Scientific Computing with Python Lab

5th Session(Feb 20th)

Kaheon Kim : https://github.com/kaheonkim/Scientific-computing-with-python-lab-material

Today

we are going to talk about

- #Some_explanation_about_print_function
- #Advanced_topics_about_function
- #Recursion
- #Q&A

Print function

```
print(f"{value:<n}", end=" ")
```

n : the interval between numbers in the frame

> or < : arrangement direction(right, left)

spacing : print()

Function

Outputs

```
def func_name(input1, input2, …, inputn)
    .
    .
    operation(input1, input2, …, inputn)
    .
    .
    return output1, output2, output3


o1,o2,o3 = func_name(i1, i2, i3)
```

Exercise

$y = Ax$

Recursion

usage

when definition of a concept or process depends on a simpler or previous version of itself

```
def recurse():
    ...
    recurse()
    ...

recurse()
```

recursive call

You should think about the stopping criteria to prevent the infinite loop!

Recursion

Example

Factorial function

Base rule : $1! = 1$

Recursion : $n! = n(n-1)! = n(n-1)(n-2)! = \cdots = n(n-1)(n-2)\cdots 2 \cdot 1$

Code

```python
def factorial(x):
    if x == 1:
        return 1
    else:
        return (x * factorial(x-1))


num = 3
print("The factorial of", num, "is", factorial(num))
```

$$T(n) = \begin{cases} 2 & \text{if } n = 1 \\ 3 + T(\lceil \frac{n}{2} \rceil) & \text{if } n > 1 \end{cases}$$

$$T(n) \simeq f(n) = 3 \log_2 n + 2$$

Compare it with two functions value with n = 1000

Exercise

Exercise

Sum of Digits

Related to Homework3

Problem 5

Iteration + Recursion

$$p(n) = \begin{cases} 0 & \text{if } n < 0 \\ 1 & \text{if } n = 0 \text{ or } n = 1 \\ \sum_{k \neq 0} (-1)^{k-1} p(n - g_k) & \text{if } n \geq 2 \end{cases}$$