

## 第2次隨堂-隨堂-QZ2

學號：112111220

姓名：葉崇聖

作業撰寫時間：180 (mins · 包含程式撰寫時間)

最後撰寫文件日期：2024/12/28

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Topic2的第6頁，目前已完成Push和Pop，請完成剩下功能部份—TopItem、IsEmpty與IsFull

Ans:

```
class ArrayStack:
    def __init__(self, capacity):
        self.capacity = capacity # 陣列的最大容量
        self.stack = [None] * capacity # 初始化固定大小的陣列
        self.top = -1 # 指向堆疊的頂部索引，-1 表示空

    def IsEmpty(self):
        """檢查堆疊是否為空"""
        return self.top == -1

    def IsFull(self):
        """檢查堆疊是否已滿"""
        return self.top == self.capacity - 1

    def Push(self, item):
        """將元素推入堆疊"""
        if self.IsFull():
            raise OverflowError("堆疊已滿")
        self.top += 1
        self.stack[self.top] = item

    def Pop(self):
        """從堆疊中移除頂部元素"""
        if self.IsEmpty():
            raise IndexError("堆疊是空的")
        item = self.stack[self.top]
        self.stack[self.top] = None # 清除位置
        self.top -= 1
        return item

    def TopItem(self):
        """取得堆疊頂部的元素"""
        if self.IsEmpty():
            raise IndexError("堆疊是空的")
        return self.stack[self.top]

# 測試範例
if __name__ == "__main__":
    stack = ArrayStack(5) # 建立一個容量為 5 的堆疊

    print("堆疊是空的嗎?", stack.IsEmpty())

    # 推入元素
    stack.Push(10)
    stack.Push(20)
    stack.Push(30)

    print("堆疊頂部的元素:", stack.TopItem())
    print("堆疊已滿嗎?", stack.IsFull())
```

```
# 移除元素
print("移除的元素:", stack.Pop())
print("移除後堆疊頂部的元素:", stack.TopItem())

# 繼續推入直到滿
stack.Push(40)
stack.Push(50)
stack.Push(60)

print("推入更多元素後堆疊已滿嗎?", stack.IsFull())
```

## 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