KATHMANDU UNIVERSITY

SCHOOL OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

LAB REPORT 1

A **Fourth year/ First Semester** Compiler Design [COMP 409]
Assignment submitted in partial fulfillment of the requirements
for the degree of Bachelor of Engineering.

Submitted by:
Ashish Pokhrel
Faculty: C.E.
Roll: 38
Registration No: 022446-17

Submitted to:
Mr. Sushil Nepal
Compiler Design (COMP 409)
Department of Computer Science and Engineering

June 16, 2021

LAB 1

Write a program to read a file named File1.txt that contains one executable program and list out all the tokens specified in that File1. Txt

Source Code:

lab1_comp409.py

```
# This lab report 1 is submitted by:::
#   Name: Ashish Pokhrel
#   Class: C.E.
#   Registration no: 022446-17
#   Roll: 38


import keyword

def check_keyword(arg):
    keyword_list = keyword.kwlist

    if arg in keyword_list:
        return True
    else:
        return False


def is_valid_delimiter(arg):
    del_list = [" ", "(", ")", "[", "]", "{", "}", ",", ":", ".",
                "`", "=", ";", "+=", "-=", "*=", "/=", "%=",
                "**=", "&=", "|=", "^=", ">>=", "<<="]

    if arg in del_list:
        return True
    else:
        return False


def is_separator(arg):
    sep_list = [".", ",", ";", "(", ")", "{", "}", "[", "]"]

    if arg in sep_list:
        return True
    else:
        return False
```

```
def is_valid_operator(arg):
    op_list = ["+", "-", "*", "**", "/", "%", "<<", ">>",
               "&", "|", "^", "~", ">", "<=", ">=", "==",
               "!=", "<>", "="]

    if arg in op_list:
        return True
    else:
        return False

def is_valid_integer(arg):
    if len(arg) == 0:
        return False

    int_list = [str(x) for x in list(range(10))]

    for x in arg:
        if x not in int_list:
            return False

    return True


def search_token(arg, tokens):
    if arg in tokens:
        return True
    else:
        return False


def detect_token(arg):
    left, right = 0,0
    length = len(arg) - 1
    tokens = []

    while (right <= length and left <= right):
        if (not is_valid_delimiter(arg[right])):
            right += 1

        if (is_valid_delimiter(arg[right]) and left == right):
            if (is_valid_operator(arg[right])):
                if (not search_token(arg[right], tokens)):
                    tokens.append(arg[right])
```

