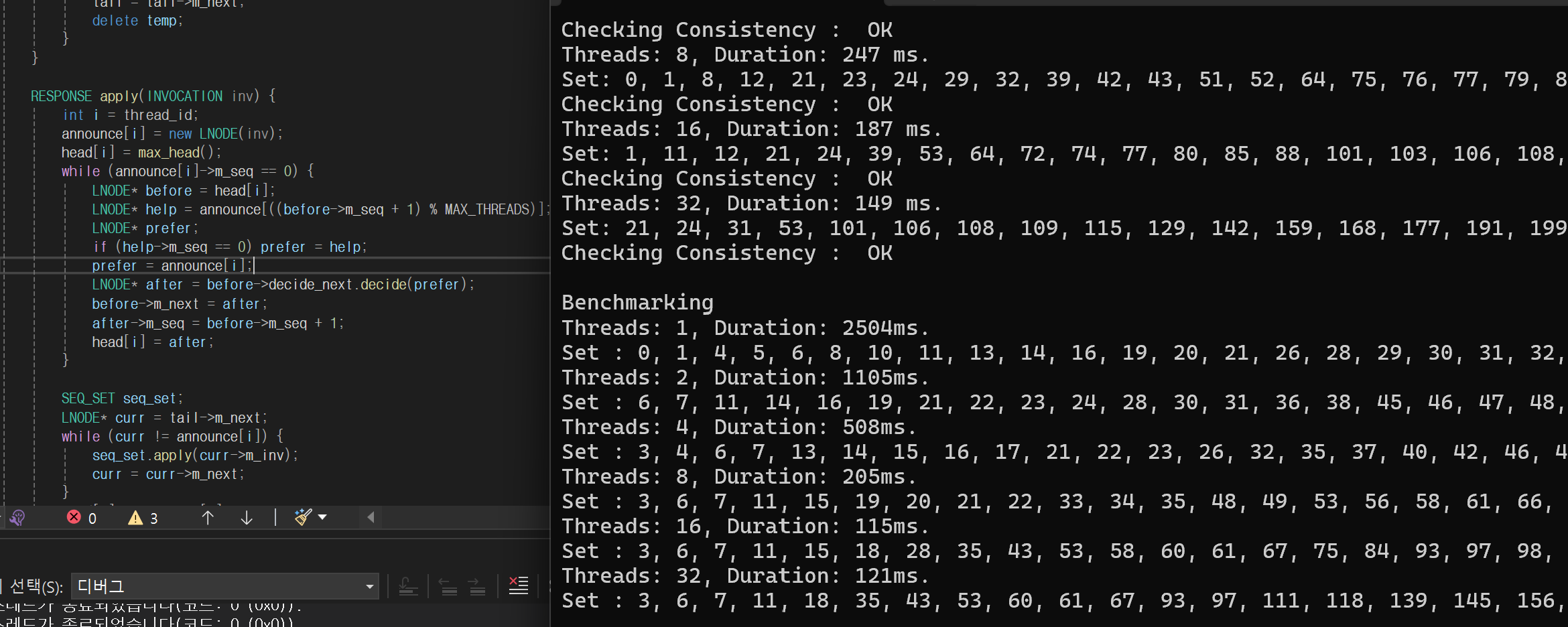
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **주차** | 9 주차 | **기간** | 2025.10.27  ~ 2025.11.02 | **지도교수** | (서명) |
| 이번주 한일 요약 | 1. 멀티코어 프로그래밍 학습  2. Boost Asio 서버 프레임워크 구축  3. 9주차 팀 회의 | | | | |

<상세 수행내용>

1. 멀티코어 프로그래밍 학습

Lock-Free Universal & Wait-Free Universal 객체를 활용한 Set 구현

속도 비교



32 thread 기준 LFU set은 162ms, WFU set은 122ms가 나왔으나 LF set에서는 16ms가 나오게 되어 해당 객체는 쓰기 힘들다는 결론을 내세웠다.

다만 어떤 객체든 무대기/무잠금 만능 병렬 객체로 변환시켜주기에 멀티쓰레드에서 모든 자료구조를 병렬화된 구조로 돌아가게 할 수 있다.

