SE3910 Quiz 5 Name:

This is a closed-book, closed-computer, etc. quiz. Review all questions before you get started. ***Show all work. Box your final answer.***

1. (10 points) Rate Monotonic Analysis
	1. Fill in the priority and CPU usage (*ui*) columns of the chart below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task name | priority (1 highest) | *ei (ms)*execution time | *pi (ms)*period | *ui (%)*CPU time |
| tn – Read nerve value |  | 0.5 | 5 |  |
| ta – Update arm control |  | 0.5 | 2 |  |
| td – Refresh configuration display |  | 15 | 30 |  |

* 1. ***Determine*** the overall CPU usage.
	2. Suppose that the CPU usage was 85%. ***Circle one***: rate monotonic scheduling **is** / **is not** guaranteed to succeed. ***Explain*** your answer.
	3. Suppose that the CPU usage was 85%. ***Circle one***: It **is** / **is not** possible for rate monotonic analysis to succeed. ***Explain*** your answer.
	4. (7 points) Label all the *happens-before* relationships with arrows in and between the code running on the two threads below. Assume both threads are running with the same object. Assume x and y are both shared, only x is volatile, and that as the threads run, the write to x happens before the read.

void a () {
 y = 5;

 x = 7;
}
void b () {
 System.out.println("x:"+x);

 System.out.println("y:"+y);
}

* 1. (3 points) ***Circle one***: This code **is** / **is not** sequentially consistent. Explain your answer. (Reasoning directly about caching and reordering does not necessarily help with this.)