Chris Park
Lab 5 Questions

1) ARTK's pre-emptive multi-tasking only applies when a ready task has a higher priority than other tasks that are running. ARTK can avoid this deadlock by setting the task priority of all the philosophers to the same level.

2) Cooperative multi-tasking, where tasks can give up control voluntarily could cause this deadlock. A possible end result could occur where every philosopher would be waiting on a semaphore held by another philosopher, creating the deadlock.

3) Setting a time limit on waiting for a semaphore, and then releasing the resource and yielding to another task once that time limit expires could be used as a preventative measure for deadlock. In the philosophers example, this would mean that a fork could be picked up, and if that philosopher reached the waiting limit on trying to pick up another fork, it would just drop the fork it already had, return to thinking, and allow another philosopher to have the fork so it can attempt to eat. There might be certain algorithms that would cause starvation in this way...perhaps round robin, or some other similarly predictable scheduling algorithm. Introducing some randomness to task selection might help here, although the worst case that a philosopher never gets switched to could still exist.