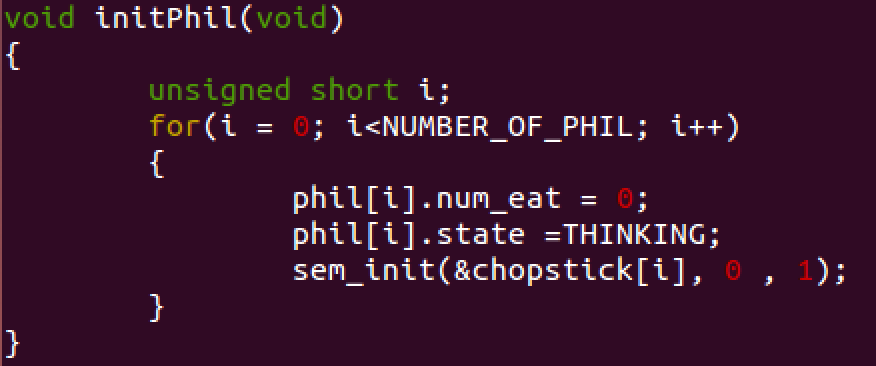
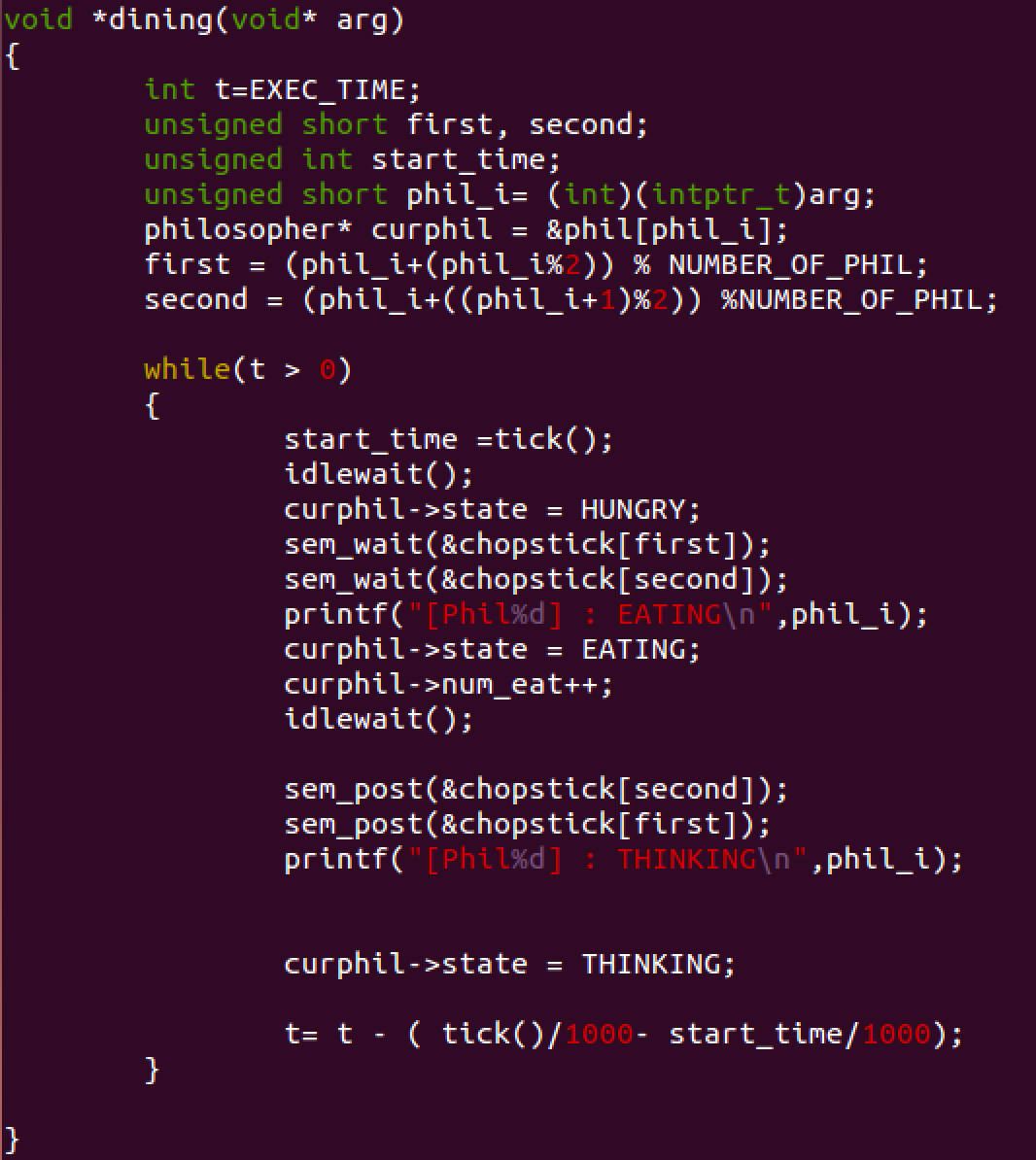
9주차 과제 수행 보고서

