

Assignment 1 ECSE 420

Elvric Trombert
260673394

October 3, 2018

-
2. (a) In our case deadlock can occur when the sleep time of Thread 1 is longer than than the sleep time of thread 2. Since Thread 1 would have acquired the first lock but will want to acquire the second lock. Which would have already been acquired by the second thread as it walk up earlier which will be waiting to acquire the first lock. Hence we are in a deadlock as neither thread is willing to give up there lock but cannot finish there execution without the other thread's lock.

(b) Hold-and-wait

Require a process to request all of its required resources at a time making it block otherwise only activating it when all the resources are available. This prohibits resources from being use optimally. Sometimes it is hard to know in advance all the resources that a process will need. The programmer will have to give all the possible resources required by the process note that these are possible not required.

No preemption

If a process holding resources is denied a further request that process must release all its unused resources and request them again.

Circular wait

Can be prevented by defining a linear ordering of resource type. If a process ahs been allocated resource of type R, then it may subsequently request only those resources of types following R in the ordering. So P1 holds are R1 so it can only request $R_i > 1$, P2 holds R2 so it can only request $R_i > 2$.