Lab 2: Functions

SUNY Korea - Francois Rameau

GitHub Classroom

Lab 2 - Functions in python

Functional programming

Functional programming is a programming paradigm where everything is a function

Even arithmetic operators

Even assignments

Even function definitions

LISP

LISP is an ancient functional programming language

```
# Python code
def foo(x):
    a = x * 5
    b = (6 + a) / 3 ** 2
    print(b)
```

```
;; LISP code (line-by-line)
(defun foo (x)
    (let ((a (* x 5))
        (b (/ (+ 6 a) (pow 3 2))))
      (print b))))
```

Functional programming

In this lab, we will mimic a functional programming style!

Note: Keep in mind that it's **highly impractical** to mimic like this, however the purpose of this exercise is to familiarize yourselves with functions

Of course, we can't do everything as a function: assignments and function definitions will remain. We will only work on arithmetic operations

Arithmetic operations

`+` → `add()`
`-` → `sub()`
`/` → `div()`
`**` → `exp()`
`*` → `mult()`

How are we going to implement them?

Functional programming

Example when replacing original operators with functions

```
# Python code
a = 5
b = (6 + a) / 3 ** 2
print(b)
```

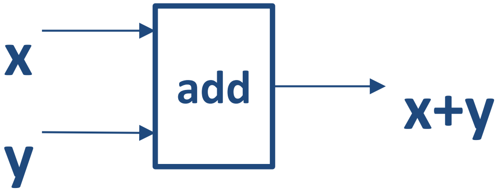
To functional

```
# Functional variant
a = 5
b = div(add(6, a), exp(3, 2))
print(b)
```

What is the behavior of add(a,b)?

What is the behavior of add()

- How many inputs?
- How many outputs?
- What will be the output?



```
# add()
def add(a,b) :
    return a + b
```

Task 1

Implement the following functions

Sub()	Div()	Mult()	exp()
-	/	*	**

exp(2,4) = 2 * 2 * 2 * 2

Also add neg() and sqrt()

- E.g., neg(5) should return -5, neg(-5) should return 5
- E.g., sqrt(16) should return 4

Follow the comments in “func.py”

Conditions

- Do not import anything
- Do not use any techniques not taught in class yet

Task 2

The next task is to reproduce the result of task 1 of Lab 1 : the quadratic formula

Except this time, use only the six functions above (variable assignments can be done as usual, of course)

Conditions

- Do not import anything
- No outside function
- No arithmetic operators outside your functions