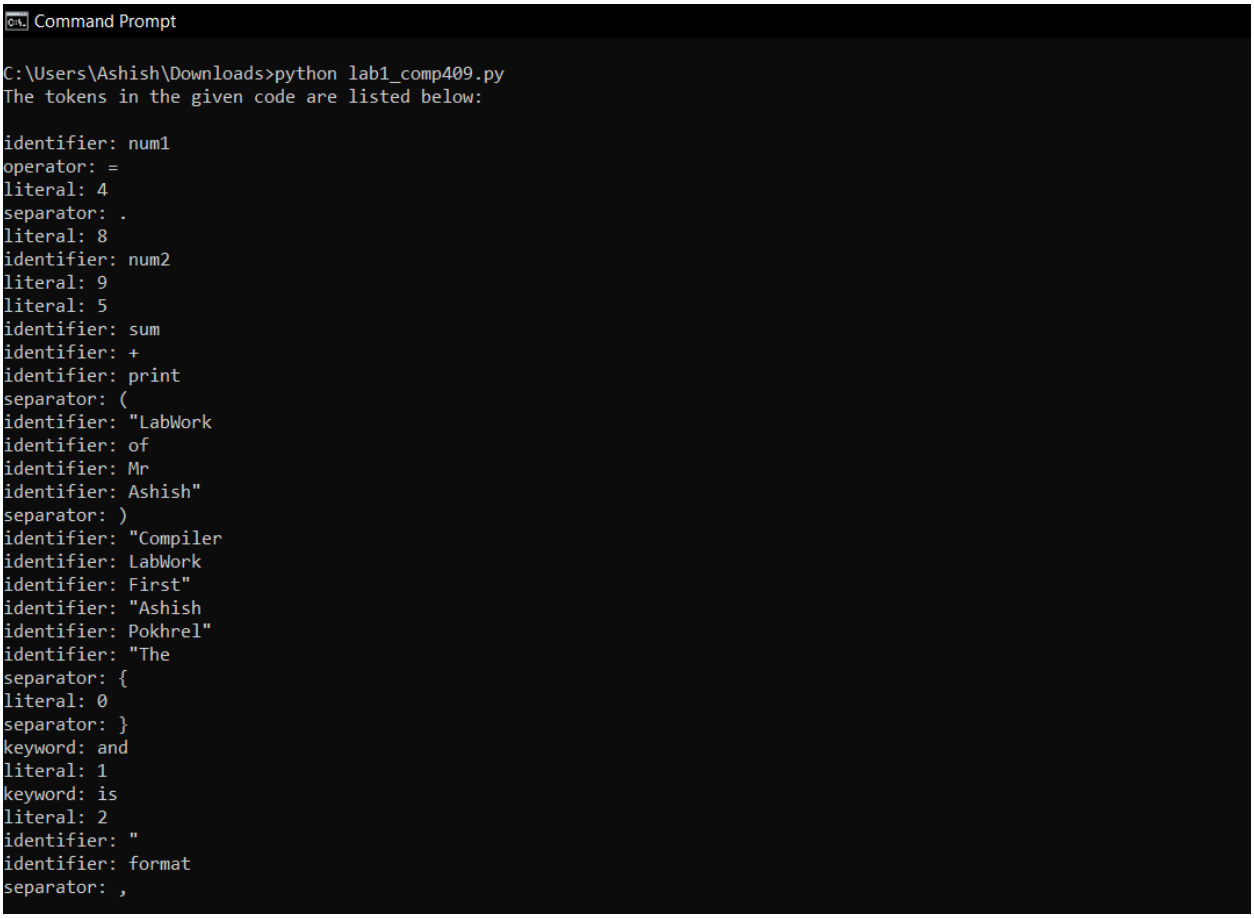
```
                    print(f"operator: {arg[right]}")

            elif (is_separator(arg[right])):
                if(not search_token(arg[right], tokens)):
                    tokens.append(arg[right])
                    print(f"separator: {arg[right]}")

            right += 1
            left = right


        elif (is_valid_delimiter(arg[right]) and left != right or
                (right == length and left != right)):

            sub = arg[left:right]

            if (not search_token(sub, tokens)):
                if (check_keyword(sub)):
                    tokens.append(sub)
                    print(f"keyword: {sub}")

                elif (is_valid_integer(sub)):
                    tokens.append(sub)
                    print(f"literal: {sub}")

                else:
                    tokens.append(sub)
                    print(f"identifier: {sub}")

            left = right



if __name__ == "__main__":
    file = open('File1.txt', 'r')
    filestr = str(file.read())

    ign_list = [" ", "\n"]

    code = ""

    for x in filestr:
        if x not in ign_list:
            code += x
        else:
```

```
            code += " "

    print("The tokens in the given code are listed below: \n")
    detect_token(code)
```

File1.txt

```
num1 = 4.8
num2 = 9.5
sum = num1 + num2


print("LabWork of Mr. Ashish")
print("Compiler LabWork First")
print("Ashish Pokhrel")
print("The sum of {0} and {1} is {2}".format(num1, num2, sum))
```

Output:


Command Prompt
C:\Users\Ashish\Downloads>python lab1_comp409.py
The tokens in the given code are listed below:
identifier: num1
operator: =
literal: 4
separator: .
literal: 8
identifier: num2
literal: 9
literal: 5
identifier: sum
identifier: +
identifier: print
separator: (
identifier: "LabWork
identifier: of
identifier: Mr
identifier: Ashish"
separator:)
identifier: "Compiler
identifier: LabWork
identifier: First"
identifier: "Ashish
identifier: Pokhrel"
identifier: "The
separator: {
literal: 0
separator: }
keyword: and
literal: 1
keyword: is
literal: 2
identifier: "
identifier: format
separator: ,