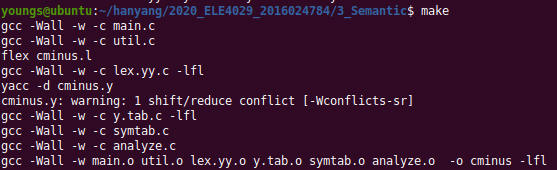
Compiler Report

2016024784 김영서

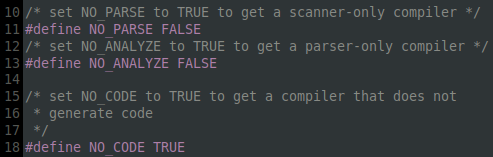
1. Compilation method and environment

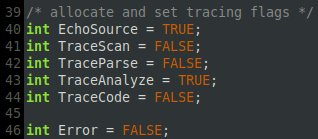
* Environment: Ubuntu 20.04.1 LTS, flex 2.6.4, bison (GNU Bison) 3.5.1
* Compile: Makefile을 이용하여 cminus 생성 (make 명령)



1. Explanation about how to implement and how to operate

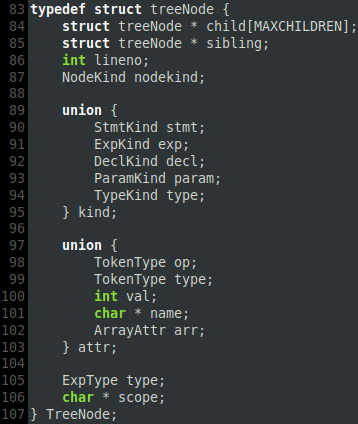
* main.c





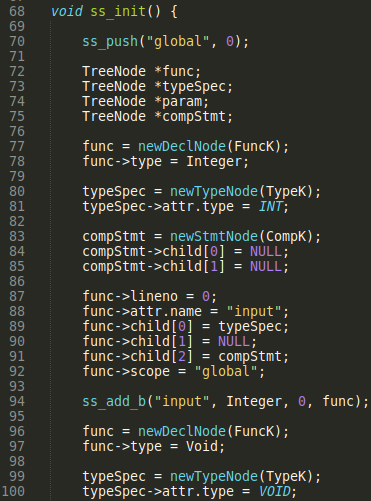
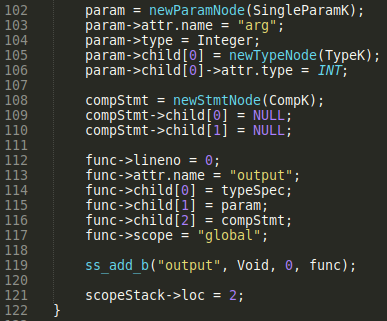
main.c의 플래그 수정

* globals.h



Type check를 위한 scope 추가

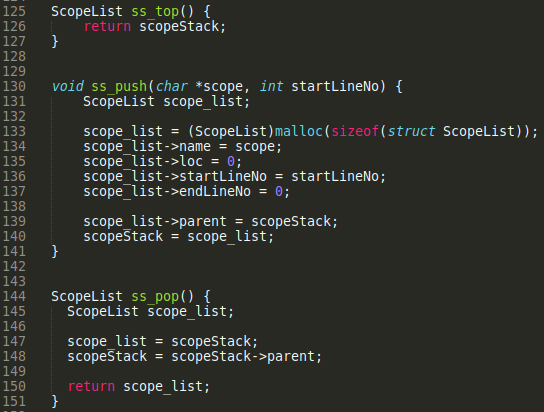
* symtab.c

ss\_init () 함수를 이용하여 ScopeStack을 초기화한다. 최초 global scope를 push한 뒤, scopeStack에 input과 output 추가.

이후 함수 선언 시 생성되는 TreeNode와 동일하게 typeNode, ParamNode, StmtNode를 FuncNode의 child로 추가하고 ss\_add\_b함수를 사용하여 ScopeStack에 추가.

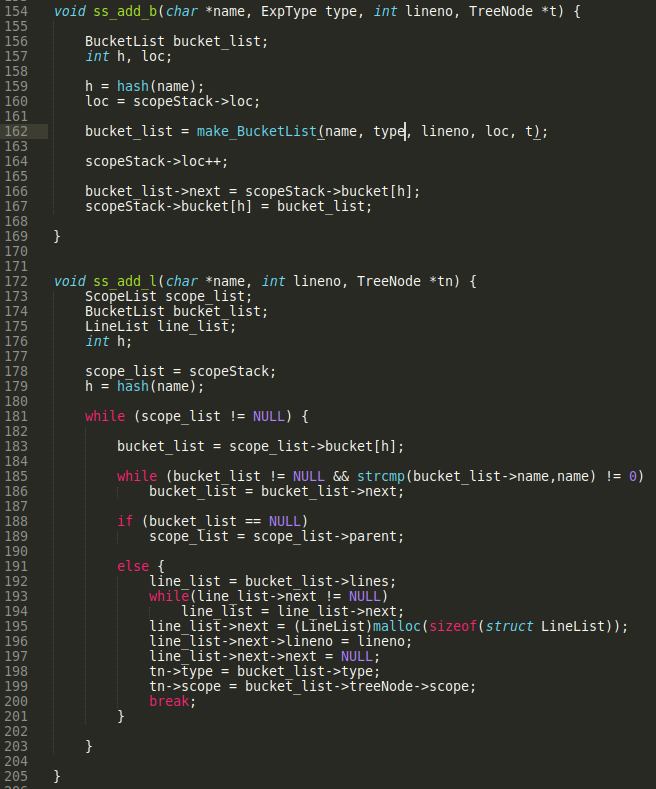
Global scope 함수들은 스택의 제일 밑에 위치하기 때문에 모든 scope에서 호출 가능.



