

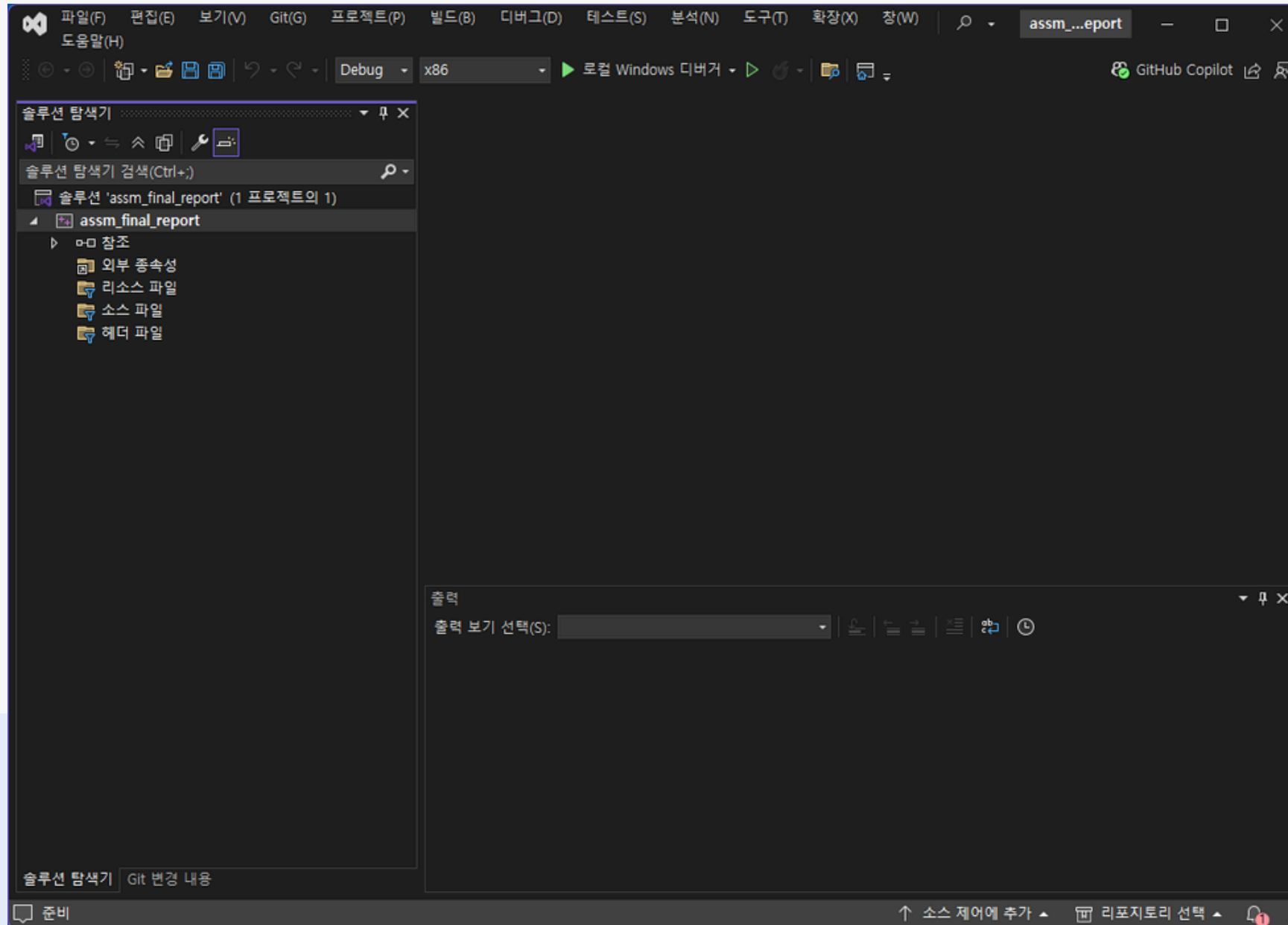
ASSEMBLY PROGRAMMING

BUBBLE SORT를

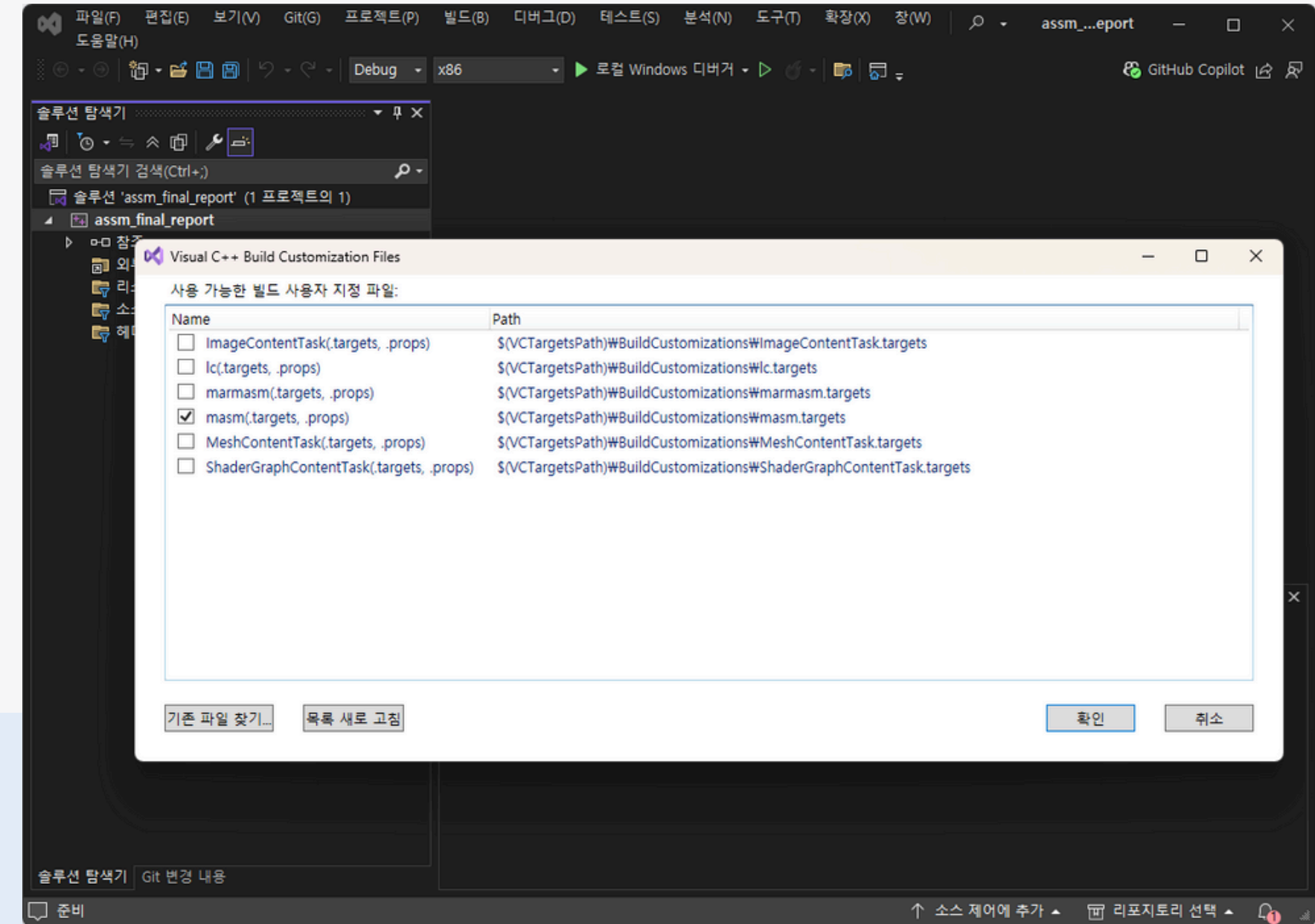
**C + ASM, LIBRARY, DLL**

2021752038 조현창

## 새 프로젝트 생성

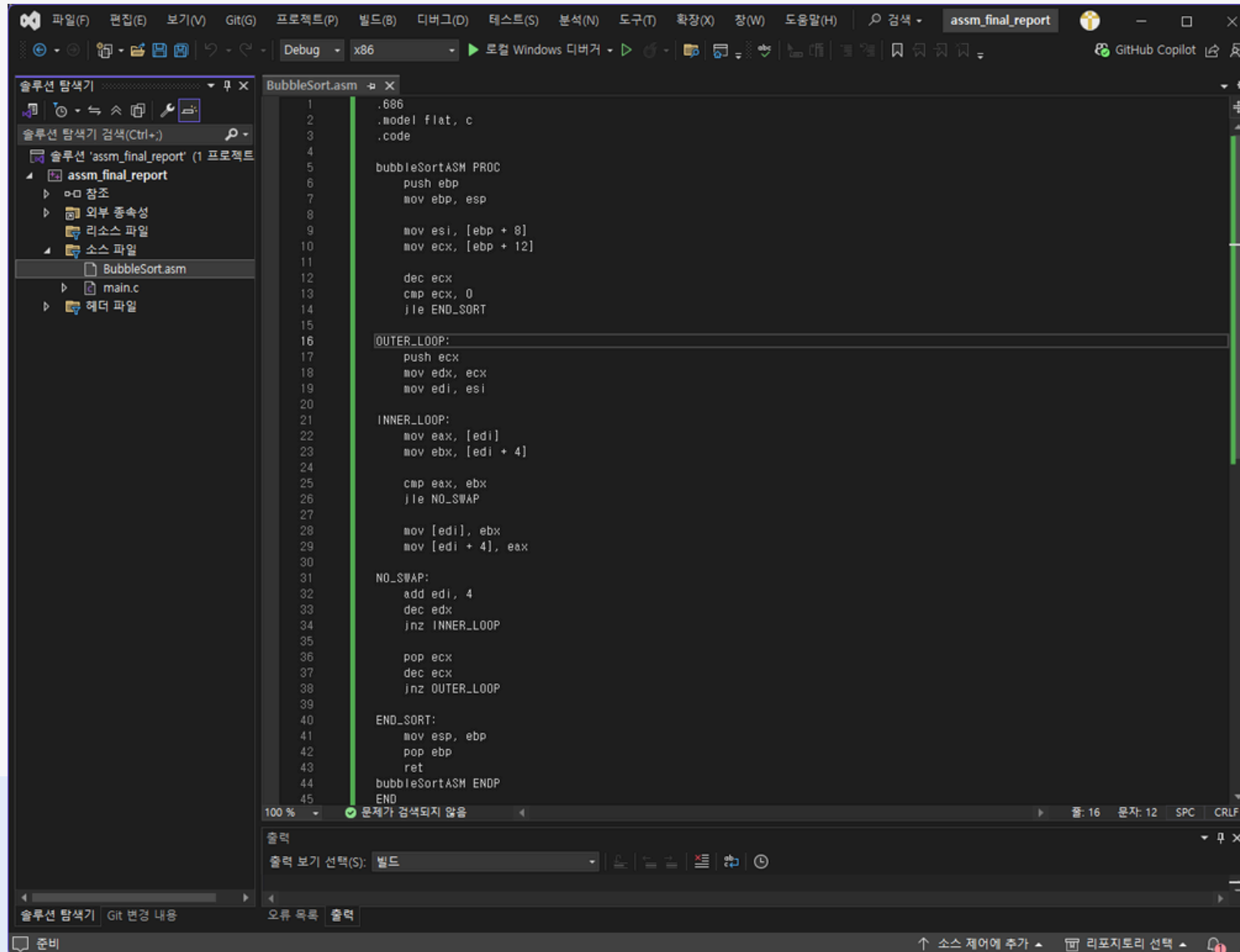


