
4주차 : 함수

커리큘럼

1. 변수, 입출력
2. 조건문, 반복문
3. 리스트, 튜플, 세트, 딕셔너리
4. 함수
5. class
6. 알고리즘 입문, 그리디 알고리즘
7. 재귀함수
8. 탐색
9. DP(Dynamic Programming)

함수의 필요성

✓ 같은 코드를 반복해서 만드는 것은 비효율적 → 미리 만들어 놓기

```
>>> def plus(a,b):
        s = a + b
        return s

>>> plus(1,2)
3
>>> plus(2.2,3)
5.2
>>> plus(1,-2)
-1
```

def 함수명(입력 파라미터):

    명령문1

    명령문2

    (return 리턴값)

return 이 있으면 함수를 변수로 활용 가능

파라미터(매개 변수)

```
>>> def func1(n,mylist):
        n=n+1
        mylist.append(0)

>>> n=1
>>> mylist=[1]
>>> func1(n,mylist)
>>> n
1
>>> mylist
[1, 0]
```

- ✓ list, dict 등과 같은 자료형(Mutable)은 따로 특별한 과정을 거치지 않아도 함수 내에서 처리한 결과가 그대로 나온다.

- ✓ int, string, tuple 등과 같은 자료형(Immutable)은 함수 내에서 결과가 처리 되지 않는다.

지역 변수 → 전역 변수 로 바꿔줘서 해결

전역 변수

```
>>> def func2(mylist): #
        global n
        n=n+1
        mylist.append(0)

>>> n=1
>>> mylist=[1]
>>> func2(mylist)
>>> n
2
>>> mylist
[1, 0]
```

- ✓ list, dict 등과 같은 자료형(Mutable)은 따로 특별한 과정을 거치지 않아도 함수 내에서 처리한 결과가 그대로 나온다.

- ✓ int, string, tuple 등과 같은 자료형(Immutable)은 함수 내에서 결과가 처리 되지 않는다.

지역 변수 → 전역 변수 로 바꿔줘서 해결

파라미터(매개 변수)

```
def func3(a,b,c=1):
    return (a/b)*c

result_1 = func3(3,4)
result_2 = func3(3,4,2)
result_3 = func3(b=2, a=4)
print(result_1, result_2, result_3)
# 0.75 1.5 2.0
```

```
def func4(*num):
    result = 0
    for i in num:
        result += i
    return result

print(func4(1,3,2,4,5)) # 15
```

모듈

✓ 파이썬 코드를 묶어서 관리하고 사용할 수 있도록 하는 것

✓ 모듈의 이름 : 파이썬 py 파일

✓ 효율적이고, 다른 파일에 똑같은 코드를 안 적어도 되는 것이 장점

```
import math

a = math.sqrt(2)
print(a) # 1.4142135623730951
```

작성 방법

```
import math as m

a = m.sqrt(2)
print(a) # 1.4142135623730951
```

```
from math import sqrt as s

a = s(2)
print(a) # 1.4142135623730951
```

```
# funcpractice.py
def func1(...):
    ...
def func2(...):
    ...
def func3(...):
    ...
def func4(...):
    ...
```



```
from funcpractice import *
```

모듈명(파일명)은 변수 규칙과 같습니다.

1. 기존 지정된 이름 불가능
2. 영어, 숫자, _ 조합으로 구성
3. 시작을 숫자로 구성 불가능

알아 두면 좋은 모듈

모듈명	기능	대표적인 함수
time	시간과 관련 기능 제공	.time()(현재 시각, 0.000001초단위), ...
math	수학적인 함수 제공	.cos(), .sin(), acos()(삼각함수의 역함수), ...
random	임의의 값을 제공	.randint(a,b)(a에서 b까지 임의의 정수), ...


과제

4-1

어떤 방정식 $xe^{-x} = 0.2$에 대해서 구간 $[a, b]$ 에 해가 존재하는 것이
확실한지 불확실한지 따져 보기 바랍니다. 이를 위해서 아래의 정리를 사용할 수 있습니다.

함수 $f(x)$가 구간 $[a, b]$ 에서 연속이면
$f(a)$ 와 $f(b)$ 사이의 임의의 값 k에 대하여,
즉 $f(a) < k < f(b)$ 또는 $f(a) > k > f(b)$ 이면

$$f(c) = k$$

를 만족하는 점 c가 a와 b 사이에 적어도 하나 존재한다.

