**Principles of Programming Languages**

Programming Assignment #1

20154098 염승윤

Class 구성

Main class

설명: 프로그램 시작점으로 파일 이름이 인자로 입력되면 파일을 읽어 LexicalAnalyzer 객체로 파일 내용을 넘겨줍니다. -v 파일 이름이 인자로 입력되면 Vmode 객체로 파일 내용을 넘겨줍니다.

LexicalAnalyzer class

설명: 정해진 문법에 따르는 지 판단하는 Parser를 Recursive Descent Parsing 기법을 이용하여 작성했습니다.

변수

Int next\_token: 토큰을 정수형으로 저장합니다.

String token\_string: 토큰을 문자열 변수로 저장합니다.

HashMap<String, String> simbolTable: 변수 이름과 변수가 가지고 있는 값을 저장합니다.

String state: 파싱한 결과에 따라 ok, warning, error 메시지를 저장합니다.

String code: Main class에서 보낸 코드를 임시 저장합니다.

메소드

lexical: 입력 스트림을 분석하여 lexeme을 찾고 분석결과를 next\_token에 저장합니다.

PROGRAMS: simbolTable을 생성하고 STATEMENTS 문법을 시작합니다.

STATEMENTS: lexical 메소드를 실행하여 세미콜론이 나올 때까지 코드를 계속 가져옵니다.

STATEMENT: 변수와 EXPRESSION 메소드의 반환 값을 simbolTable에 저장합니다. 만약 변수 := 형태가 아니라면 에러메시지를 출력합니다. 정의되지 않은 변수가 있음 Unknown 값으로 선언됩니다.

EXPRESSION: +, - 연산자를 기준으로 코드를 나누어 TERM과 TERM\_TAIL 메소드로 전달 합니다.

TERM: (가 있다면 )를 찾고 그 사이 코드를 EXPRESSION 메소드로 전달합니다. 만약 \*, / 연산자가 있다면 \*, / 연산자를 기준으로 코드를 나누어 FACTOR와 FACTOR\_TAIL 메소드로 전송합니다.

FACTOR: 변수(IDENT)를 simbolTable에서 찾아 값을 반환해 줍니다. 이때 simbolTable에 변수가 없다면 값이 Unknown으로 저장하고 Unknown을 반환합니다. 상수(CONST)가 있다면 그 상수를 반환합니다.

TERM\_TAIL: +, - 연산자가 있는지 확인하고 TERM, TERM\_TAIL 메소드로 전달합니다. +, - 연산자가 중복 되어있으면 맨 앞에 있는 연산자로 계산합니다.

FACTOR\_TAIL: \*, / 연산자가 있는지 확인하고 FACTOR, FACTOR\_TAIL 메소드로 전달합니다. \*, / 연산자가 중복 되어 있으면 맨 앞에 있는 연산자로 계산합니다.

splitInt, splitStr: 프로그램 만들던 중에 ArrayList로 선언한 변수를 분할해야 할 때가 있어 (괄호 EXPRESSION 괄호 등등의 문법) start, finish 인덱스를 입력해주면 start~finish-1로 분할해주는 메소드입니다.

validVal: 문법에 해당하는 메소드들 반환 값으로 연산을 진행하는데 Unknown 값이 있는 경우 연산을 할 수 없어 따로 처리하기위해 만든 메소드입니다.

Vmode class

설명: 프로그램 실행할 때 -v 옵션을 주면 이 class가 실행됩니다. LexicalAnalyzer 객체와 같은 코드로 구성되어 있지만 토큰만 출력하도록 구현되어 있습니다.

변수

LexicalAnalyzer class와 동일합니다.

메소드

LexicalAnalyzer class 중에서 토큰이 변할 때 어떤 토큰인지 화면에 출력하는 기능만 구현되어 있습니다.

Error 메시지 나오는 경우

1. 정의되지 않은 변수가 사용되었을 때

simbolTable에 변수 이름이 없으면 Unknown으로 처리합니다.

2. 상수에 값을 대입하려고 할 때

1 := 4 + 2 ; 처럼 상수에 값을 대입하려고 하면 Error 메시지가 나옵니다.

3. 대입 연산자 뒤에 피 연산자가 없을 때

변수 := ; 처럼 대입 연산자가 있는데 입력 대입하려는 값이 없을 때 Error 메시지가 나옵니다.

4. 대입 연산자 없이 변수만 있을 때

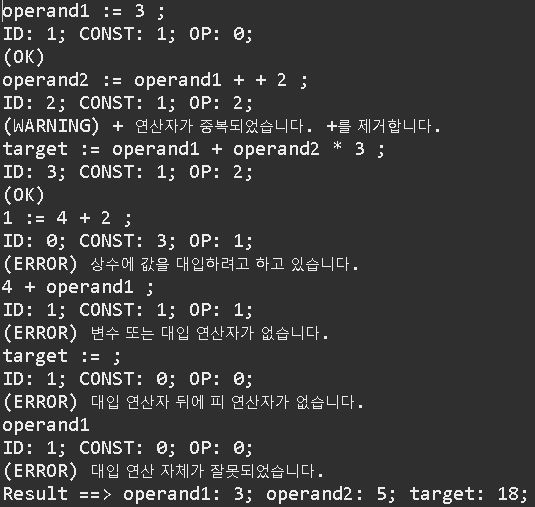
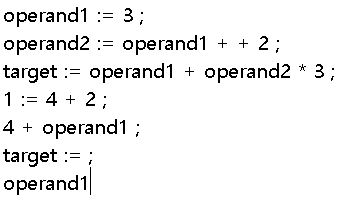
변수 ; 처럼 대입 연산자가 없을 때 Error 메시지가 나옵니다.

Warning 메시지 나오는 경우

1. 연산자 2개가 연달아 나오는 경우

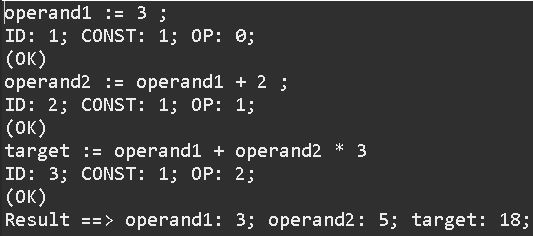
+ + 나 - \* 처럼 2개 연산자가 연달아 나오는 경우 맨 앞에 있는 연산자로 계산하고 나머지 연산자는 제거합니다.

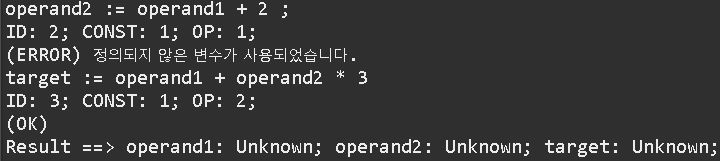
예외처리 예시 사진



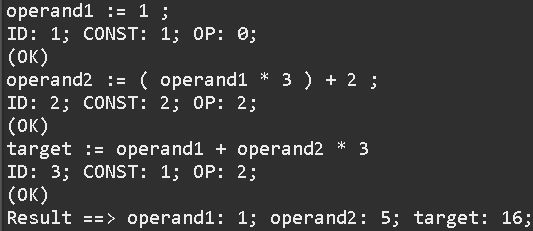
처리 예 실행 사진

eval1.txt



eval2.txt

eval3.txt



v 옵션 실행 예 사진

-v eval1.txt -v eval2.txt -v eval3.txt