## MASM 설정

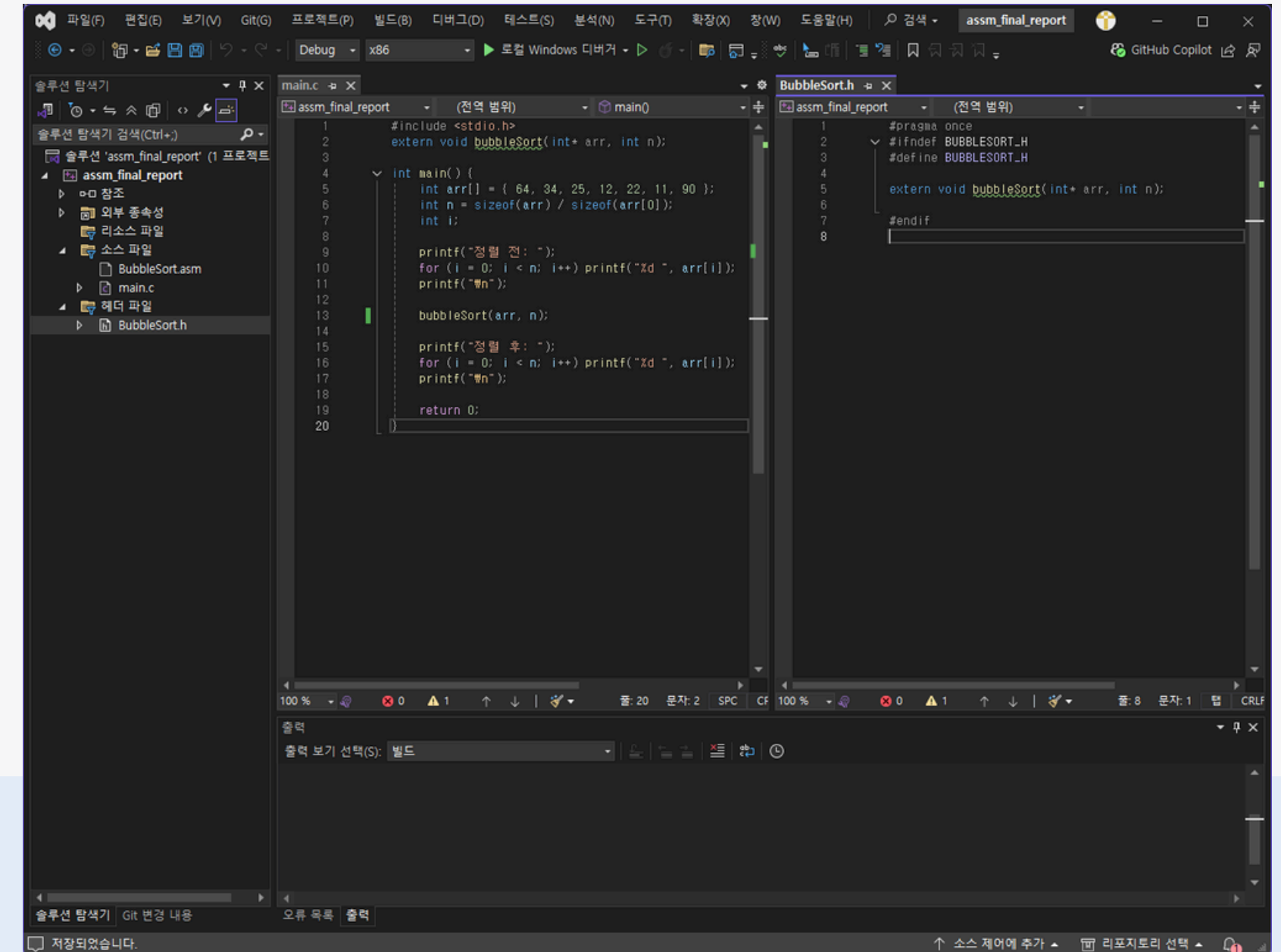


어셈블리어를 인식하기 위해 MASM 설정을 한다  
프로젝트 우클릭 → 빌드 종속성 → 사용자 지정 빌드 → masm(.targets, .props) 체크 → 확인

