3. More Recursions

Photo by MIGUEL GASCOJ on Unsplash

Note

Outline

Note

Roadmap

1. This topic: Recursion
2. Course: Python 1
3. Subject: Programming
4. Field
   a. `Software Engineering (SE) <>`__
   b. `Computer Science and Information Engineering (CSIE) <>`__
   c. `Electrical/Electronics Engineering (EE) <>`__

Note

Version

1. v20230728: Split into 3 sections.
2. v20220501: Refer to requirements.txt
3. v20210722: Build catalog
4. v20201118: Merge documents of /Code/Python/Python-Bsic/Recursion to /DS/Stack/Recursion
5. v20200808; v20200805; v20191125; v20180603; v20170727
6. v20121127

See also

1. Because all contents are merged to [Data Structures] book in 20201119, please refer to [Data Structures: Stack/Recursion].
2. Because review contents for information technology teachers every summer from 201607 to present, please refer to the following part [PatchUp: Recursion].
3. Because we publish a book for senior high textbook, 普通高中資訊科技Pyhon版教課書, 勁園, 20220901, please also refer to the chapter at the [4.7.2. 遞迴 (Recursion)].

3.1. Indirect Recursion

1. Syntax

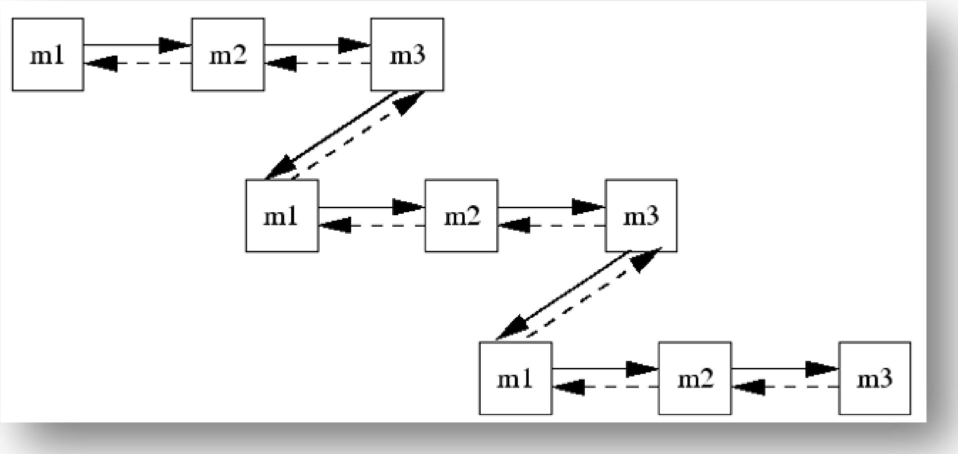
Indirect Recursion

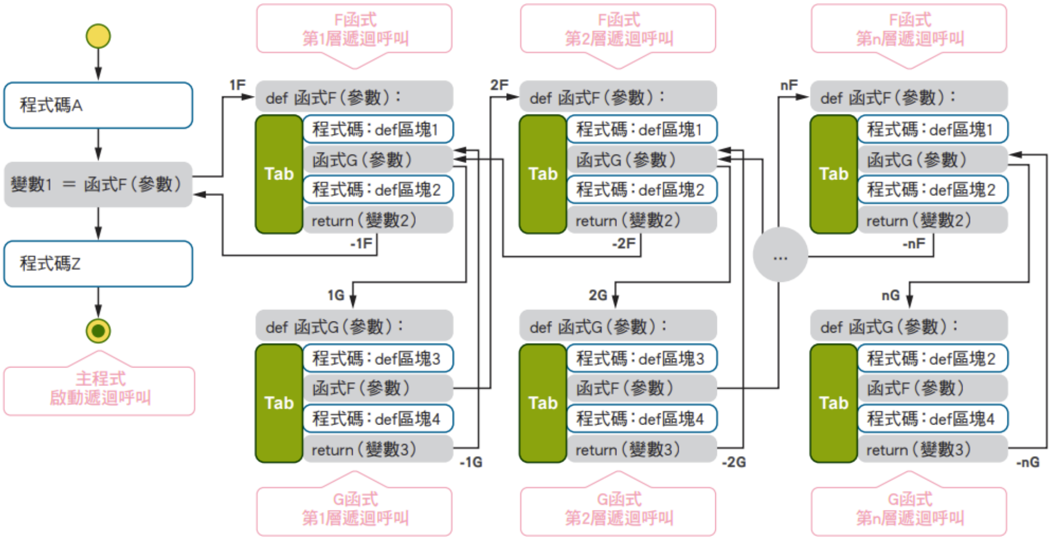
```
def  function_name1 (argument1):
    statements
    function_name2(argument2)

def function_name2 (argument2):
    statements
    function_name3(arugment3)

def function_name3 (argument3):
    statements
    function_name1(arugment1)
```

