

컴파일러 (2015) Assignment 2

제출 기한: 9월 30일(수) 밤 23:59:59

사용 언어: Java

제출 방식: 보고서와 소스코드를 압축하여 업로드.

[분반]학번_이름.zip ([00]2010501234_홍길동.zip)

- 보고서 : 분반, 이름, 학번 명시.
예제 코드 실행 결과 캡처
추가로 작성한 C 코드와 그 실행 결과 캡처
문제 해결 방법 작성하기.
- 소스코드 : 설명이 필요한 부분이나 동작 원리를 주석 형태로 설명하기.

antlr 파일소스, 화면 캡처 등을 조교가 지정하는 방식으로 제출하시오.

과제내용

“antlr를 이용하여 간단한 문법에 대해 아래와 같이 매치된 규칙번호와 구성하는 규칙번호가 출력되도록 구문분석기를 구현하시오”

“압축 파일에 포함된 예제 C 코드를 구문분석기로 분석하시오”

“제시된 예제 코드 이외에 구문분석기가 해석 가능한 C 코드를 작성하시오. 단 Rule 0번부터 15번까지 출력되어야 함. (Rule 1-1, Rule 1-2, 가 있다면 Rule 1-1만 출력해도 Rule 1 출력으로 인정) 만약 출력 불가능한 Rule이 있다면 그 이유를 서술하시오.”

예) MiniC.g4

```
grammar MiniC;
program      : decl+           {System.out.println("Rule 0");};
decl         : var_decl        {System.out.println("Rule 1-1");};
              | fun_decl        {System.out.println("Rule 1-2");};
var_decl     : type_spec IDENT ';' {System.out.println("Rule 2-1");};
              | type_spec IDENT '[' ']' ';' {System.out.println("Rule 2-2");};
```

....
.... // 아래는 lex 부분

```
VOID: 'void';
INT: 'int';
WHILE: 'while';
IF: 'if';
ELSE: 'else';
RETURN: 'return';
OR: 'or';
AND: 'and';
LE: '<=';
GE: '>=';
EQ: '==';
NE: '!=';
```

```

IDENT : [a-zA-Z_]
      ( [a-zA-Z_]
      | [0-9]
      )*;
LITERAL: DecimalConstant | OctalConstant |
HexadecimalConstant ;
DecimalConstant
: [1-9] [0-9]*
;
OctalConstant
: '0' [0-7]*
;
HexadecimalConstant
: '0' [xX] [0-9a-fA-F] +
;
WS : ( ' '
      | '\t'
      | '\r'
      | '\n'
      )+
    -> channel(HIDDEN)
;

```

주의.

- lexical analysis 부분에서는 출력하지 않는다.
- ANTLR는 각자의 컴퓨터에서 jar를 다운받아서 사용 (www.antlr.org)
- Eclipse 를 사용하고, Antlr 플러그인을 설치해서 사용하기를 권장함 (Eclipse메뉴에서 Help - Eclipse market place 에서 “ANTLR”를 키워드로 찾은 후 다운로드)
- Eclipse 메뉴에서 Project - Properties에서 ANTLR4 - Tool - Options에서 directory를 적절히 조절하면 어떤 디렉토리에 파서를 생성시킬지 결정할 수 있음.
- 내가 작성한 메인을 돌리려면, Run - Run Configurations에서 적절한 main 클래스를 선택 (ANTLR 의 디폴트 테스트인 TestRig 가 수행되는 경우가 있음)
- 그 외 자세한 사항은 조교가 업로드하는 추가자료를 참고할 것 (문법 등)

테스트를 위한 main 메소드는 아래와 같다.

```

import org.antlr.v4.runtime.*;
public class TestMiniC {
    public static void main(String[] args) throws Exception
    {
        MiniCLexer lexer = new MiniCLexer( new ANTLRFileStream("test.c"));
        CommonTokenStream tokens = new CommonTokenStream( lexer );
        MiniCParser parser = new MiniCParser( tokens );
        ParseTree tree = parser.program();
    }
}

```