

# CSE3013 (컴퓨터공학 설계 및 실험 I)

## UNIX-2 예비 보고서

서강대학교 컴퓨터공학과 박수현 (20181634)

서강대학교 컴퓨터공학과

### 1 목적

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

### 2 예비 학습

C/C++ 소스 파일의 빌드 과정은 프리프로세싱<sup>preprocessing</sup>, 컴파일링<sup>compiling</sup>, 어셈블링<sup>assembling</sup>, 링킹<sup>linking</sup>으로 구성된다.

#### 2.1 프리프로세싱

프리프로세싱은 선행 처리기라고도 불리는 프리프로세서<sup>preprocessor</sup>에 의해 이루어진다. 이름에서 보듯이, 코드를 컴파일러가 처리하기 전에 특정 변수를 미리 정의된 문자열로 치환하는 등의 작업을 수행한다.

C/C++에서는 매크로<sup>macro</sup>와 매크로 확장<sup>macro expansion</sup>을 정의할 수 있다. 사용예는 다음과 같다.

```
1  #include <stdio.h>
2  #define PI 3.141592653
3  #define area(x) (PI * (x) * (x))
4
5  int main() {
6      printf("%f\n", area(10.0f));
7      return 0;
8  }
```

이 코드는 프리프로세서에 의해 컴파일러가 처리하기 전에 다음과 같이 바뀐다.

```
1  #include <stdio.h>
2
3  int main() {
4      printf("%f\n", (3.141592653 * (10.0f) * (10.0f)));
5      return 0;
6  }
```

## 2.2 컴파일링

컴파일은 **컴파일러** compiler에 의해 이루어진다. C/C++ 소스 파일은 컴파일러를 통해 어셈블리 소스 파일이 된다. 대표적인 컴파일러로는 gcc가 있는데, gcc는 프리프로세서의 역할도 한다.

컴파일 과정은 전단부 front-end, 중단부 middle-end, 후단부 back-end로 구성된다.

전단부에서는 소스코드를 토큰 token으로 나누고, 이 토큰들로 파스 트리 parse tree를 구성하여 문법적, 기능적 오류를 검출한다. 마지막으로 GIMPLE 트리를 만들어 중단부에 제공한다.

중단부에서는

## 2.3 어셈블링

C/C++ 코드가 컴파일된 결과물을 목적 파일 object file이라고 한다.