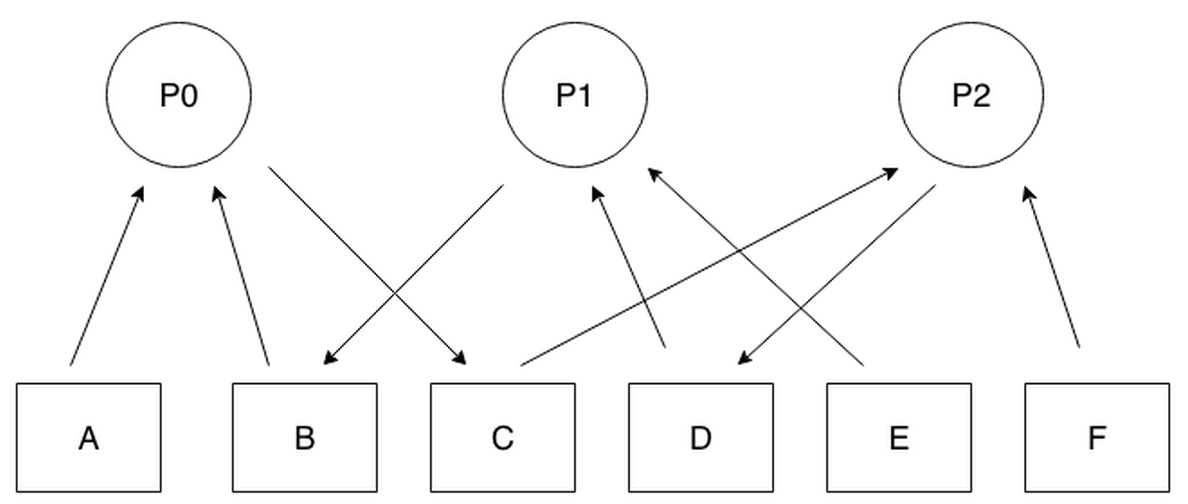
John Carroll  
COMP 3500 – Group Homework 2

  
a) Since all the process P0 is waiting for process P2 which itself is waiting for P1 which is waiting for P0, deadlock occurs due to cyclic waiting.

b)

Process P0-

get(B)

get(C)

get(A)

Process P1-

get(D)

get(E)

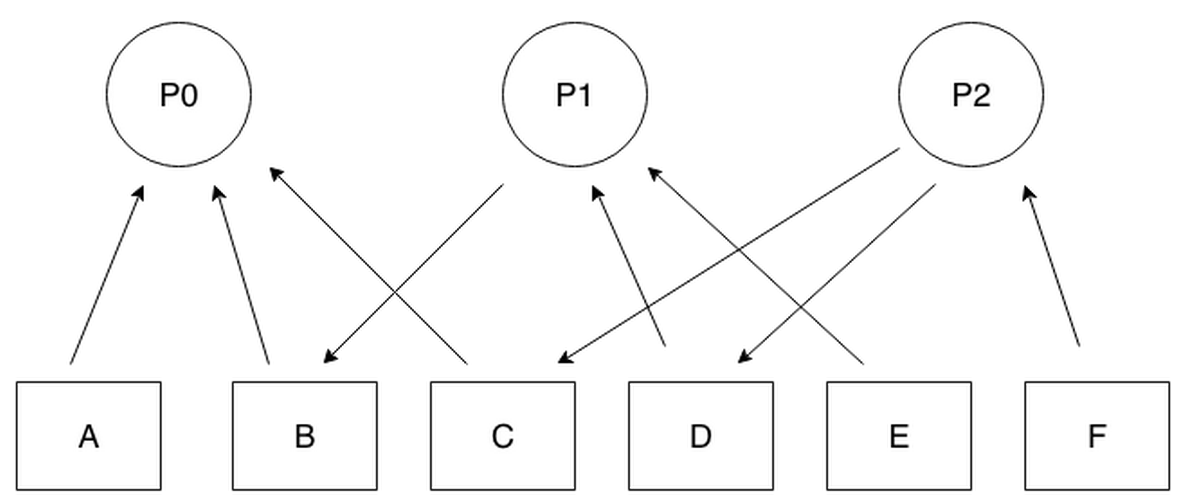
get(B)

Process P2-

get(F)

get(D)

get(C)



1. **Yes,** these processes can be blocked forever. **If statement 1 of bar() executes just after statement 1 of foo()** then both semaphores R and S would be 0 and would wait on statement 2 of bar() and foo().
2. Avoidance: Don't share resources across processes / mulitple threads

Prevention: When accessing shared resources, use a semaphore. If locking multiple semaphores, be sure to unlock in the reverse order of locking. Always be sure to handle errors within the critical sections so the semaphore is released under all conditions.

1. Yes, this system is deadlock-free. Proof by contradiction. Suppose the system is deadlocked. This implies that each process is holding one resource and is waiting for one more. Since there are three processes and four resources, one process must be able to obtain two resources. This process requires no more resources and, therefore it will return its resources when done.