[ Read Prefeing RW Lock ]

대기 중인 writers의 수를 추적하기 위해 writers 필드 추가.

writelock을 대기하기 전에 writers 증가, lock을 획득한 후 writers 감소.

대기 중인 writers가 있는 동안 read 작업이 대기. writer가 대기 중일 때 read 작업이 더 이상 들어오지 않도록 함.

writers가 0보다 크면, writers가 0이 될 때까지 기다림.

writer가 deadlock 상태에 빠지지 않도록 하며, writer가 대기 중이지 않을 때는 reader가 우선적으로 lock을 획득.

[ Write Prefeing RW Lock ]

대기 중인 writers의 수를 추적하기 위해 writers 필드 추가.

writelock을 대기하기 전에 writers 증가, lock을 획득한 후 writers 감소.

writers가 0이 되면 read lock 해제.

대기 중인 writers가 있는 동안 read 작업이 대기하며, writer가 우선적으로 lock을 획득할 수 있도록 함.

모든 read 작업이 끝나면 대기 중인 writer 중 하나가 진행할 수 있도록 write lock 해제.

reader가 deadlock 상태에 빠지지 않도록 하며, writer가 우선적으로 lock을 획득할 수 있도록 함. reader가 대기 중이라도 writer가 우선적으로 lock을 획득.

[ Sequence RW Lock ]

seqlock 구조체를 초기화. seq 변수를 0으로 설정하고, mutex를 초기화.

write lock 획득. 이를 위해 mutex를 사용하여 lock을 걸고, seq 변수 증가. seq가 홀수이면 write 중임.

write lock 해제. seq 변수를 다시 증가시켜 짝수로 만듦. seq가 짝수이면 write가 완료되었음을 나타냄.

read 작업 시작 seq 변수가 짝수일 때까지 기다림. 짝수일 때 seq 변수를 반환하여 read 작업이 시작됨. 메모리 배리어를 사용하여 명령어 순서를 조정.

read 작업이 유효한지 확인. read 작업이 시작할 때의 seq 값과 현재 seq 값을 비교하여 다르면 read 작업이 다시 시도되어야 함. 메모리 배리어를 사용하여 명령어 순서를 조정.

[ Test Result ]

