

第3次隨堂-隨堂-QZ3

學號：112111209

姓名：張香裕

作業撰寫時間：120 (mins, 包含程式撰寫時間)

最後撰寫文件日期：2024/11/25

本份文件包含以下主題：(至少需下面兩項，若是有多者可以自行新增)

- ☒ 說明內容
- ☒ 個人認為完成作業須具備觀念

說明程式與內容

開始寫說明，該說明需說明想法，並於之後再對上述想法的每一部分將程式進一步進行展現，若需引用程式區則使用下面方法，若為.cs檔內程式除了於敘述中需註明檔案名稱外，還需使用語法```語言種類 程式碼```，其中語言種類若是要用python則使用py，java則使用java，C/C++則使用cpp，下段程式碼為語言種類選擇csharp使用後結果：

```
public void mt_getResult(){  
    ...  
}
```

若要於內文中標示部分網頁檔，則使用以下標籤```html 程式碼```，下段程式碼則為使用後結果：

```
<%@ Page Language="C#" AutoEventWireup="true" ...>  
  
<!DOCTYPE html>  
  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head runat="server">  
<meta http-equiv="Content-Type" ...>  
    <title></title>  
</head>  
<body>  
    <form id="form1" runat="server">  
        <div>  
            </div>  
    </form>  
</body>  
</html>
```

更多markdown方法可參閱<https://ithelp.ithome.com.tw/articles/10203758>

請在撰寫"說明程式與內容"該塊內容，請把原該塊內上述敘述刪除，該塊上述內容只是用來指引該怎麼撰寫內容。

1. 請參閱投影片Topic5的第31至35頁，請用物件導向方式進行新增與刪除。(請參照題目pdf)

Ans:

```
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None

class LinkedStack:
    def __init__(self):
        self.top = None

    # 新增節點 (Push)
    def push(self, data):
        new_node = Node(data)
        new_node.next = self.top
        self.top = new_node
        print(f"新增節點: {data}")

    # 刪除節點 (Pop)
    def pop(self):
        if self.top is None:
            print("堆疊裡為空, 無法POP.")
            return None
        popped_data = self.top.data
        self.top = self.top.next
        print(f"拋出的節點: {popped_data}")
        return popped_data

    # 查看堆疊 (Display)
    def display(self):
        current = self.top
        print("堆疊裡從頂至底部為:")
        while current:
            print(current.data, end=" -> ")
            current = current.next
        print("None")

if __name__ == "__main__":
    stack = LinkedStack()

    # 嘗試從空堆疊中刪除節點
    print("嘗試從空堆疊中刪除節點:")
    stack.pop()
    print()
    # 分行好進行分類

    # 新增節點
    stack.push(10)
    stack.push(20)
    stack.push(30)
    stack.display()
```

```
# 移除節點
stack.pop()
stack.display()

# 清空堆疊後再次測試
stack.pop()
stack.pop()
stack.pop()
```

個人認為完成作業須具備觀念

開始寫說明，需要說明本次練習需學會那些觀念 (需寫成文章，需最少50字，並且文內不得有你、我、他三種文字)且必須提供完整與練習相關過程的notion筆記連結

要對物件導向方法有一定了解，才能使用其解決程式設計問題。
並且鏈結串列的操作過程可以透過繪圖，來讓思路更容易理解。

⚠ 所需的觀念為

> 堆疊(Stack):

- 後進先出(LIFO)
- 基本操作：
 - ◆ Push: 新增元素至堆疊頂端
 - ◆ Pop: 刪除頂端的元素

> 鏈結串列(Linked List):

- 節點(Node)結構: 包含資料(data)和指標(next)
- ◆ 新增節點到堆疊頂端的方法為: 將新節點的指標指向現有的"top"節點並更新"top"。
- ◆ 刪除節點的方法: 將"top"更新為下個節點，釋放原頂端(top)節點

> 物件導向程式設計(OOP):

- 節點(Node)類別: 定義節點結構。
- 堆疊(LinkedStack)類別: 實行堆疊功能。