**2023.2   
Principle of Programming Language  
Programming Assignment #1**

**Internal Documentation**

|  |  |
| --- | --- |
| 20184256 | 박성민 |
| 20226041 | 김규리 |
|  |  |
|  |  |

**전반적인 프로그램의 동작 절차**

작성한 분석 프로그램은 main 함수 호출 인자에서 filename을 전달받아 입력 파일 스트림을 생성하면서 시작된다. 만약 파일을 열 수 없다면 에러메시지를 출력하고 프로그램을 종료한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

그 후, LexicalAnalyzer 클래스의 객체를 생성하고 멤버 변수 analyzeInputFile 함수에 파일스트림을 인자로 전달해 토큰에 대한 분석을 진행한다. 토큰 분석 결과와 그 과정에서 생성한 symbolTable을 인자로 받아 Parser 객체를 생성해준 뒤, 입력에 대한 구문 분석을 진행한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

모든 처리가 완료되면 파일스트림을 닫고 프로그램을 종료한다.

텍스트, 스크린샷, 폰트, 블랙이(가) 표시된 사진

자동 생성된 설명

**Lexical Analyzer의 동작**

설계한 Lexical Analyzer 클래스의 인터페이스는 아래와 같다. 주어진 파일 스트림에 대한 토큰 분류 결과를 담는 \_lexResult, 그 과정에서 생성된 심볼 테이블을 담는 \_symbolTable, 입력 문자열에 대해 토큰을 분류하는 analyzeString 함수와, 로직 처리를 시작하고 결과를 반환하는 함수들로 구성되어 있다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

주어진 파일의 코드를 토큰으로 분류하기 위해, 아래처럼 토큰을 enum 타입으로 정의하였다.

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

아래는 입력된 문자열에 대해 특수 기호가 아닌 숫자, 알파벳, \_이 연속으로 입력된 경우, 이를 IDENT, CONST, UNKNOWN으로 분류해주는 함수이다. 숫자로 시작하지 않으면 IDENT로, 숫자로 시작할 때 모든 문자가 숫자이면 CONST로, 그 외에는 UNKNOWN으로 분류한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

아래는 주어진 파일스트림을 읽으면서 토큰을 분류하는 함수이다. 파일에 남은 문자가 없을 때 까지 한 글자씩 반복하며 진행한다. line 변수에는 현재 읽은 문자가 혼자서 토큰이 될 수 없는 경우(IDENT, CONST)를 저장하는데에 사용된다.

받아온 글자가 공백문자인 경우에는 line에 있는 문자열의 토큰을 분류하고 결과에 추가한다.

받아온 글자가 알파벳, 숫자, \_ 인 경우에는 line에 추가해 다음 문자와 함께 처리한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

나머지 기호들에 대해서는 앞서 입력받던 line에 대해 처리를 해주고 나서 입력에 해당하는 토큰으로 분류해 결과에 추가해준다. 주어진 문법에 없는 기호의 경우 UNKNOWN으로 처리하였다.

텍스트, 스크린샷, 디자인이(가) 표시된 사진

자동 생성된 설명

모든 입력을 다 처리한 뒤에는 마지막으로 line에 남아있는 문자열을 처리한 뒤, END\_OF\_FILE 토큰을 결과에 넣고 분석 결과를 저장한다..