## BUBBLE SORT 어셈블리 코드 작성



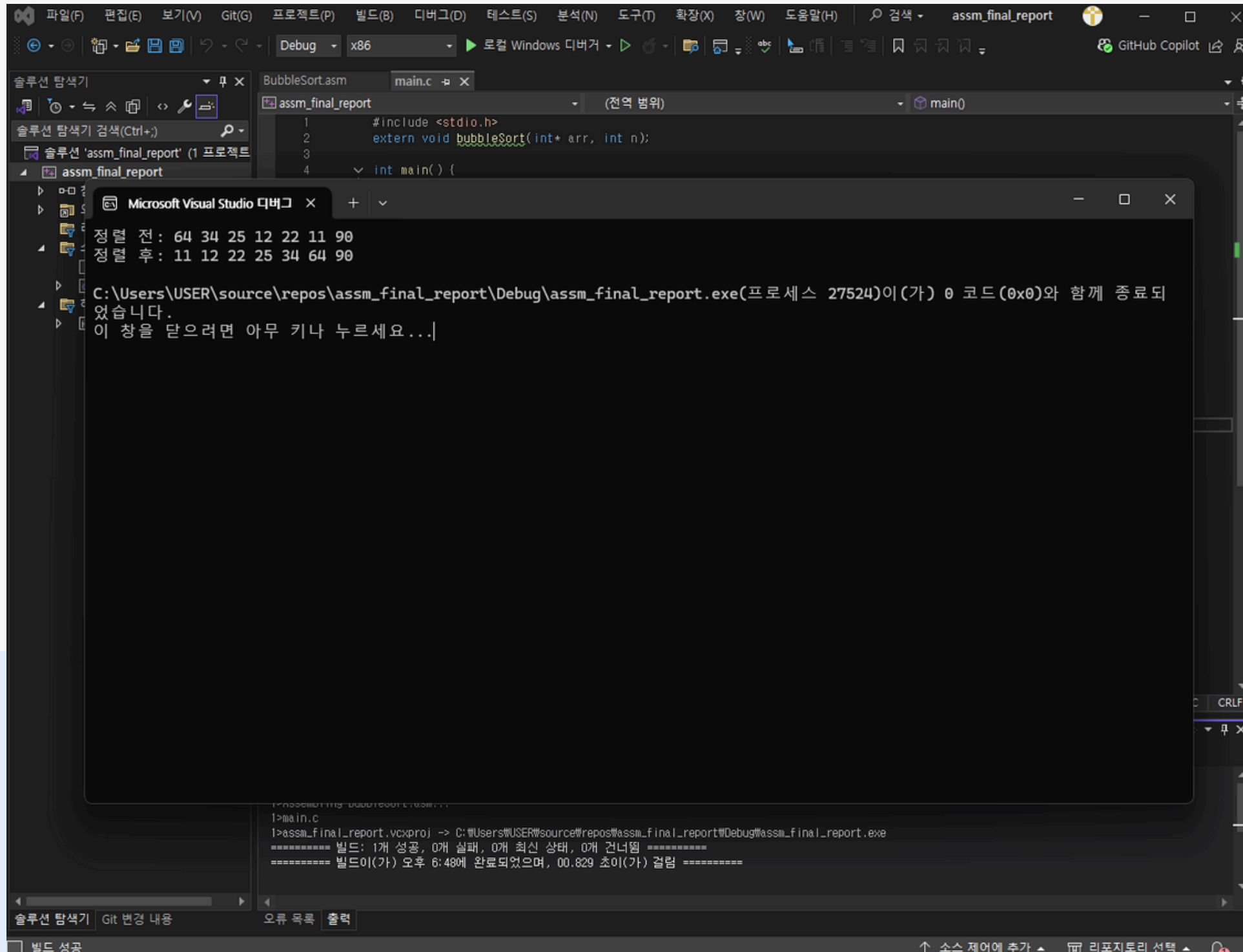
## MAIN역할을 할 C 파일과, 헤더 파일 생성



사진에 있는 MAIN.C 파일은 자기 파일 내용만 알기 때문에 어셈블리로 작성된 BUBBLE SORT 함수가 호출되면 에러가 발생합니다. 이때 헤더파일이 다른 파일에 정의 되어있다고 알려주는 역할을 해줍니다.

