

실험 UNIX-2 예비보고서

전공: 컴퓨터공학과

학년: 2

학번: 20191559

이름: 강상원

1. 목 적

UNIX 상에서 제공하는 C/C++관련 도구를 미리 사용해 봄으로써, 수업시간에 실습이 원활히 진행될 수 있도록 한다.

2. 예비 학습

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

다음 쪽에 계속 ->

C/C++ 프로그램의 컴파일 과정은 크게 프리프로세싱(preprocessing) 단계, 컴파일링(compiling) 단계, 어셈블링(Assembling) 단계, 링킹(linking) 단계로 나누어진다.

1. Preprocessing

컴파일러가 코드를 처리하기 전에 특정 변수를 미리 정해진 문자열로 바꾸는 작업을 한다.

헤더 파일을 포함하고 매크로 확장(매크로 생성)을 한다.

```
ex) #include <stdio.h>
      #include <stdlib.h>
      #define MAX 987654321;
```

와 같은 구문을 처리하는 역할을 하는 것이다.

2. Compiling

컴파일링 단계에서 C/C++ 코드가 어셈블리어로 변환된다.

컴파일링 단계는 크게 세 과정으로 나뉘는데, front-end, middle-end, back-end로 구성되어 있다.

front-end에서는 문법 오류, 구문 오류, 기능 오류를 검출하고 언어 독립적인 GIMPLE Tree를 생성하여 전달한다.

middle-end에서는 GIMPLE Tree를 Static Single Assignment (SSA) 형태로 변환하여 환경(OS, CPU Architecture)에 종속적이지 않은 최적화를 한다. 이후 back-end에 Register Transfer Language (RTL) 형태를 전달한다.

back-end에서는 환경에 종속적인 최적화를 한다. 이후 어셈블리 코드를 생성하게 된다.

gcc에서 컴파일링 단계까지만 진행하려면 gcc -S hello.c 와 같이 쓰면 된다.

출력 결과물은 hello.s 파일이다. 사용자가 작성한 코드가 무슨 언어로 작성되었든 간에, 컴파일링 단계에서 기계어로 변환되므로 다음 단계부터는 사용자가 어떤 High-level 언어를 사용했는지 걱정할 필요가 없다.

3. Assembling

어셈블링 단계에서는 컴파일링 단계의 결과물을 한번 더 처리한다. 기계어로 번역하여 오브젝트 파일을 만드는데, gcc에서 이 단계까지만 진행하려면 gcc -c hello.c 와 같이 쓴다.

출력 결과물은 hello.o 파일이다.

이는 오브젝트 파일로, 보통의 경우 컴파일을 끝마친 후 삭제되지만, 위와 같이 -c 옵션을 주면 파일을 남길 수 있다.

4. Linking

어셈블링 단계에서 만들어진 오브젝트 파일을 읽어들이어 실행 가능한 파일을 생성한다. 바이너리 파일이 서로 연결되고, 라이브러리와의 링킹한다. gcc에서 기본 출력 파일은 a.out 이지만 gcc hello.c -o hello와 같이 이름을 바꿀 수 있다. 최종적으로 운영 체제가 실행할 수 있는 실행 파일(exe)을 만들어 낸다.