GHULAM ISHAQ KHAN INSTITUTE OF
ENGINEERING SCIENCES AND TECHNOLOGY

Lexical Analyzer MiniLang

CS424 (CS) Compiler Construction - Assignment 1

Submitted By: 2020515 Wardah Tariq

Submitted to: Sir Usama Janjua

I. Introduction

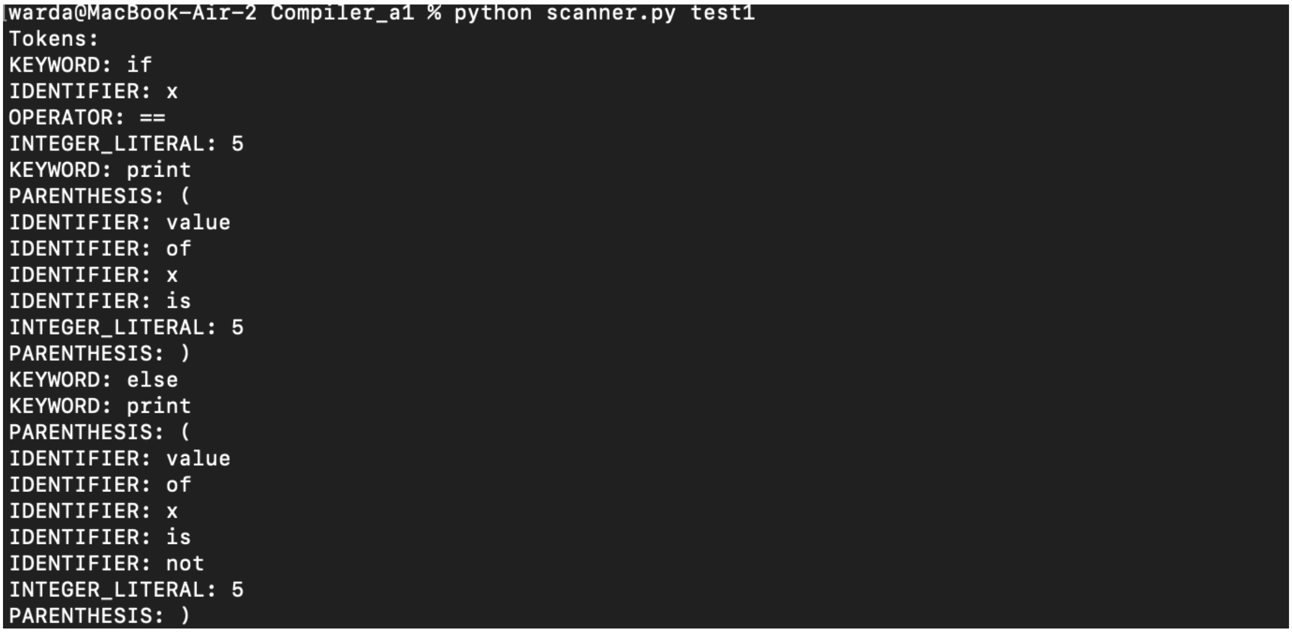
The scanner takes as input a source file that contains the contents that we want to read. The contents of the file are sent to our scanner also known as the lexical analyzer which then reads the contents of the file and breaks it down into lexemes. Each lexeme contains the token along with the detected tokens value.

II. Tokens

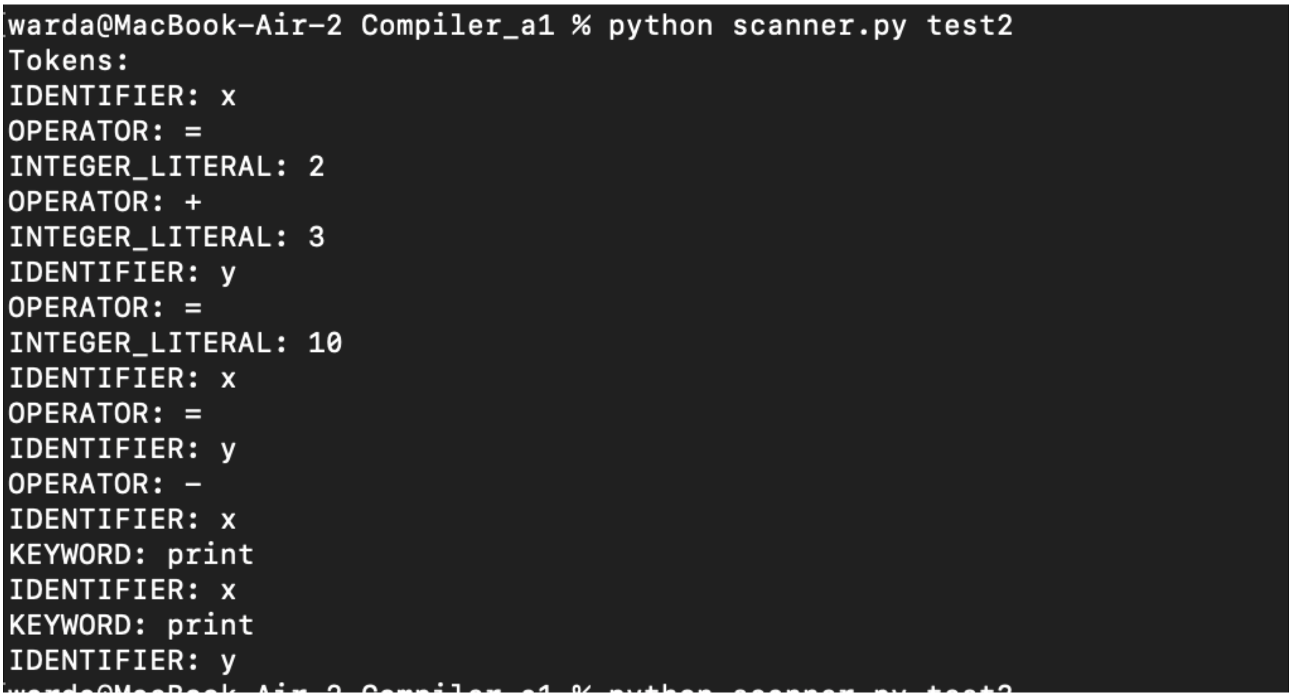
Our tokens will be categorized into 6 different types which are explained below:

1. Keywords: These are the other reserved keywords that represent certain actions such as print, true, false, if and else.
2. Identifiers: An identifier represents the name of the variable. For example, in the line int age = 5, age is the keyword.
3. Operators: These represent the different type of operators in our program that preform mathematical operations.
4. Parenthesis: We also have included tokens for parenthesis that are used to define the scope of a function or statement.
5. Bool Literals: These are literals representing boolean values (true or false)
6. Integer Literals: These are integer values that are explicitly specified in the code. For example, in the line int age = 5; "5" is an integer literal.

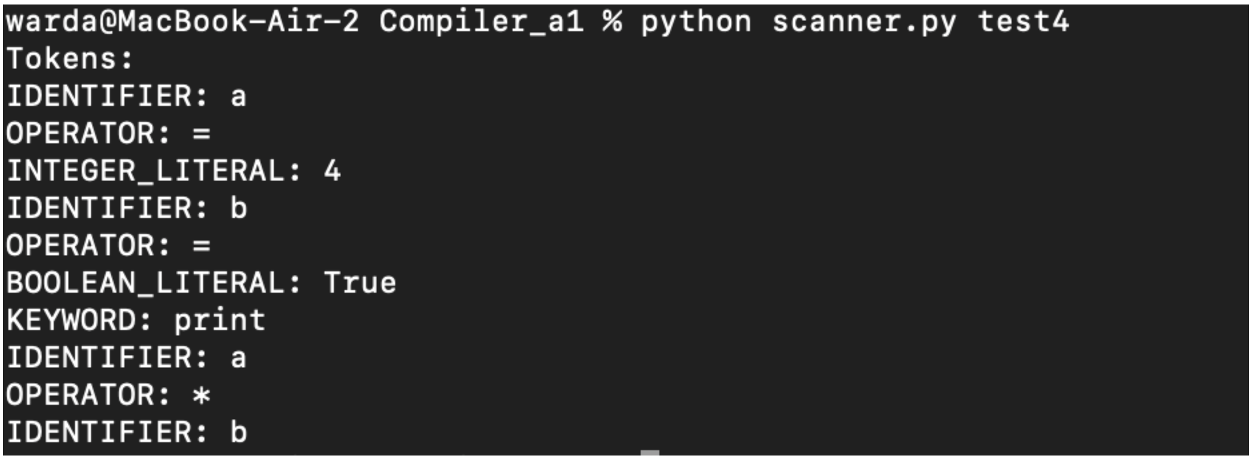
III. Test Cases



```
warda@MacBook-Air-2 Compiler_a1 % python scanner.py test1
Tokens:
KEYWORD: if
IDENTIFIER: x
OPERATOR: ==
INTEGER_LITERAL: 5
KEYWORD: print
PARENTHESIS: (
IDENTIFIER: value
IDENTIFIER: of
IDENTIFIER: x
IDENTIFIER: is
INTEGER_LITERAL: 5
PARENTHESIS: )
KEYWORD: else
KEYWORD: print
PARENTHESIS: (
IDENTIFIER: value
IDENTIFIER: of
IDENTIFIER: x
IDENTIFIER: is
IDENTIFIER: not
INTEGER_LITERAL: 5
PARENTHESIS: )
```



```
warda@MacBook-Air-2 Compiler_a1 % python scanner.py test2
Tokens:
IDENTIFIER: x
OPERATOR: =
INTEGER_LITERAL: 2
OPERATOR: +
INTEGER_LITERAL: 3
IDENTIFIER: y
OPERATOR: =
INTEGER_LITERAL: 10
IDENTIFIER: x
OPERATOR: =
IDENTIFIER: y
OPERATOR: -
IDENTIFIER: x
KEYWORD: print
IDENTIFIER: x
KEYWORD: print
IDENTIFIER: y
warda@MacBook-Air-2 Compiler_a1 % python scanner.py test3
```



```
warda@MacBook-Air-2 Compiler_a1 % python scanner.py test4
Tokens:
IDENTIFIER: a
OPERATOR: =
INTEGER_LITERAL: 4
IDENTIFIER: b
OPERATOR: =
BOOLEAN_LITERAL: True
KEYWORD: print
IDENTIFIER: a
OPERATOR: *
IDENTIFIER: b
```