2. Process





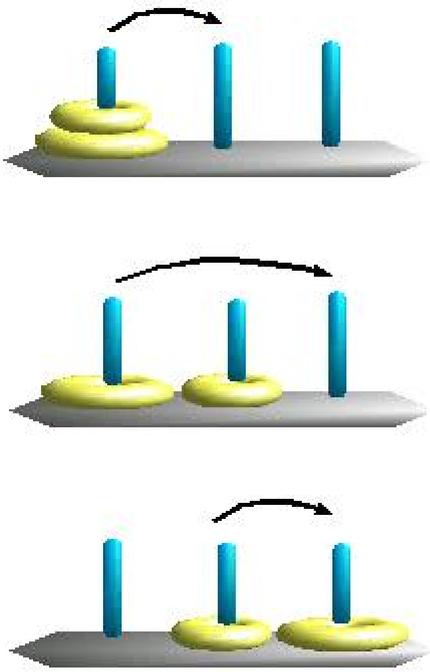
3.2. Hanoi Towers

3.2.1. Description

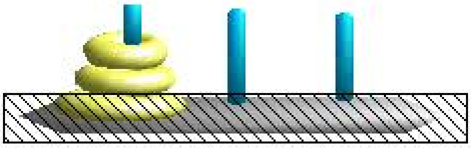
1. 河內塔(Hanoi Towers)是法國人M. Claus於1883年從泰國帶至法國的，河內為越戰時北越的首都，即現在的胡志明市； 1883年法國數學家 Edouard Lucas曾提及這個故事，據說創世紀時Benares有一座波羅教塔，是由三支鑽石棒（Pag）所支撐， 開始時神在第一根棒上放置64個由上至下依 由小 至大排列的金盤（Disc），並命令僧侶將所有的金盤從第一根石棒移至第三根石棒， 且搬運過程中遵守大盤子在小盤子之下的原則，若每日僅搬一個盤子，則當 盤子全數搬運完畢之時，此塔將毀損，而也就是世界末日來臨之時。

3.2.2. Solution

1. 如果柱子標為ABC，要由A搬至C，在只有一個盤子時，就將它直接搬至C，當有兩個盤子，就將B當作輔助柱。



2. 如果盤數超過2個，將第三個以下的盤子遮起來，就很簡單了，每次處理兩個盤子， 也就是：A->B、A ->C、B->C這三個步驟，而被遮住的部份，其實就是進入程式的遞迴處理。



3. 事實上，若有n個盤子，則移動完畢所需之次數為 $2^n - 1$，所以當盤數為64時，則所需次數為

> $$2^{64} - 1 = 18446744073709551615$$

4. That is, 5.05390248594782e+16年，也就是約5000世紀， 如果對這數字沒什麼概念，就假設每秒鐘搬一個盤子好了，也要約 5850億年左右。

5. Algorithm

> **hanoi-algorithm.py**
>
> ```python
> def Hanoi(n, A, B, C):
>     if (n == 1):
>         print("Move plate %d from %d to %d." %(n, A, C))
>     else:
>         Hanoi(n-1, A, C, B)
>         Hanoi(1, A, B, C)
>         Hanoi(n-1, B, A, C)
> ```

6. Codes

> **Code: Python** Code: C++ Code: C Code: Java Code: JavaScript Code: Ruby
>
> **hanoi.py**
>
> ```python
> def hanoi(iN, cA, cB, cC):
>     if (iN == 1):
>         return([(cA, cC)])
>     else:
>         return(hanoi(iN - 1, cA, cC, cB) +
>                hanoi(1, cA, cB, cC) +
>                hanoi(iN - 1, cB, cA, cC))
> 
> iN = eval(input("請輸入整數："))
> for move in hanoi(iN, 'A', 'B', 'C'):
>     print("盤由 %c 移至 %c" % move)
> ```

3.3. Depth First Search

3.4. Exercise

3.4.1. Ex1

1. Question+Code

> **Question** Code: Python

> 請利用遞迴函式(函式重複呼叫本身)的方法輸出一個9x9乘法表，其中不能使用任何迴圈。

See also

1. R. Sedgewick, K. Wayne, “Computer Science: An Interdisciplinary Approach”, Princeton U, 2018. [2.3 Recursion]
2. https://slidesplayer.com/slide/11680061/
3. https://openhome.cc/Gossip/AlgorithmGossip/

Note

- Start: 20170719
- System Environment