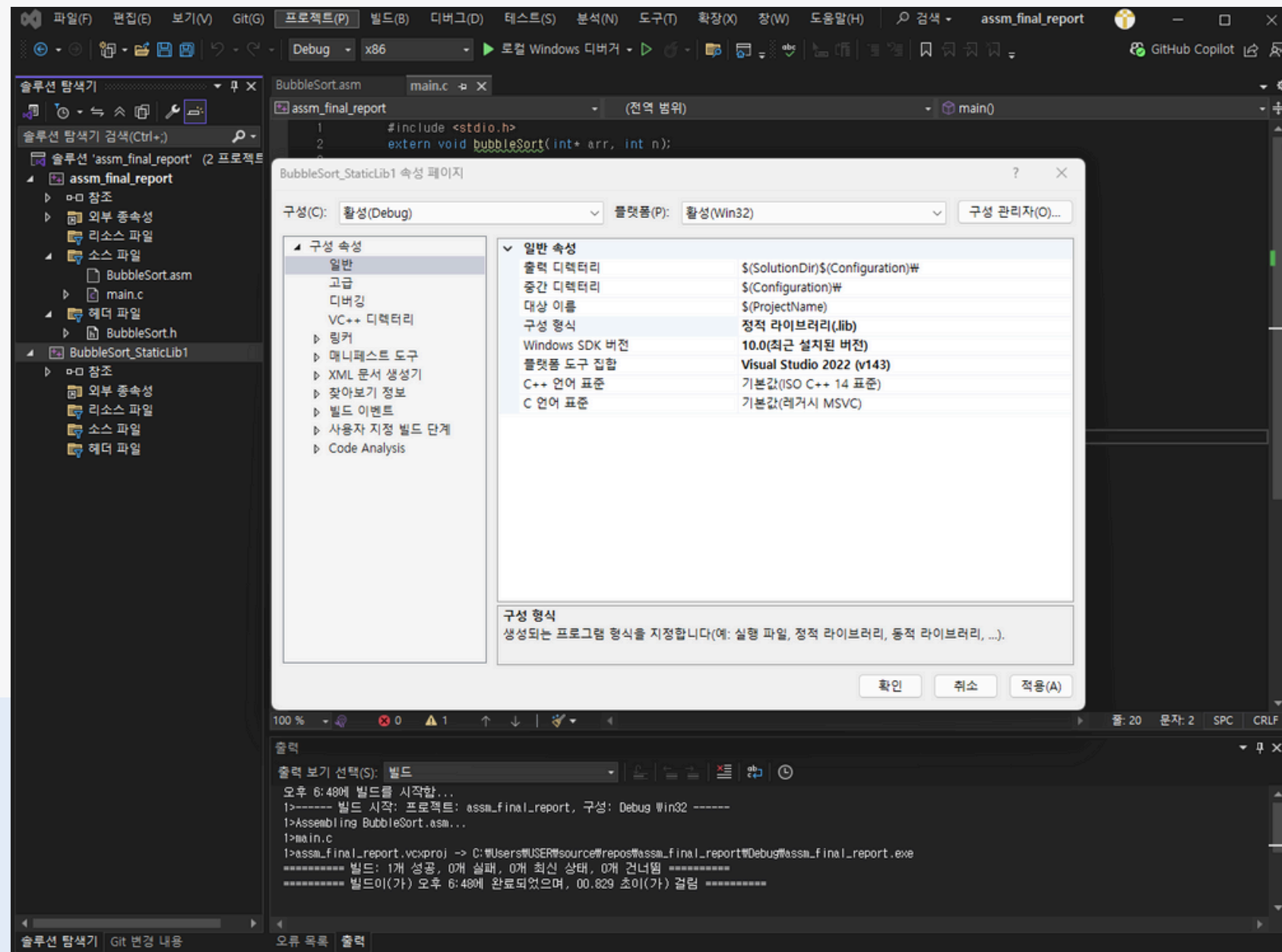
즉, 다른 소스 파일에 작성된 함수의 존재나 사용법을 컴파일러에게 미리 알려주어 문제없이 링크하기 위해서 사용됩니다.

