

PA2: Word Counting 201421023_홍준엽

인적사항

- 사이버보안학과
- 201421023
- 홍준엽

과제 수행 정도

병렬화 - 완료

- 2개 이상의 쓰레드로 fgets를 해오면서 tokenization 실행
- (n번째 줄) % (쓰레드 개수) == (쓰레드 number) 일 때만 tokenization을 실행
- 아닐때는 while문을 계속 돌면서 대기

race condition 문제 해결 - 완료

- mutex_lock & mutex_unlock을 통해 해결

과제를 수행하면서 배운 것

쓰레드

쓰레드 생성 & 종료

- pthread_create
- pthread_join
- pthread_exit

race condition 문제

- 쓰레드 별로 전역변수에 접근할때 동기화 문제가 발생
 - mutex, semaphore 등의 해결방법이 있음
 - 이번 과제에서는 mutex를 사용
 - mutex: lock 변수 하나당 하나의 쓰레드만 동작 가능
 - semaphore: n개의 쓰레드가 접근 가능
- reader & writer 별로 lock을 걸 수도 있음
 - pthread_rwlock_init으로 초기화
 - pthread_rwlock_rdlock & pthread_rwlock_wrlock 을 사용해서 lock을 건다
 - pthread_rwlock_unlock 으로 lock을 푼다
 - reader lock이 걸려 있으면 여러 쓰레드가 읽기 가능, 쓰기 불가
 - writer lock이 걸려 있으면 다른 쓰레드는 접근 불가

pipelining

- 빨래를 할 때 세탁기와 건조기를 돌린다면 A가 빨래를 끝낼때 까지 B가 기다리는 것이 아니라 A가 세탁기 사용을 끝내고 건조기를 돌릴때 B가 세탁기를 사용하는 것처럼 한 시점에 여러 프로세스 혹은 스레드가 각기 다른 작업을 하는 것

과제에 대한 피드백

- 가장 오래 걸릴 것이라고 생각되는 부분 (파일을 읽어오는 부분: fgets)을 concurrent하게 동작하게 함.
- 한 쓰레드는 홀수 번째 줄을, 다른 쓰레드는 짝수 번째 줄을 읽어와서 토큰화.
- 다른 부분도 concurrent 하게 동작 할 수 있는 부분이 있음.
- 조금씩 바뀌가면서 시도해볼 예정