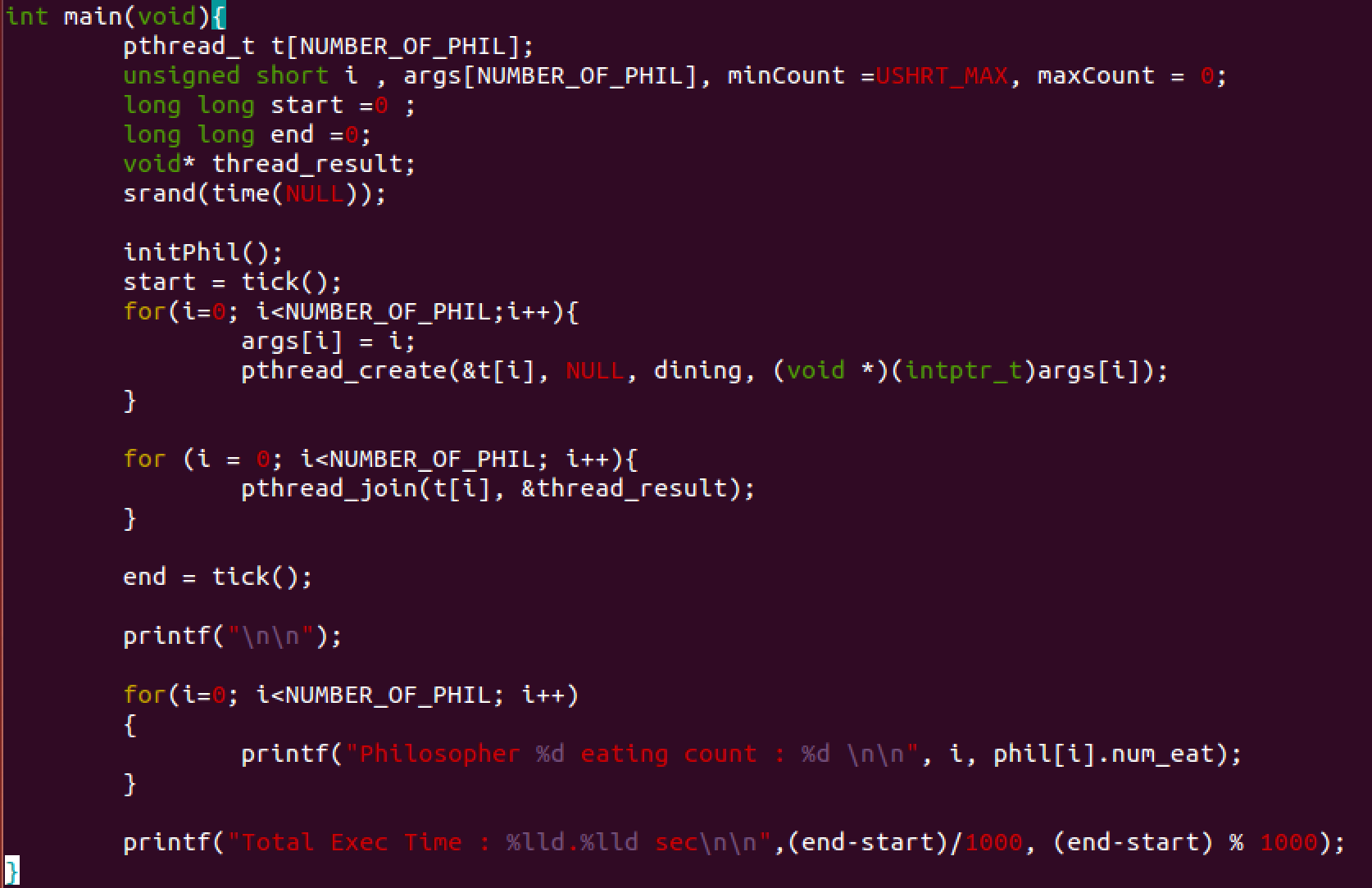
Solving The Dining Philosopher Problem

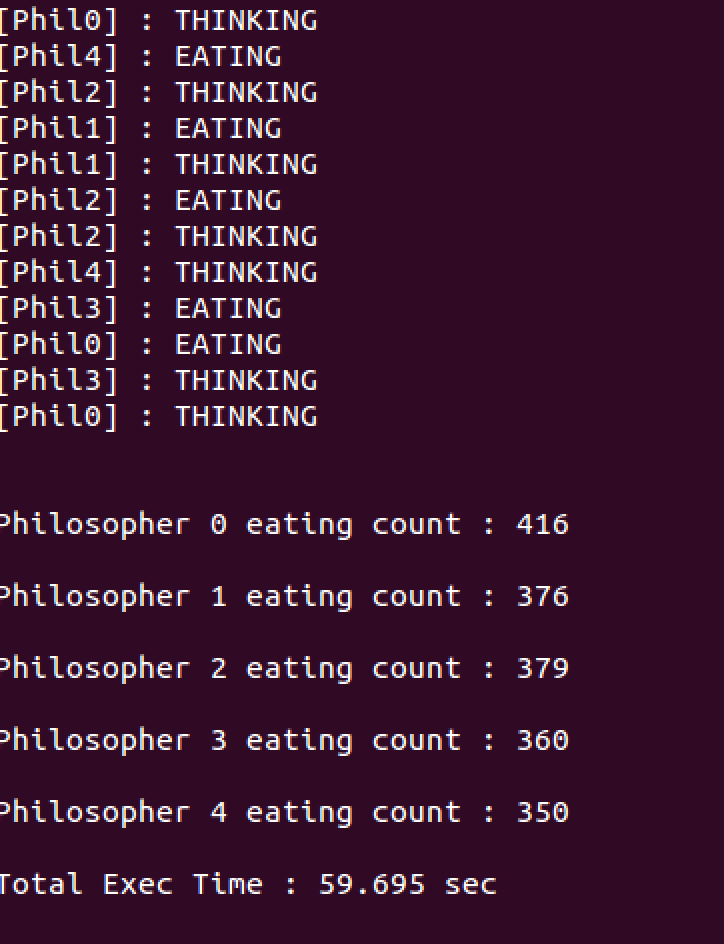
2013011554

컴퓨터전공

이새한

1. 과제내용 요약  
   The Dining Philosopher Problem의 전제는 자신의 양옆의 젓가락만 사용할 수 있으나 최악의 케이스로 각 철학자마다 젓가락을 가지고 있는 상태가 되면 서로 밥을 못먹는 Deadlock 문제 발생을 semaphore을 통해서 문제를 해결하고자 한다.
2. 소스코드 설명  
     
   우선 철학자 구조체에 대해서 모두 Thinking으로 초기화 하며 아직 먹은 횟수가 없으니 0, 젓가락에 해당되는 세마포도 초기화시킨다.  
     
     