## 실행

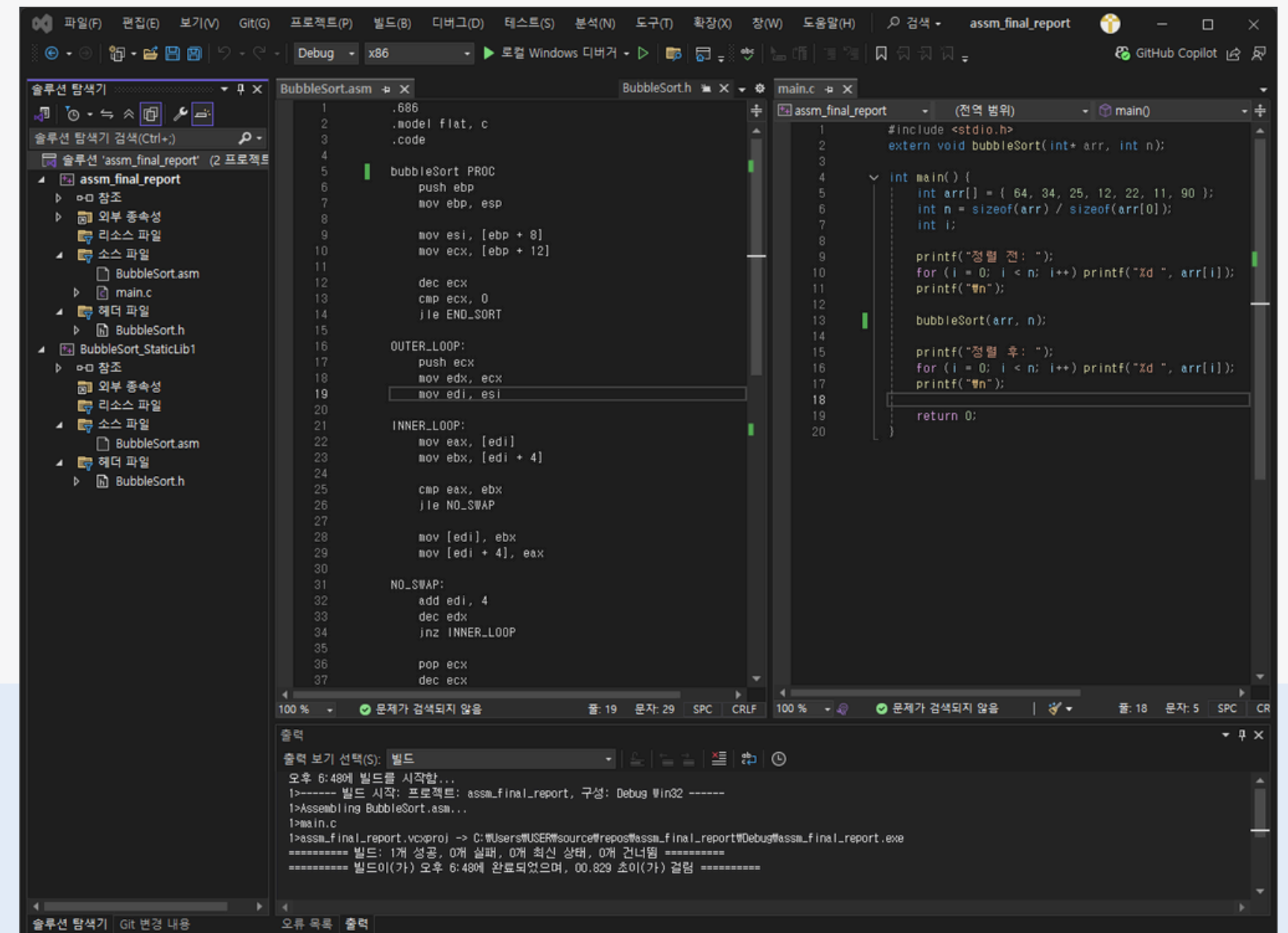


실행을 하게 되면 어셈블리어로 작성된 BubbleSort 함수가 Main인 C 파일에서 정상적으로 호출되고 작동하는 모습을 볼 수 있습니다

## 솔루션에 프로젝트 추가 후 구성형식을 정적 라이브러리로 변경



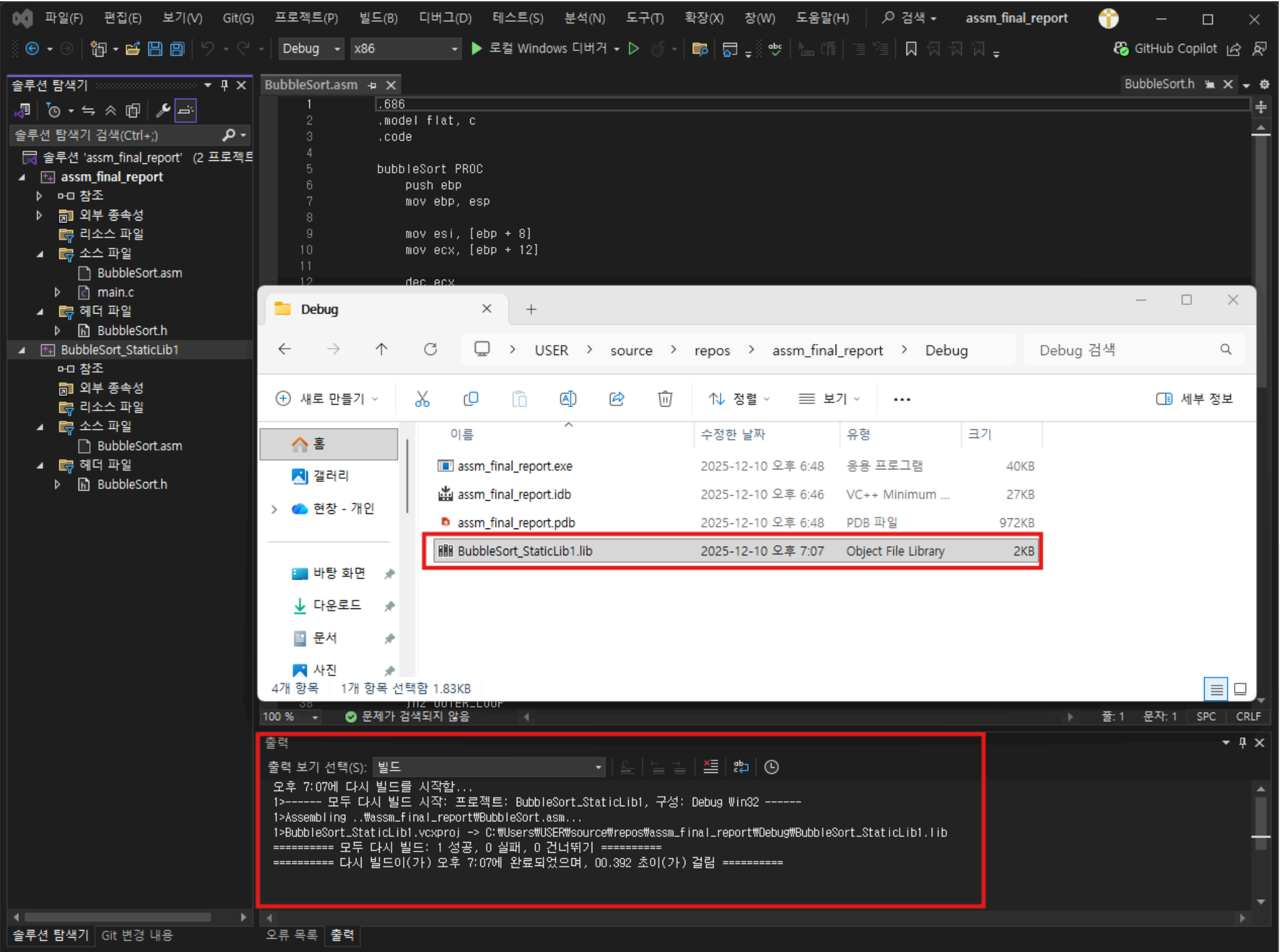
## 앞에 작성 했던 어셈블리 소스파일과 헤더파일 복사 해오기



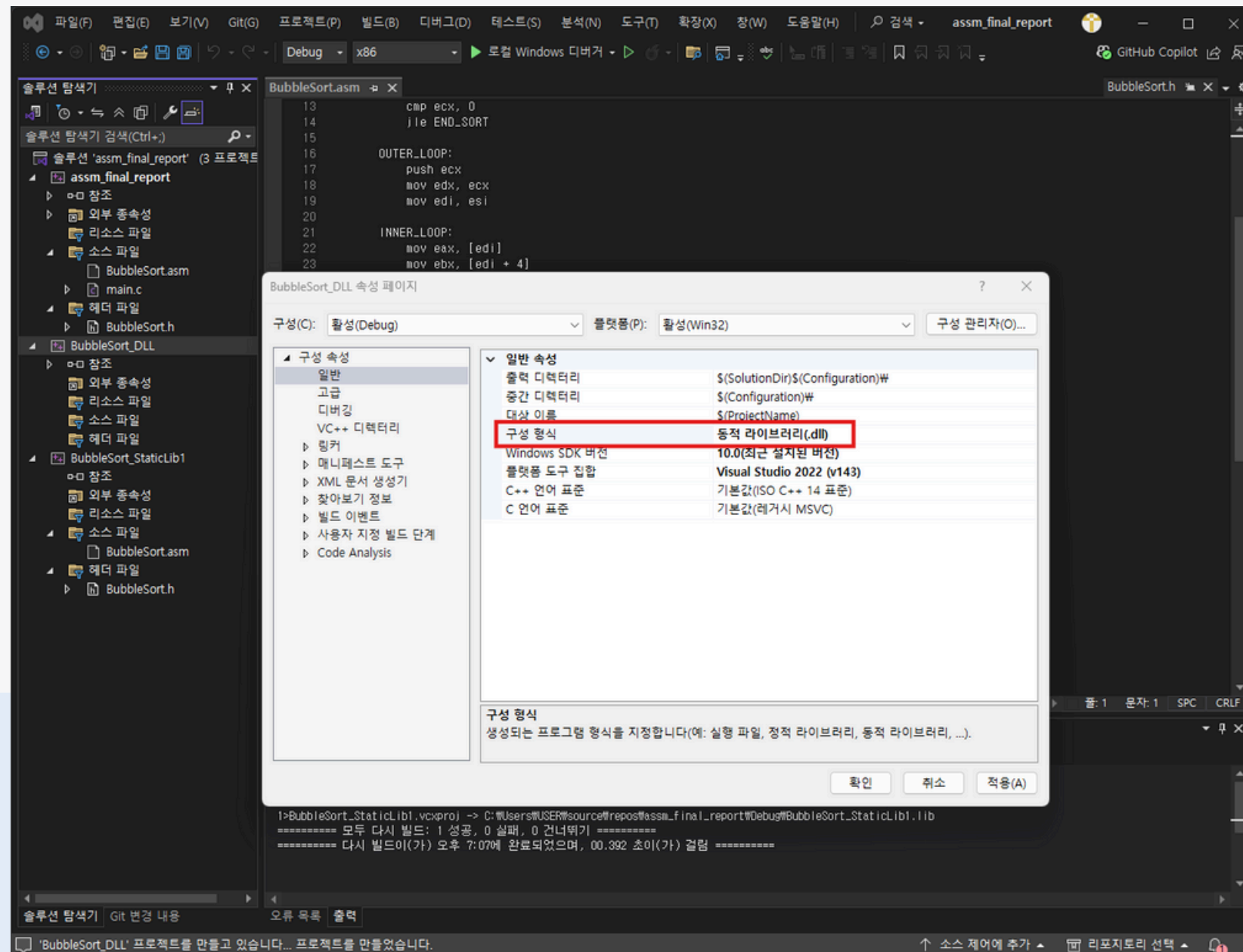
- 솔루션 우클릭 → 추가 → 새 프로젝트 → 빈 프로젝트
- 프로젝트 우클릭 → 속성 → 구성형식 → 정적 라이브러리 선택



