4. (4 pts) Which of the following approaches risks deadlock? (circle all that apply)
   (a) Using the Non-preemptive Critical Section protocol (NPCS)
   [✓ (b)] Resource locking with semaphores
   [✓ (c)] Using the Priority Inheritance Protocol
   (d) Using the Priority Ceiling Protocol

5. (3 pts) Suppose your real-time operating system supports resource locking only. Your system contains three resources, X, Y and Z, and each task needs access to two <u>or more</u> of them. Use what you learned from the dining philosopher's problem to describe a policy you would use to avoid deadlock.
   Task should take resources in order, say X, Y then Z

6. (6 pts) Three protocols are described below. State the name for each one.

   (a) Tasks run at assigned priorities, but jobs in T_i are not allowed to start until π_i has a higher priority than $\hat{\Pi}(t)$. Resources are always granted upon request.
   protocol name: Stack based Priority Ceiling protocol

   (b) Each task runs at its assigned priority until one task takes a resource, at which time that task runs at the highest possible priority until all resources are free.
   protocol name: Non-preemptive Critical section protocol

   (c) When T_i requests a resource, it is granted only if the resource is free and either (i) π_i is higher priority than $\hat{\Pi}(t)$, or (ii) the T_i holds resource X such that $\hat{\Pi}(t) = \Pi(X)$.
   protocol name: Priority Ceiling Protocol

7. (2 pts) Name one protocol that allows all tasks to share the same stack.
   Stack based priority ceiling protocol

8. (2 pts) ([✓ true]/false) A problem with the nonpreemptive critical section protocol is that one task can block another even though there is no resource conflict.