```
(d) #include <stdio.h>
    #include <stdlib.h>
    #include <pthread.h>

    pthread_t philosopher[5];
    pthread_mutex_t ~~chopstick~~ fork[5];

    void * Func(int n){
        printf("Philosopher %d is thinking \n",n);
        //when phil 5 is eating he takes fork 1 and 5
        pthread_mutex-lock(&fork[n]);
        pthread-mutex_lock(&fork[(n+1)%5]);
        printf("Philosopher %d is eating \n",n);
        sleep(3);
        pthread_mutex-unlock(&fork(n]);
        pthread-mutex_unlock(&fork[(n+1)%5]);

        printf("Philosopher %d finished eating \n",n);
        return(NULL);
      }
  int main(){
      int i;
      for(i=0;i<5;i++)
          pthread_mutex-init(&fork[i],NULL);

      for(i=0;i<5;i++)
        pthread_~~mutex~~create(&philosopher[i],
          NULL, (void *) Func, (void *) i);
    for(i=0; i<5; i++)
        pthread_join(philosopher[i], NULL);

    for(i=0; i<5; i++)
        pthread_mutex_destroy(&fork[i]);
}
```