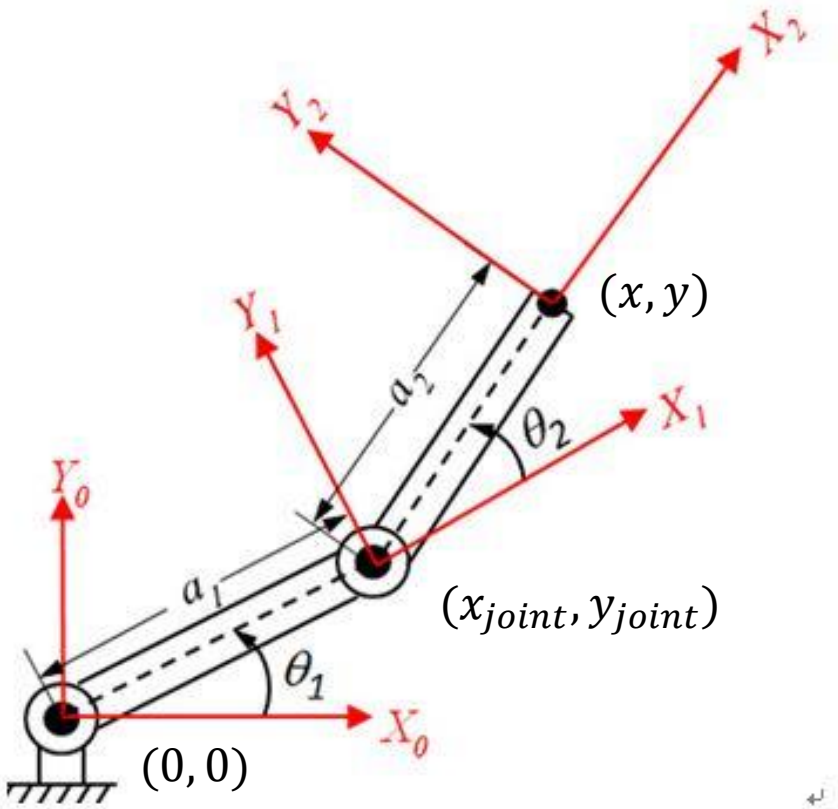
0 2

입력 예시: a, b 입력

확실합니다.

출력 예시

4-2



링크가 2개인 로봇의 말단부 위치 (x, y)에 대해서 각 링크의 회전각은 아래와 같은 식을 만족합니다.

$$\theta_1 = \operatorname{atan2}(y, x) \pm \arccos \frac{x^2+y^2+a_1^2-a_2^2}{2a_1\sqrt{x^2+y^2}}$$

$$\theta_2 = \mp \arccos \frac{x^2 + y^2 - a_1^2 - a_2^2}{2a_1 a_2}$$

(복부호 동순)

이 때, Joint의 위치를 구하기 바랍니다.
단, 평면상에서 링크가 2개인 로봇의 Joint의 위치는 총 2개가 나옵니다.

4-2(이어서)

$$\text{atan2}(y,x) = \begin{cases} \arctan\left(\frac{y}{x}\right) & (x > 0) \\ \arctan\left(\frac{y}{x}\right) + \pi & (x < 0 \; and \; y \geq 0) \\ \arctan\left(\frac{y}{x}\right) - \pi & (x < 0 \; and \; y < 0) \\ +\frac{\pi}{2} & (x = 0 \; and \; y > 0) \\ -\frac{\pi}{2} & (x = 0 \; and \; y < 0) \\ undefined & (x = 0 \; and \; y = 0) \end{cases}$$

입력 예시: a_1, a_2, x, y 입력	출력 예시
4 6 3 5	[3.889499332542204, -0.9336995995253227] [-0.889499332542204, 5.9336995995253226]

4-3

어떤 양의 정수 X의 각 자리가 등차수열을 이룬다면, 그 수를 한수라고 합니다.

예를 들어 5, 12, 99, 123 은 한수입니다. 하지만 124, 121 은 한수가 아닙니다.

N이 주어졌을 때, 1보다 크거나 같고, N보다 작거나 같은 한수의 개수를 출력하는 프로그램을 작성하시기 바랍니다.

입력 예시
110
1
210

출력 예시
99
1
105