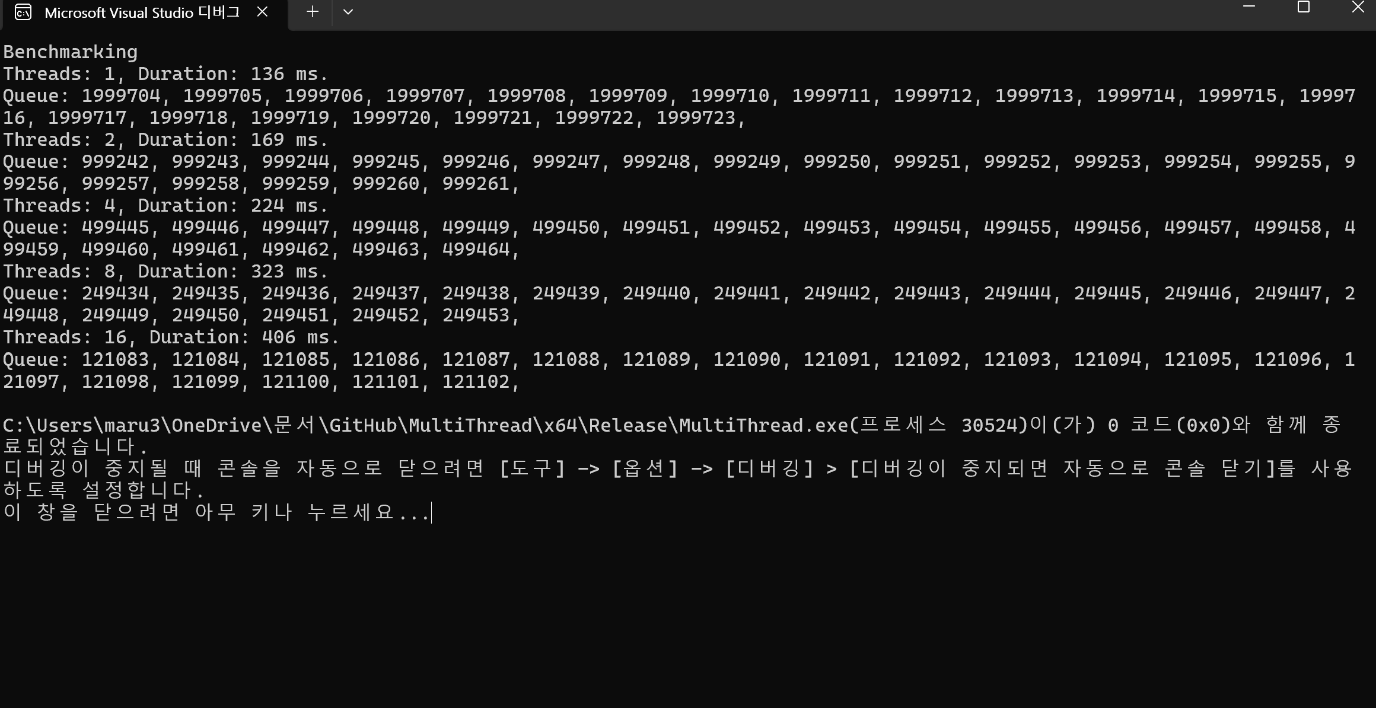
Coarse Grain 무제한 완전 Queue를 구현하고 멀티쓰레드 환경에서 적용

Enqueue와 Dequeue를 멀티쓰레드에서 실행시키기 위해 mutex 사용

돌아가는 것을 확인하기 위해 thread를 생성하고 benchmark 프로그램 실행

루프 결과를 화면에 출력 및 시간 측정

실행 및 속도 결과



2. Boost Asio 서버 프레임워크 구축

작업 환경에서 스마트 포인터 구현에 오류 발생 - unique\_ptr -> shared\_ptr 변경

(독자적 포인터 -> 공유 포인터)

서버 Session 생성 tcp socket으로 처리

Boost Asio에 맞게 함수 및 이벤트로 처리하는 부분을 Lambda로 처리

Server – Client Google protobuf 사용

3. 9주차 팀 회의

한 주 동안 작업 내용 공유

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** |  | | |
| **해결방안** |  | | |
| **다음주차** | 10 주차 | **다음기간** | 2025.11.03~ 2025.11.09 |
| **다음주 할일** | 1. git 학습  2. 멀티코어 프로그래밍 학습  3. Boost Asio 서버 프레임워크 구축  4. Unreal Engine 학습  5. 10주차 팀 회의 | | |
| **지도 교수**  **Comment** | 서버만 너무 하면 연동에 지장이 있을 것, 클라이언트도 어느정도 알아야 어느 부분을 설정해줘야 할지 알 수 있다. | | |