빌드 및 실행

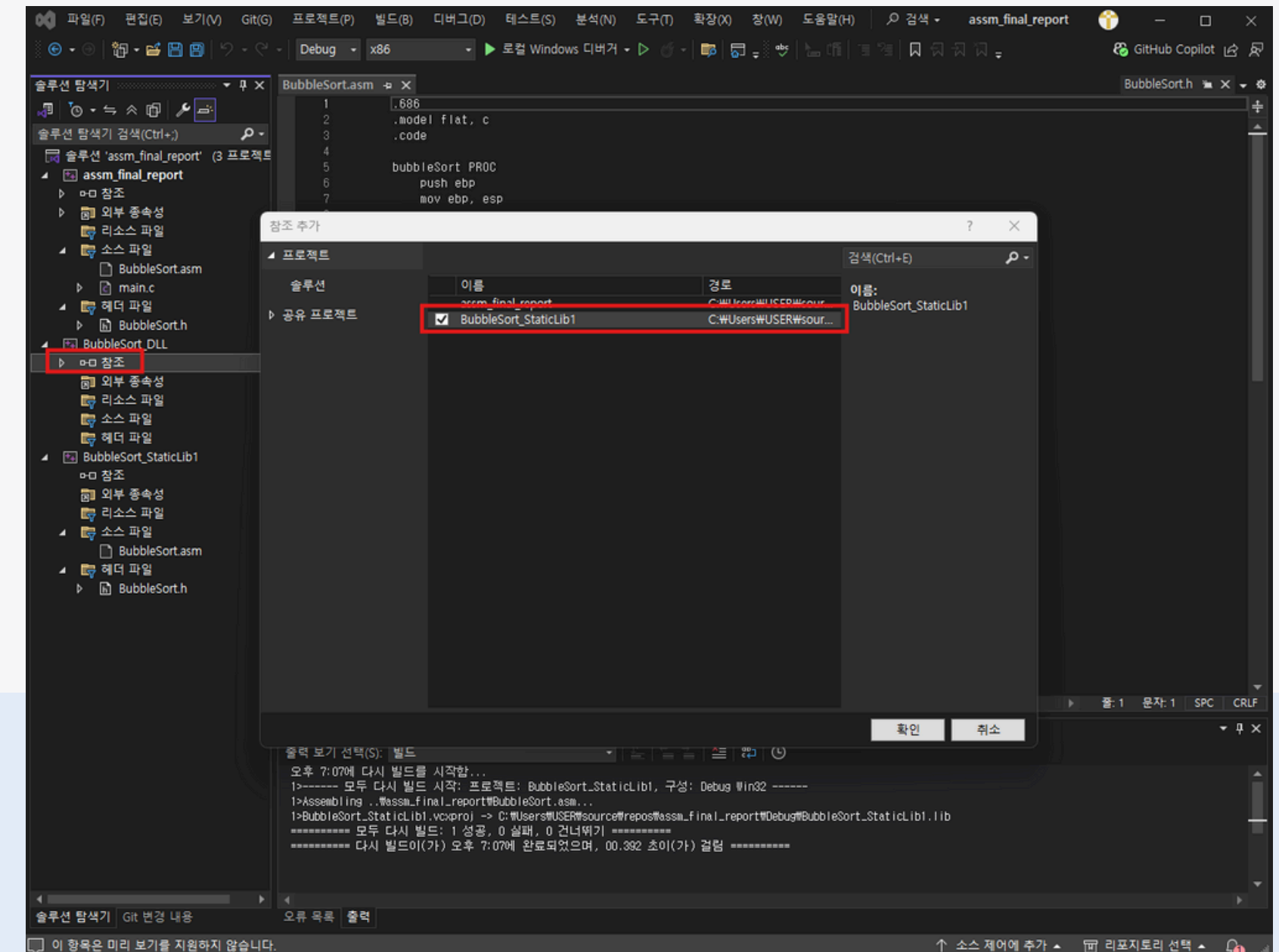


## 솔루션에 프로젝트 추가 후 구성형식을 동적 라이브러리로 변경



- 솔루션 우클릭 → 추가 → 새 프로젝트 → 빈 프로젝트
- 프로젝트 우클릭 → 속성 → 구성형식 → 동적 라이브러리 선택

