**컴파일러 설계**

**프로젝트 1: 스캐너**

**2013012278 하경모**

**0. compilation environment**

- Ubuntu 18.04 LTS

- gcc 7.3.0

**1. Tiny Compiler modified**

**1.1 목적**

- tiny compiler를 수정해서 c-minus 스캐너 구현

**1.2 디자인**

- tiny compiler의 키워드와 심볼을 c-minus에 맞게 수정

- 키워드 : else, if , int, return, void, while

- 심볼 : + - \* / < <= > >= == != = ; , ( ) { } [ ] /\* \*/

- 그에 따른 Makefile 수정

**1.3 구현**

- globals.h, scan.c, util.c에 새로운 키워드와 심볼 추가

- 스캐너로 쓰기 위해 main.c의 define값 조정

- 새롭게 추가된 키워드와 심볼에 맞게 scan.c의 getToken 함수 수정

- 주석 심볼 /\*, \*/와 분수 심볼 /를 구분하기 위해 다음 캐릭터를

읽어 구분

- 이상, 초과와 이하, 미만을 구분하기 위해 다음 캐릭터를 읽어 구분

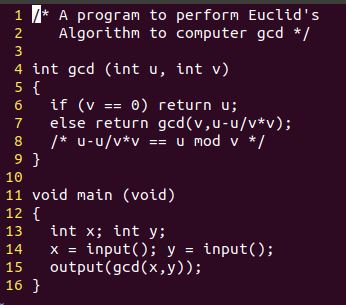
**1.4 실행 결과**

- 실행 방법

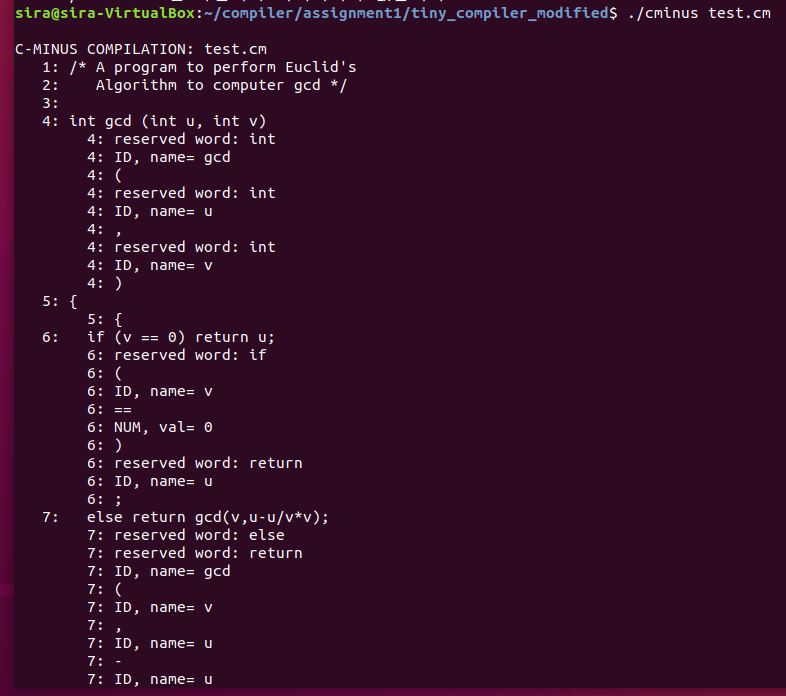
- make

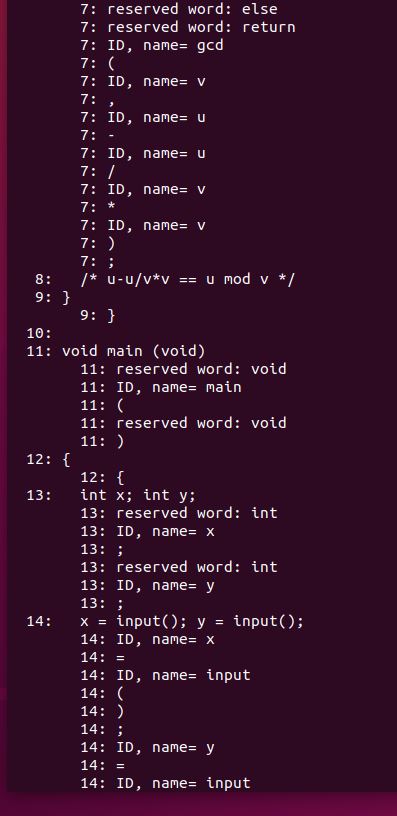
- ./cminus test.cm

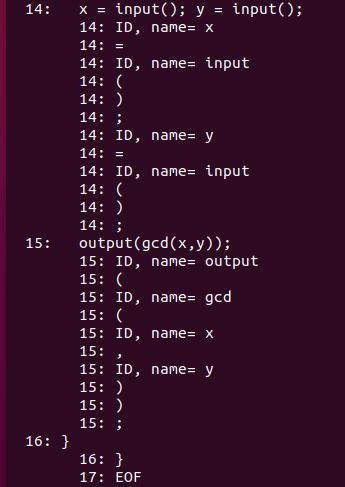
테스트 예제

****

실행 결과







**2. Using lex**

**2.1 목적**

- 렉스 파일을 수정해 flex를 이용해 c-minus 스캐너 구현

**2.2 디자인**

- 렉스 파일의 키워드와 심볼을 c-minus에 맞게 수정

- 키워드와 심볼은 tiny compiler modified와 동일

- 그에 따른 Makefile 수정

**2.3 구현**

- globals.h, main.c, util.c를 tiny compiler modified와 같이 수정

- cminus.l (원본은 tiny.l) 파일에 새로운 심볼과 키워드 추가

- 주석문자 /\* 와 분수문자 /를 구분하기 위해 별도 처리

**2.4 실행 결과**

- 실행 방법

- make

- ./cminus\_flex test.cm

- 예제 코드는 위와 동일

실행결과 (아래에 명시)

