Compiler Semantic Analysis Report

|  |  |
| --- | --- |
| 학부 | 컴퓨터소프트웨어학부 |
| 학번 | 2018008977 |
| 이름 | 이종범 |

# Environment

Compiler: clang 14.0.3

Flex: 2.6.4 Apple(flex-34)

OS: Sonoma 14.1

Device: Apple MacBook Pro M1 Max 32GB

# How to Run?

* 경로 이동 후 Compile

$ make all

* 파일 실행

$ ./cminus\_semantic [C-Minus File to Run]

# 주요 구현 사항

* analyze.c 파일의 스켈레톤 코드를 참고하여, Semantic Analysis에 필요한 checkNode, insertNode 함수를 완성함
* 제공되는 테스트 케이스의 통과 여부를 비교하기 위해, 자동화 테스트 Script를 작성하여 사용

# analyze.c 내 checkNode 함수

* 텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

  자동 생성된 설명Params
  + Void-Type Paramter, Redefined Variables Error을 탐지
  + Paramter의 Type을 확인하여 Void 또는 VoidArray인지 확인하고, 맞다면 VoidTypeVariableError를 일으킴
  + 현재의 Scope에서 SymbolTable을 검색하고, 동일한 이름의 Symbol이 존재한다면 RedefinedError를 일으킴
  + 모든 검사를 통과했다면 Symbol Table에 해당 Symbol을 삽입함
* VarAccessExpr

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

* + Symbol Table에 해당 Variable이 존재하는지 확인하고, 존재한다면 Symbol Table에 Line Number를 추가
  + 해당 Variable이 존재하지 않는다면, UndeclaredVariableError를 일으킴

# analyze.c 내 checkNode 함수

* IfStmt, WhileStmt

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

* + If 문 또는 While 문 내 조건이 명시되어 있지 않거나, Integer가 아닐 경우 InvalidConditionError를 일으킴
* ReturnStmt

텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

* + 함수의 Return State가 반환하고자 하는 Expression의 Type과 동일한지 확인하고, 그렇지 않다면 InvalidReturnError를 반환
  + 함수의 반환형이 Void일 경우에만 Return Statement의 Expression이 없는 것을 허용
* AssignmentExpr / BinOpExpr

텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

* + Left Child 또는 Right Child가 없거나, 양 Child의 Type이 일치하지 않을 경우에 예외를 일으킴
  + AssignExpr일 경우 InvalidAssignmentError를 일으키고, BinOpExpr의 경우 InvalidOperationError를 일으킴
* CallExpr

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

* + 텍스트, 스크린샷, 소프트웨어, 운영 체제이(가) 표시된 사진

    자동 생성된 설명Symbol Table 내에 호출하고자 하는 함수가 존재하는지 확인
  + Paramter가 함수가 필요로하는 Parameter와 동일한지 확인
  + paramNode와 argNode의 sibling을 순차적으로 탐색하여 Type이 일치하는지 확인
  + 일치하지 않는다면 InvalidFunctionCallError를 일으킴
  + Paramter의 수가 일치하지 않는다면 InvalidFunctionCallError를 일으킴
* VarAccessExpr

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

* + Symbol Table 내에 해당 Variable이 존재하는지 확인하고, 존재하지 않는다면 Error를 일으킴
  + 해당 노드에 Child가 존재하는 경우, Array인 경우로 판단함
  + Child가 존재하는 경우, Array인지 확인하고 Array가 아니라면 ArrayIndexingError2를 일으킴
  + Array라면, 해당 Child가 Index를 나타내기 위한 Integer Type인지 확인하고, Integer가 아니라면 ArrayIndexingError를 일으킴

# 자동화 테스트 Script 작성

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

* 해당 Project 3에서 testcase\_result.sh 파일을 통해 테스트를 자동으로 실행할 수 있는 스크립트를 제공하고 있음
* 하지만, 이는 동작만 자동화된 것으로, 많은 테스트 케이스에 있어 결과를 비교하기에는 오랜 시간이 소요되며, 휴먼 에러가 발생할 수 있을 것으로 판단함
* 이에 자동으로 테스트 결과를 확인할 수 있는 자동화 테스트 Script를 작성함

# Test

* $ sh ./compare\_test.sh 를 실행하여 테스트 진행
  + 실행 이전 $ sh ./testcase\_result.sh 를 실행하여 테스트 결과를 생성해두어야 함
* 테스트 실행 결과 모든 테스트 케이스에 대해서 차이가 없는 것을 확인

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명