   자신의 젓가락은 항상 오른쪽에다가 둔다는 가정을 하며 자신이 짝수번째면 자신의(=오른쪽) 젓가락을 먼저 잡으며 홀수면 왼쪽의 젓가락을 먼저 집는다. 이후 식사를 랜덤의 시간동안 하며 이후 역순으로 젓가락에서 손을 뗀다.

  
p\_thread를 초기화 시키고 이것을 실행시키는 메인 함수이다 만일 모든 쓰레드가 다 종료되면 철학자들의 먹은 회수를 출력한다.

1. 실행화면  
   
2. 피드백  
   지금과 같은 알고리즘으로 실행했을 때 0번 철학자가 나머지 철학자들에 비해서 압도적으로 많이 먹었음을 볼 수 있다. 이는 서로 대기를 하게 되는 중요한 세마포가 나머지 철학자들은 가지고 있지만 0번째 철학자는 그러한 영향을 가지고 있지 않다. 아마도 이러한 여파로 1번이 더 많은 횟수를 먹었을 것이라고 예상 가능하다.  
   짝수명의 철학자들의 케이스에서라면 이와 다르게 서로 충돌하게 되는 세마포를 모두 가지게 되므로 상대적으로 더 균등하게 나올 것이라 예상하였고 이와 비슷하게 출력이 되었다.  
   