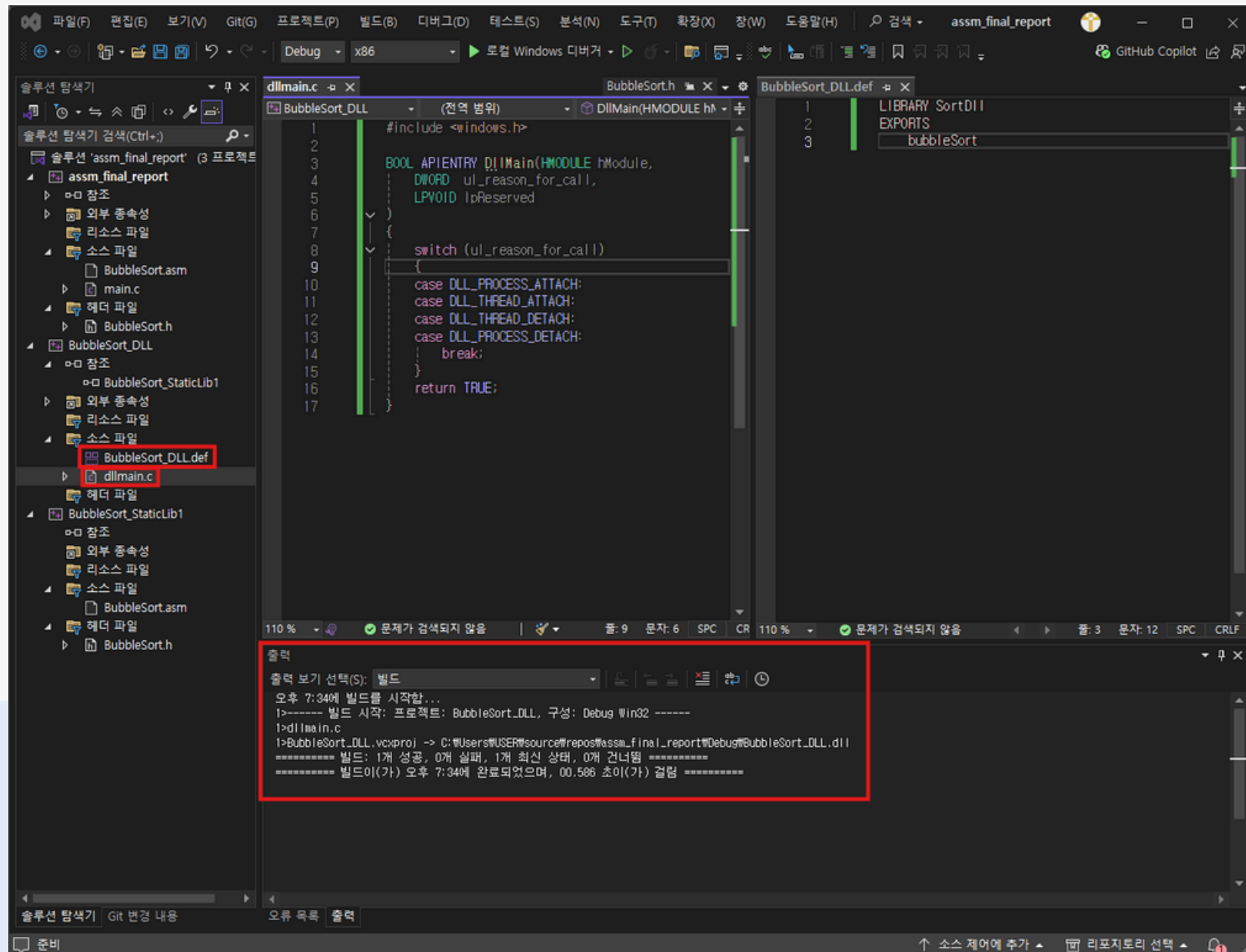
## 정적 라이브러리(Lib) 연결



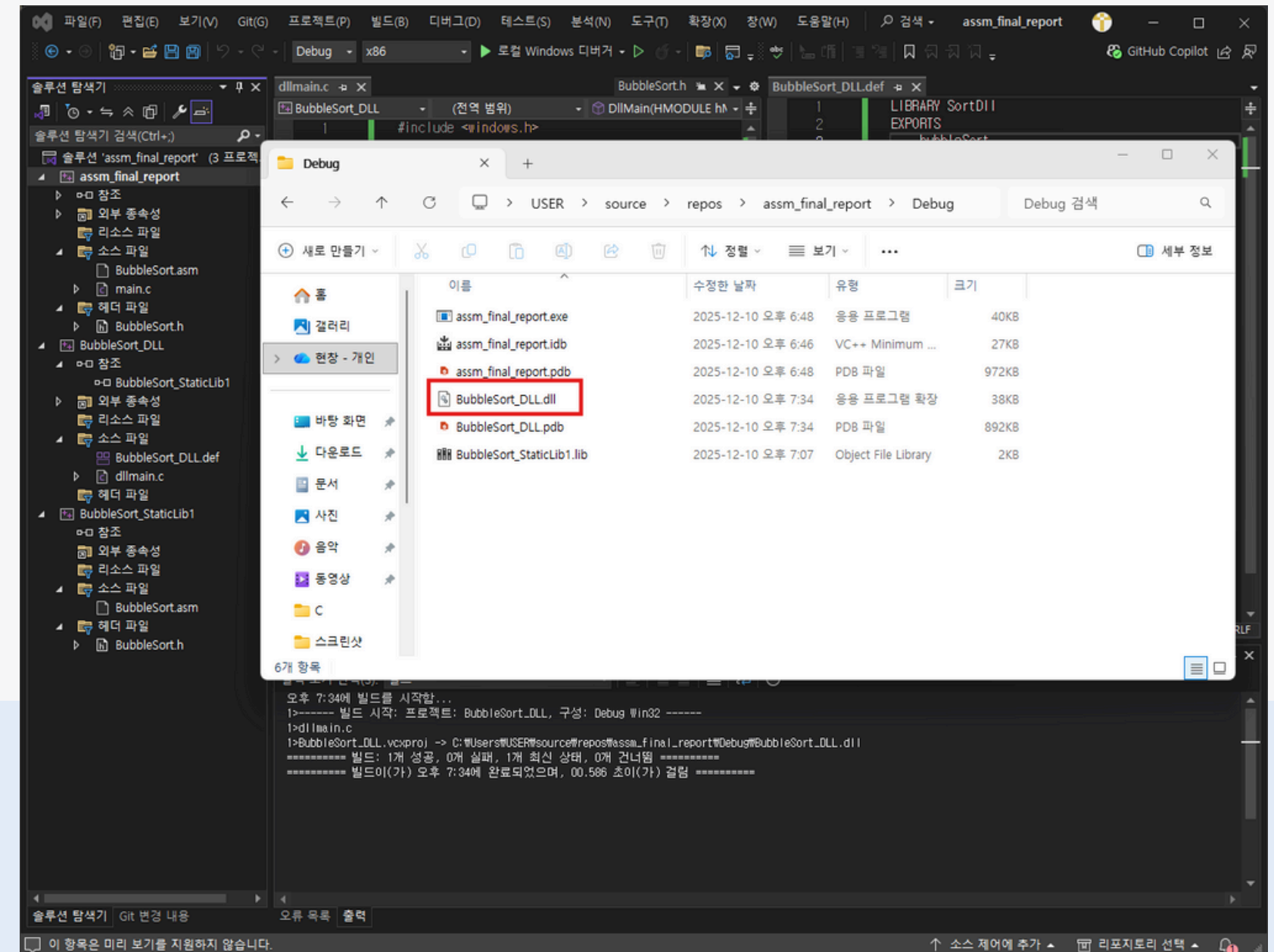
- 참조 우클릭 → 참조 추가 → 만들어둔 정적 라이브러리 선택 → 확인

DLL이 만들어질 때 만들어둔 어셈블리 부품(Lib)를 가져다 쓰기 위함입니다.

