

# 프로그래밍언어론

실습과제 5 Lexical Analyzer 구현

제출기한 4월 20일(월) 오전 9시

Clite의 Lexical Analyzer를 C로 구현한다. 조별 토론후 각자 자신의 Lexer를 구현하여 제출한다.

## 토론

1. 이 언어의 Token을 regular expression으로 명세하라.
2. Token을 인식하는 Finite State Machine을 그려보라.
3. Source program에 오류가 있을 경우 어떻게 하는 것이 좋을까 생각해보라.
4. Lexical level과 Syntax level의 오류는 어떤 차이가 있는가?
5. Compiler가 만나는 오류의 종류를 level 별로 분류할 수 있는가? 그 종류를 나열해보라.
6. Lexer는 syntax level의 오류를 어떻게 처리하는 것이 좋을 지 생각해보라.

## 구현

3. 함수 Token getToken(); 를 구현하라. 이 함수는 parser가 next token을 필요로 할 때 사용하는 함수로, 입력 파일에 있는 다음 토큰을 반환한다.
4. 타입 Token의 자료구조를 설계하고 필요한 토큰을 상수로 정의하라.

프로그램 lexer.exe는 다음 예시처럼 입력 source program을 분석하여 token과 줄 번호를 출력한다. getToken() 함수를 이용하여 이 프로그램을 구현하라.

## 입력 파일

```
// this is a comment to be ignored
int a, b;
a245= 53 + b + 42.345;
// a=b+ c+ b
if (a245<= (182957) Variable25.12 ==what1919$;
for (i = 0; a < x; i++) {
    a = a*2;
```

## 출력

```
Line 1 // this is a comment to be ignored
Line 2 int a, b;
Int
Id a
Id b
;
```

```

Line 3 a245= 53 + b + 42.345;
      Id a245 =
      IntLiteral 53
      +
      Id b
      +
      FloatLiteral 42.345
      ;
Line 4 // a=b+c+b
Line 5 if (a245<= (182957) Variable25.12 ==what1919$;
If
(
... 이하 생략

```

#### 제출물 (LMS 제출)

1. 질문 5, 6에 대한 답
2. lexer.exe의 Source program
3. 위 입력 프로그램에 대한 출력 (비공개 소스 프로그램으로 검사 예정)
4. 자신이 직접 과제를 수행했음을 보여줄 설명 - 어려웠던 점, 느낀 점 ..