

2주차 예비보고서

전공: 신문방송학과

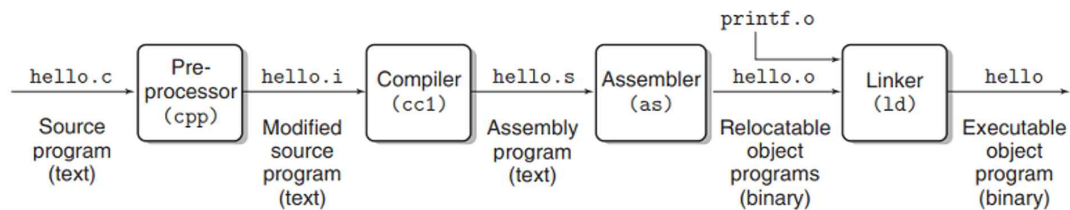
학년: 3학년

학번: 20191150

이름: 전현길

1. 본문을 읽고 c/c++ 프로그램의 컴파일 과정에 대하여 요약하라. 각 단계별로 하는 일들과 관련된 도구들 또한 명시하라.

C 프로그램의 컴파일 과정은 크게 전처리(preprocess), 컴파일(compile), 어셈블링(asmbling), 링킹(linking)의 네 단계로 구분된다. 만약 우리가 UNIX 환경에서 'hello, world!'를 출력하는 c 프로그램 'hello.c'를 작성하고, 이를 'gcc -o hello hello.c'라는 명령어를 통해 실행 파일 'hello'를 생성했다면, 이는 아래와 같은 과정을 거친 것이다.



[GCC] GCC 컴파일러 개념 정리, <https://80000coding.oopy.io/d5fa7c87-192f-4c68-95eb-aa4af5b9dbf5>

gcc(GNU Compiler Collection)는 이 과정에서 전처리기(preprocessor), 컴파일러(compiler) 역할을 주로 수행하지만, 내부적으로 UNIX의 assembler인 as, linker인 ld를 호출하기 때문에 사실상 컴파일 과정 전반에 영향을 미친다고 볼 수 있다.

먼저 **전처리 과정**에서는 명령어에 따라 전처리기 지시자를 처리하는 작업을 수행한다. 구체적으로는 헤더 파일(#include <stdio.h>, etc.)을 삽입하고, 매크로(#ifdef, #define, etc.)를 치환한다. 전처리 단계를 거치면 소스 파일 hello.c의 내용을 바탕으로 확장 소스 파일인 hello.i가 생성된다.

다음으로 **컴파일 단계**에서는 확장 소스 파일인 hello.i를 어셈블리어(assembly language)로 **번역**하고, 소스 코드의 **오류를 처리**하며, **코드를 최적화**하는 작업이 수행된다. 컴파일 과정에서 발생한 오류는 컴파일 에러(compile error)라고 부르며, 컴파일 옵션 명령어(O0, -O1, -O2, etc.)를 통해 코드 최적화의 정도를 제어할 수 있다. 컴파일 단계를 통해 어셈블리어 파일 hello.s가 생성된다.

셋째로 **어셈블 단계**에서는 어셈블리어 파일 hello.s를 바탕으로 기계어로 된 오브젝트 파일 'hello.o'를 생성한다. 이 때 생성된 파일은 명령어와 데이터가 일정한 규칙을 갖고 구분되어 있으며, LINUX 기반 시스템의 경우 ELF binary format의 구조를 따른다. 링킹 단계에서는 위의 포맷을 바탕으로 실행 파일을 생성하게 된다.

마지막으로 **링킹 단계**에서는 hello.o 오브젝트 파일, 프로그램의 표준 C 라이브러리(`printf()`, `scanf()` 함수 등), 유저 라이브러리가 연결되어 실행 파일을 생성한다. 이때 표준 C 라이브러리는 별도로 명시하지 않아도 자동으로 링크된다. 링킹 단계를 통해서 실행 가능한 파일인 'hello' 파일이 최종적으로 생성되게 된다.