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

**Parser의 동작**

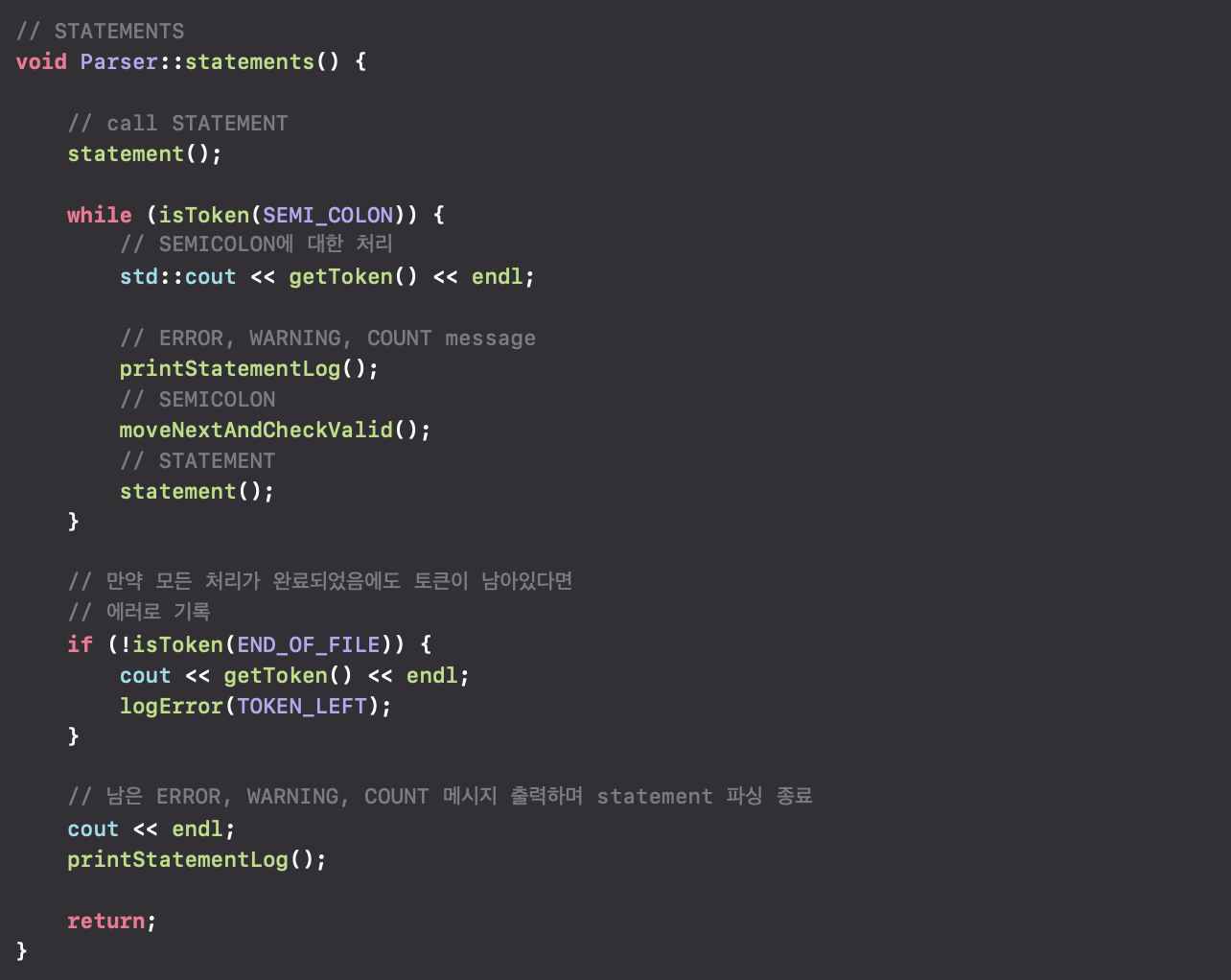
Parser에서 Program함수를 실행하면서 파싱을 시작한다. Program 함수에서는 값을 초기화하고 statements 함수를 호출한 뒤 파싱이 완료된 뒤 Symbol Table의 데이터를 출력하면서 함수를 종료한다..

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

­­­

statements 함수에서는 statement 함수를 호출해 하나의 statement를 처리한다. 이후 세미콜론이 나올 때 초기화를 진행하고 statement 함수를 다시 호출해 처리한다. 모든 statement에 대한 처리가 끝나면 END\_OF\_FILE 토큰을 확인하고 처리를 종료한다.



모든 입력을 다 처리한 뒤에는 마지막으로 line에 남아있는 문자열을 처리한 뒤, END\_OF\_FILE 토큰을 결과에 넣고 분석 결과를 저장한다..