

# Homework 12

## Objective :

熟悉使用 pointer、structure 的用法

## Exercise(100%):

12

Stack 是一種常見的資料結構, 其資料的進出型態是越先進去的越後出 (FILO)。Stack 主要是由兩個函式操作, 一個是 push, 另一個是 pop。請設計一個

程式, **使用指標**實作 Stack, 讓使用者可以操作 Stack, 並輸出目前 Stack 中的元素。

使用者可以有以下兩種輸入方式:

1. **數字 1+空格+數字 N**: 此種輸入方式為 push, 當偵測到使用者前面輸入為 1 時, 將後面的數字 N 做為要 push 入 Stack 的數字, 將其放入 Stack 中, 並刷新 Stack

的狀態。Ex: (1 23)、(1 5)、(1 45)...

2. **數字 2**: 此種輸入方法為 pop, 當偵測到使用者前面輸入為 2 時, 將 Stack 中最上面的元素拿出來, 並刷新 Stack 的狀態。Ex: (2)。

必須讓使用者可以**無限輸入, 直到 EOF 後才停止**。程式中必須包含兩個函式:

1. **void mypush(int)**: 傳入一個整數 N, 該數字要被 push 到 Stack 中。

2. **int mypop()**: 將 Stack 中最上面的元素拿掉並回傳裡面的數值, 該函式必須要做判斷, 當 Stack 中已經沒有元素可以拿出時, 必須要回傳錯誤訊息, 可以利用回傳 -1 來完成。

Note: 該次作業不可使用內建 push、pop。

Note: 測資不會輸入錯誤, 可以不用防呆。

## Output:

```
Nothing in stack.
Please input the instruction:2
Error pop.
Nothing in stack.
Please input the instruction:1 3
push 3 into stack.
*      *
*   3   *
*      *
*****
Please input the instruction:1 5
push 5 into stack.
*      *
*   5   *
*      *
*****
*      *
*   3   *
*      *
*****
Please input the instruction:1 20
push 20 into stack.
*      *
*  20   *
*      *
*****
*      *
*   5   *
*      *
*****
*      *
*   3   *
*      *
*****
Please input the instruction:1 444
push 444 into stack.
*      *
```

```
Please input the instruction:1 444
push 444 into stack.
*      *
*  444   *
*      *
*****
*      *
*   20   *
*      *
*****
*      *
*   5   *
*      *
*****
*      *
*   3   *
*      *
*****
Please input the instruction:1 9
push 9 into stack.
*      *
*   9   *
*      *
*****
*      *
*  444   *
*      *
*****
*      *
*   20   *
*      *
*****
*      *
*   5   *
*      *
*****
*      *
*   3   *
*      *
*****
Please input the instruction:2
pop 9 from stack.
*      *
*  444   *
*      *
```

Please input the instruction:2  
pop 9 from stack.

```
*      *  
*  444  *  
*      *  
*****  
*      *  
*   20  *  
*      *  
*****  
*      *  
*    5  *  
*      *  
*****  
*      *  
*    3  *  
*      *  
*****
```

Please input the instruction:2  
pop 444 from stack.

```
*      *  
*   20  *  
*      *  
*****  
*      *  
*    5  *  
*      *  
*****  
*      *  
*    3  *  
*      *  
*****
```

Please input the instruction:2  
pop 20 from stack.

```
*      *  
*    5  *  
*      *  
*****  
*      *  
*    3  *  
*      *  
*****
```

Please input the instruction:^Z

-----  
Process exited after 30.06 seconds with return value 0  
請按任意鍵繼續 . . .

## Bonus(50%):

Stem-and-leaf display 是一種常用的統計圖表。請進行班級成績的排序，並進行圖表的繪製。

1. 需要先隨機產生兩個班級的成绩(Class A 和 Class B)，每班 50 人，成績 0 到 100 分，並且生成存入檔案 output1.txt, 格式如範例輸出
2. 讀入 output1.txt 對兩班成績進行排序，並再寫入 output1.txt，格式如範例輸出。
3. 對排序好的成績繪製 Stem-and-leaf display，並計算兩班中位數與眾數，把繪製好的圖形寫入新生成的 output2.txt。

## Output:

```
output1.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
Class A original
38 91 75 39 8 14 92 52 33 29
47 45 29 10 66 41 15 43 19 43
6 33 83 92 0 43 79 61 22 95
2 58 41 74 81 52 36 39 25 37
7 95 57 7 66 72 38 20 0 33

Class B original
97 29 28 92 52 70 24 49 93 83
57 60 93 31 76 63 55 8 12 45
7 48 49 24 83 95 59 40 73 70
53 30 23 65 83 6 86 2 84 65
71 20 32 84 29 86 79 86 89 49

Class A sort
0 0 2 6 7 7 8 10 14 15
19 20 22 25 29 29 33 33 33 36
37 38 38 39 39 41 41 43 43 43
45 47 52 52 57 58 61 66 66 72
74 75 79 81 83 91 92 92 95 95

Class B sort
6 7 8 12 20 23 24 24 28 29
29 30 31 32 40 45 48 49 49 49
52 53 55 57 59 60 63 64 65 65
70 70 71 73 76 79 83 83 83 84
84 86 86 86 89 92 93 93 95 97

*output2.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
Stem-and-leaf display Class A & Class B

          Class A      ClassB
                10
          12255 9 75332
                13 8 966644333
          2459 7 963100
                166 6 55430
          2278 5 97532
          1133357 4 999850
          333678899 3 210
          02599 2 9984430
          0459 1 2
          0026778 0 876

median A : 40
mode A : 33 43

median B : 59.5
mode B : 49 83 86
```