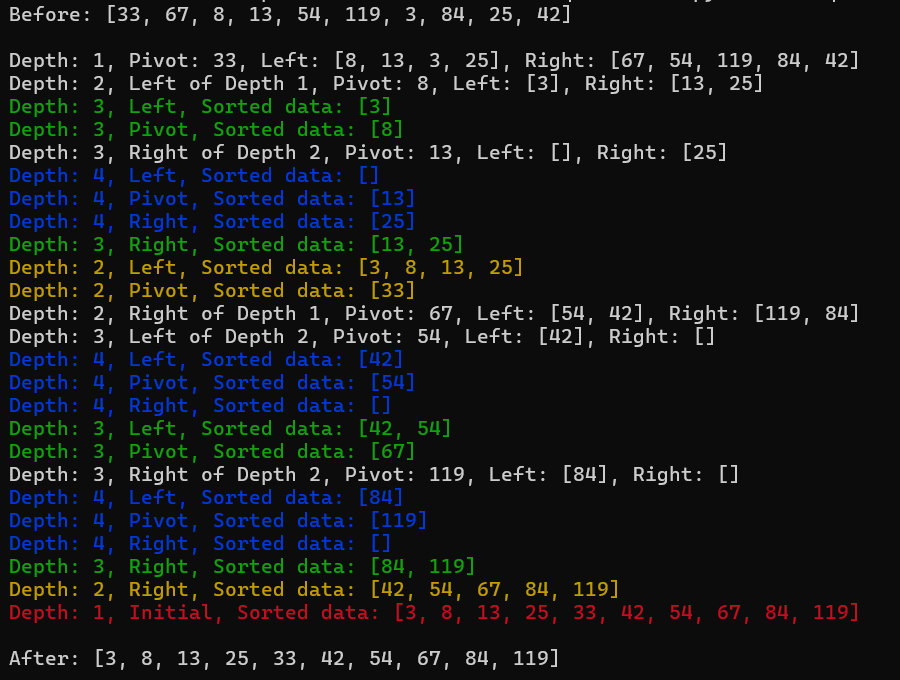
**一、執行結果:**

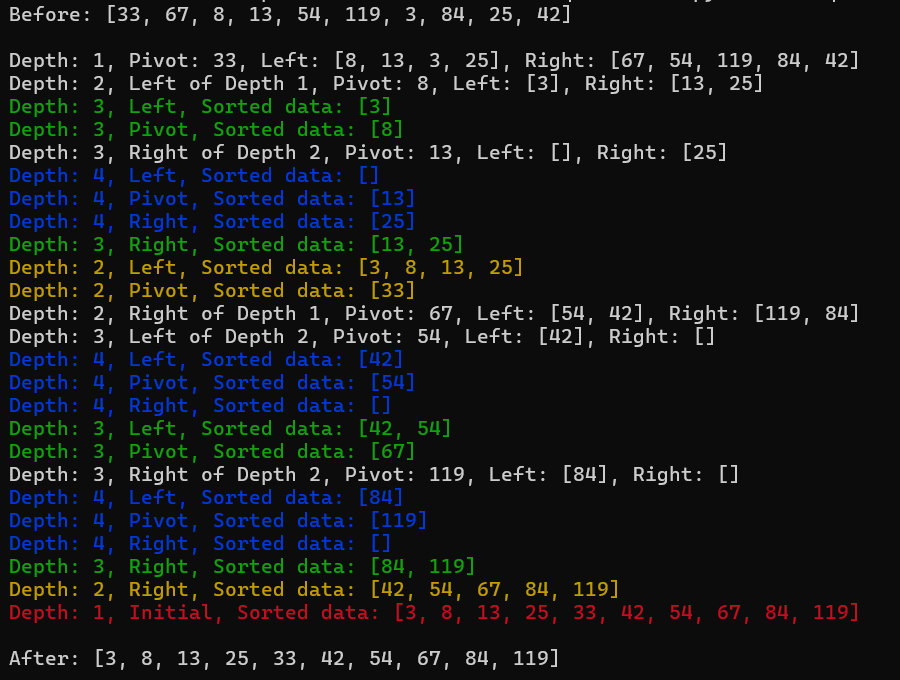
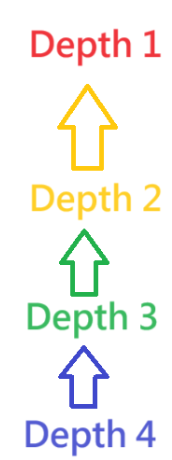
給定指定數列 [33,67,8,13,54,119,3,84,25,42]

輸出排序數列 [3,8,13,25,33,42,54,67,84,119]



**二、運作概念:**

根據顏色代表遞迴的深度，最深為藍色，其次為綠色，再來為黃色，最後為紅色。並且藉由一層一層的遞迴來計算並決定每一層深度的樞紐、左邊(小於樞紐的值)、右邊(大於樞紐的值)分別為何，最終將其整合成為新的排序數列。



Final output

Depth2 of Depth1

Depth4 of Depth3

Depth3 of Depth2

**三、程式碼**

from termcolor import colored

color\_set = ["red", "yellow", "green", "blue"]

def quicksort(data\_set,depth=1,direct=None):

color = color\_set[depth-1]

if len(data\_set) <= 1:

result = data\_set

print(colored(f"Depth: {depth}, {direct if direct else 'Initial'}, Sorted data: {result}" ,color))

return data\_set

pivot = data\_set[0]

left = []

middle = []

right = []

for x in data\_set:

if x < pivot:

left.append(x)

elif x == pivot:

middle.append(x)

else:

right.append(x)

if direct == None:

print(f"Depth: {depth}, Pivot: {pivot}, Left: {left}, Right: {right}")

else:

print(f"Depth: {depth}, {direct} of Depth {depth-1}, Pivot: {pivot}, Left: {left}, Right: {right}")

result = quicksort(left,depth+1,"Left") + quicksort(middle,depth+1,"Pivot") + quicksort(right,depth+1,"Right")

print(colored(f"Depth: {depth}, {direct if direct else 'Initial'}, Sorted data: {result}" ,color))

return result

data\_set = [33, 67, 8, 13, 54, 119, 3, 84, 25, 42]

print(f"Before: {data\_set}\n")

sorted\_data\_set = quicksort(data\_set)

print(f"\nAfter: {sorted\_data\_set}")

**四、github專案網址**

https://github.com/Yen-Ling-127/hw3\_quick\_sort.git