Project 3. Semantic

2016025823

JUNG JAE, LEE

1. Compilation method and environment

* Edit - MaxOS, VisualStudio Code
* Run – Virtual Box, Ubuntu 16.04LTS
* Manage – Github(in progress), Gitlab(done)
* HowtoRun
  + make 🡪 Compile and link
  + ./cminus ./test\_file\_name 🡪 Execute

1. analysis implementation process and source code

* 텍스트이(가) 표시된 사진

  자동 생성된 설명symtab.c – scope는 스택의 개념을 활용하여 구현하였다. 또한, 모든 스코프 데이터는 symtab.c 에서 선언 및 접근하기 때문에, 명세와는 다르게 st\_insert 함수의 파라미터에서 스코프 포인터를 제거하였다.

Figure . st\_insert()

* 텍스트이(가) 표시된 사진

  자동 생성된 설명텍스트이(가) 표시된 사진

  자동 생성된 설명텍스트이(가) 표시된 사진

  자동 생성된 설명analyze.c – 명세처럼 input과 output을 global scope에 추가해 주었고, node 정보를 바탕으로 symbol table을 만들수 있는 insertNode 함수와 에러를 찾을 수 있는 checkNode 함수를 구현하였다. 또한, 각각의 에러를 출력할 수 있는 함수를 구현하였다.

Figure . error detection

Figure . buildSymboltable

Figure . Error 출력

* 텍스트이(가) 표시된 사진

  자동 생성된 설명global.h – compound-stmt TreeNode 등에서 node를 통해 스코프에 접근해야할 때를 위해, 스코프 포인터 변수를 추가하였다.
* main.c – 명세를 따라 flag 들을 변경하였고, 출력을 위해 코드를 수정하였다.

텍스트이(가) 표시된 사진

자동 생성된 설명

1. Example and Result Screenshot

* 텍스트이(가) 표시된 사진

  자동 생성된 설명Test1 – gcd : 명세와 동일하게 정상 출력된다.
* 텍스트이(가) 표시된 사진

  자동 생성된 설명Test2 – Error check : 명세와 동일하게 에러를 탐지한다.