컴파일러 Project 5-3 – 결과(설계)보고서 –

팀장 : 김석호

조원 : 민윤기, 권현송

## 조원(개인, 팀)

|  |  |  |  |
| --- | --- | --- | --- |
| 학번 | 이름 | 맡은 역할 | 비고 |
| 201502025 | 김석호 | GUI 구현, BytecodeGenListener 관련 추가 구현 |  |
| 201302267 | 민윤기 | PythonGenListener 관련 구현 및 변수명, 타입 에러 체크 기능 구현 |  |
| 201701198 | 권현송 | JavaGenListener 관련 구현 및 변수명, 타입 에러 체크 기능 구현 |  |

Github Repo 주소 – <https://github.com/SeokBA/Compiler_Term>

(현재 private이므로, 추후에 public으로 바꿔야 함)

## 주제 - 지금까지 배운 컴파일러 이론과 실습 지식들을 활용하여 MiniC 파일을 Bytecode 뿐만 아니라 다른 언어로의 변환을 시도하고자 Java Swing을 이용한 GUI 바탕의 MiniC 파일을 입력으로 하고 Antlr 라이브러리를 이용하여 Python, Java, Bytecode로 변환해 파일 형태로 출력하는 프로그램을 제작하는 것을 주제로 선정하였습니다.

## 구현 코드 및 방법 설명(이미지 첨부 가능, 함수 단위로 설명 가능)

1. GUI (TranslatorGUI.java)  
   사용자가 언어를 변환할 경우 사용이 편리하도록 하기 위해서 TranslatorGUI를 구현하였습니다. GUI는 Java GUI Library들을 이용하여 MiniC파일(.c)을 선택해 가져와서 변환할 언어를 선택한 뒤 선택한 저장 위치에 변환된 언어를 저장하는 방식으로 구현했습니다. 또한, walker 실행 중 또는 변환 중 오류가 탐지될 경우 오류 메시지를 GUI 내부의 console에 띄우도록 했습니다.
2. BytecodeGenListener.java 외 bytecode package 내부 코드  
   bytecode는 MiniC.g4에 맞춰 전번 기초구현에 더하여 해당 사항들을 추가하였습니다.  
     
   1) 배열 사용 가능  
   - SymbolTable에 VarInfo를 extends하는 class ArrInfo와 기타 배열 사용에 필요한 메소드들을 추가하여 배열을 저장하는 객체를 구현해 Listener에서 이용하도록 제작하였습니다.  
     
   2) 전역변수 사용 가능 (구현해야함)   
   -  
     
   3) tap 정렬 기능   
   - exitFun\_decl에서 블록안의 개행들을 전부 tap을 추가해주는 기능을 구현하였습니다.  
     
   4) 코드 최적화 - '\_' 명령 변환   
   - iload와 같은 명령어는 0~3까지는 \_명령을 사용하므로, 해당 명령어에 맞춰 변환하는 기능을 구현하였습니다.  
     
   5) 코드 최적화 - not (javap console과 최대한 비슷하게, 구현해야 함)   
   -
3. PythonGenListener.java 외 python package 내부 코드  
   -
4. JavaGenListener.java 외 java package 내부 코드  
   -

## 결과 요약

위의 주제를 구현하여 다음과 같이 구현하였습니다.

## 장, 단점 분석 및 아쉬운 점

- 기타 내용 추가가능

- 팀장 연락처 (이메일)

- 김석호 : cpswo123@naver.com