## DLL 기본 설정과 함수 이름 외부 공개(EXPORT)를 위한 내보내기 설정



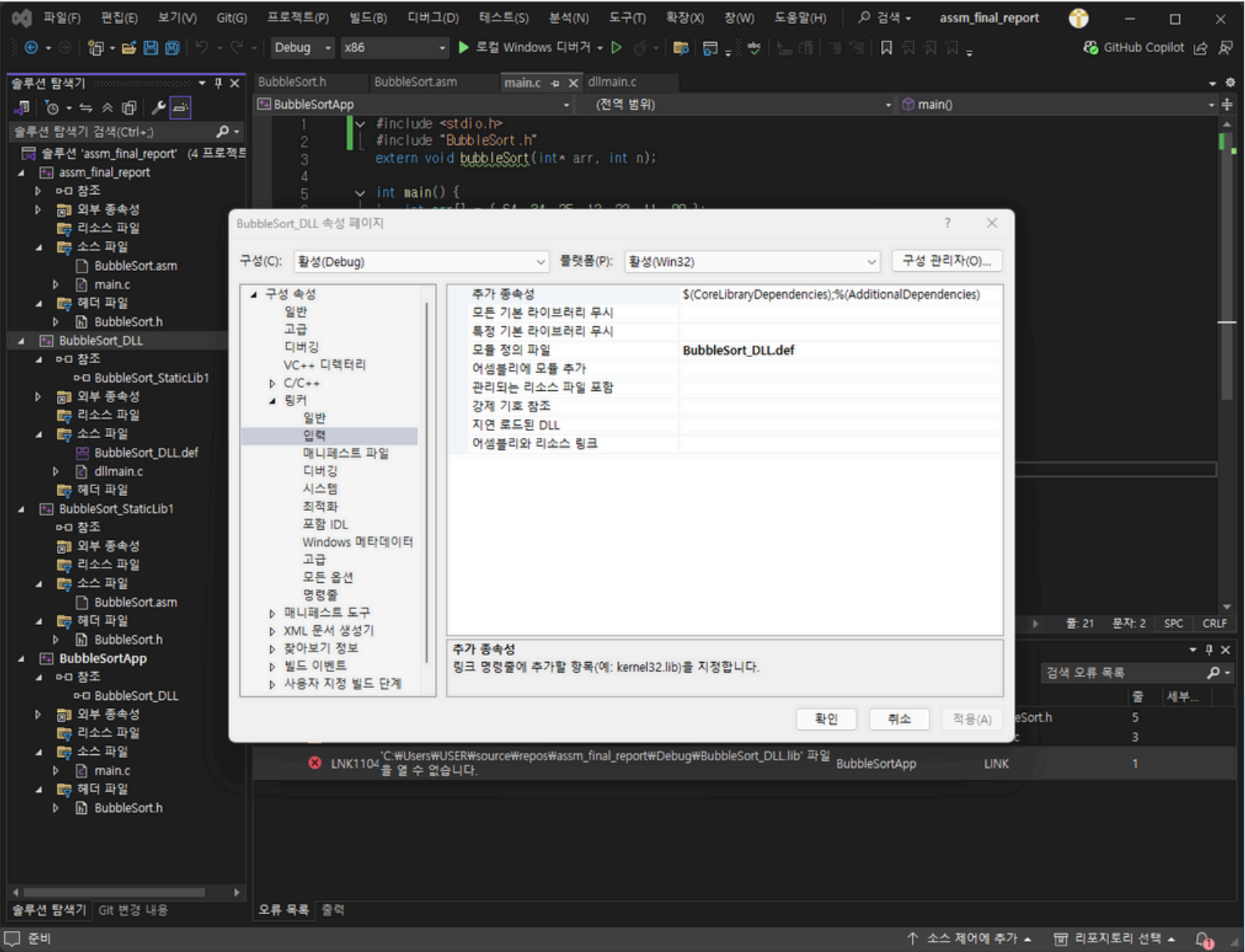
## 빌드 및 실행 확인



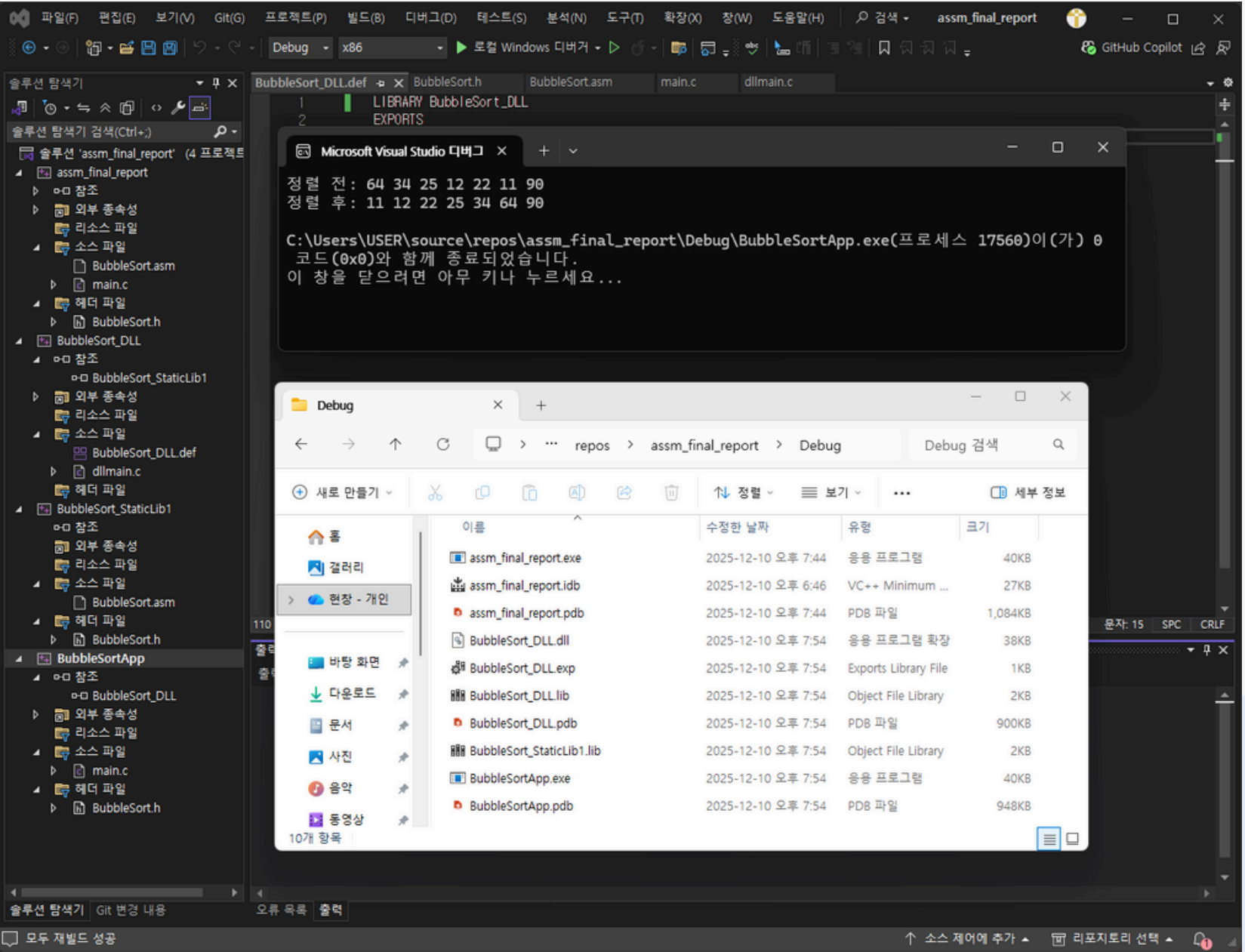
- DLL의 필수 구성 요소 파일 생성
- 소스파일 우클릭 → 추가 → 새항목 → 텍스트 파일 → 이름.def → 확인
  - 외부 프로그램들이 확인을 하고 사용 할 수 있도록 하는 설정




모듈 정의 파일 설정



빌드 및 실행 확인





# 점수: 10점

- 제 시간에 완료
- 모든 요구사항 문제 없이 완료



# 감사합니다

레퍼런스: <https://gemini.google.com/>