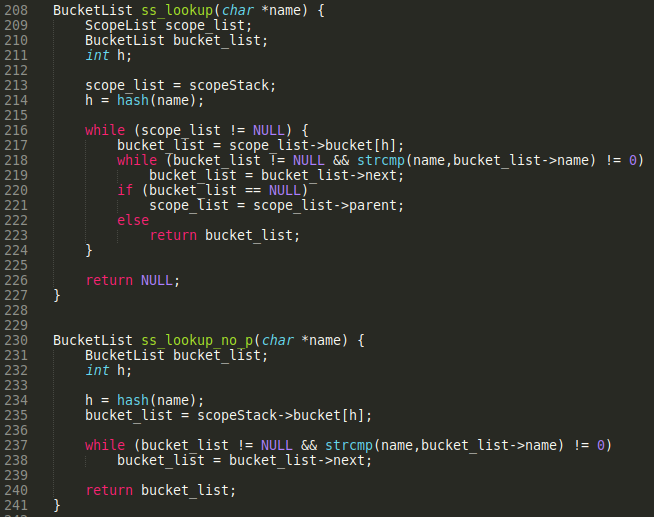
ss\_top을 통해 현재 scope에 대한 정보를 얻는다.

ss\_push를 통해 scope를 추가한다.

ss\_pop을 통해 scope 종료 후 scope를 symbol table에 추가한다.



ss\_add\_b, ss\_add\_l를 통해 새로운 BucketList를 scope의 적절한 위치에 추가한다. 만약 이전에 선언되어 있는 symbol을 호출 및 참조하는 경우, 해당 symbol scope의 LineList를 추가한다.

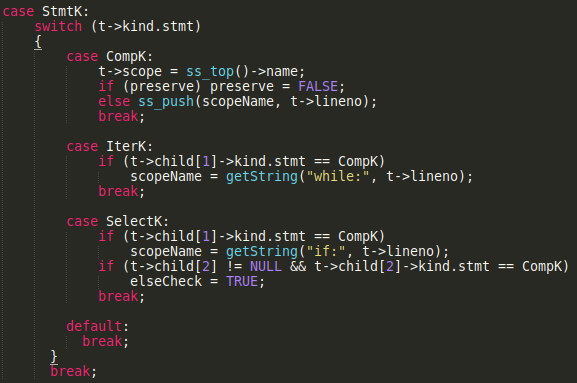


ss\_lookup은 ScopeStack을 확인하여 찾고자 하는 symbol의 유무를 확인하는 함수이다.

Ss\_loockup\_no\_p 함수는 찾는 이름을 가진 심볼이 scope에 있는지 알기 위한 함수이다.

St\_inset 함수 내부에서 ss\_pop을 이용해 심볼테이블이 ScopeList의 부모가 되어 주는 방식으로 scope의 symbol 정보를 symbol table에 추가하였다.

* Analyze.c



insetNode() 함수는 treeNode의 타입에 맞게 scope에 추가하는 역할을 수행하는 함수이다.

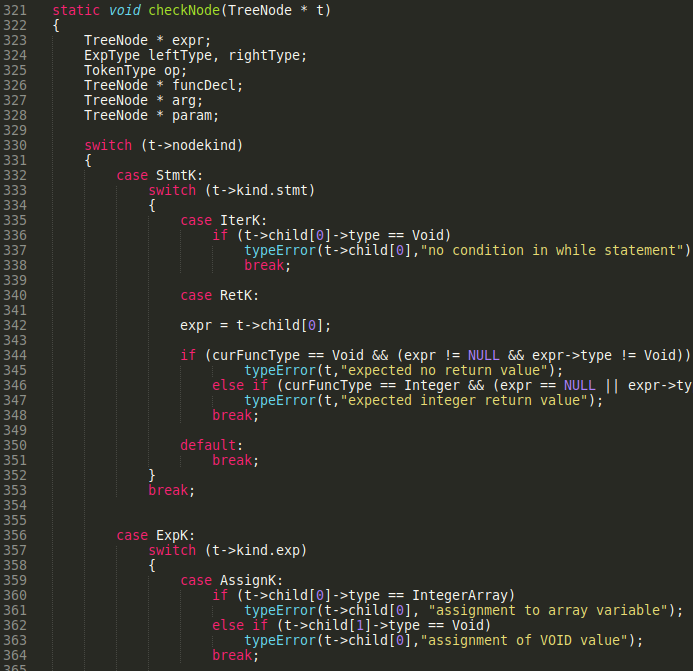
CompK 노드에 대한 예시이다.

Preserve 변수를 이용해 현재 노드의 scope 추가 여부를 확인한다. 새로운 함수 정의 시 각각의 case에 도달 전 preserve를 TRUE로 설정하여 함수 scope 추가를 제한한다.



ExpK 노드에 대한 예시이다.

현재 treeNode가 이미 존재하는 심볼에 대한 참조인 경우, lookup 함수를 이용해 에러 유무를 판단하였다.



노드의 type 에러를 확인하기 위한 함수이다. RetK 노드를 예시로 들면, child의 expression과 curFunType을 비교하여 다르다면 에러를 발생시킨다.

AssignK, OpK인 경우는 LHS, RHS의 타입을 비교해야 한다.

CallK의 경우는 각 파라미터와 args의 타입을 배교해야 한다.

1. Result

