

# Data Structure

---

Programming assignment 1

# Problem1

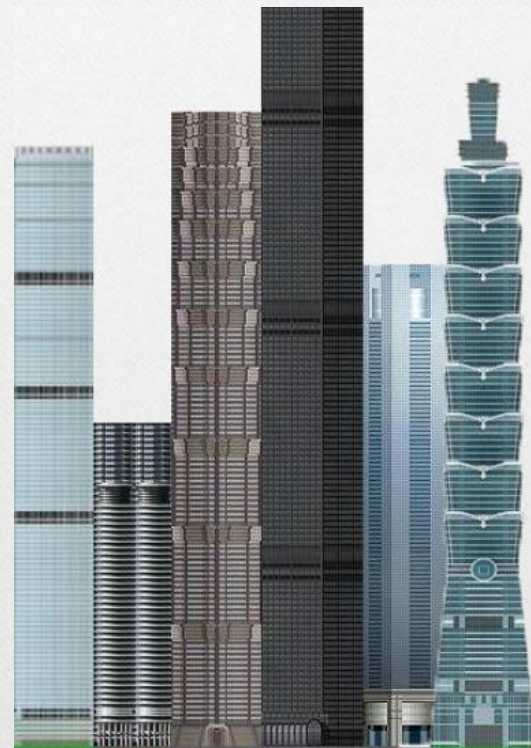
---

- According to the teacher's class, please implement the following functions of the stack.
- **push()**: Put the data in the top of the stack.
- **pop()**: Delete the data in the top of the stack.
- **top()**: View the data in the top of the stack.



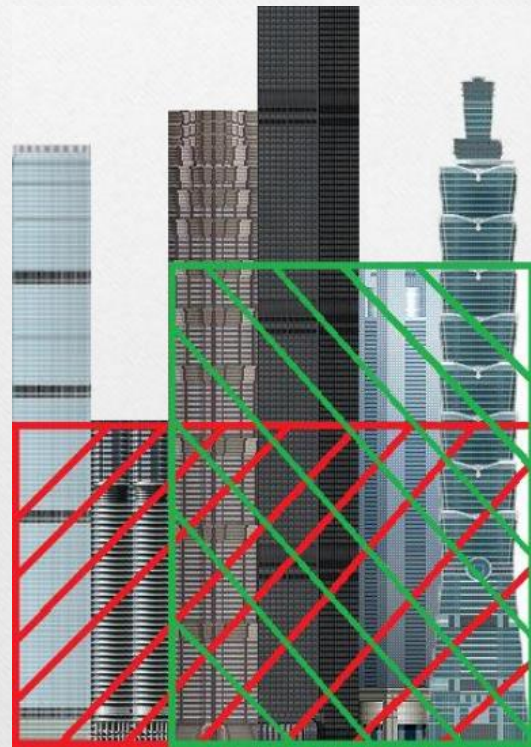
# Problem2

- 今天校方想要做一張非常醒目的廣告，因此，需要一個矩形範圍最大的牆面，考慮到不同建築的高度不同，我們想做的事情就是在不超出建築物的情況下來達到這個條件才行。



## Problem2

- 今天給定建築物高度為 $[5, 4, 6, 7, 6, 6]$ ，則我們可以發現到最大的牆面為24，受限於高度為4的建築，我們只能提供 $4 \times 6$ 的範圍。但還有另一種方案，那就是使用後面4棟建築物也可以達成，變成 $6 \times 4$ 的範圍。今天我們被要求必須找出24這個答案才行。





# Assumptions

---

- 假設每個建築物每層樓高度都相同。
- 假設每個建築物寬度也都相同。
- 只需考慮最後輸出的最大牆面即可。
- 使用您寫的`stack`來解決此問題即可，我們可以將建築物高度遞增時放入`stack`，遞減時來移出堆疊。

# Requirement

---

- Please use **python3** to write the program.
- Please fill in the code of Problem1 **corresponding to the function** in hw1.py.
- Please fill in the code of Problem2 in the **function named solution** in hw1.py.
- Execute your code to know your **score** and the **wrong place** of the code.



# Input & Output

---

- The input to the function is a list representing the **height of each building**, and the output can only number representing **the size of area**.

# Score

---

- Code 100%, 10% of each data.
- We will use the same data in the sample program to check the correctness of program.
- 請將檔案壓縮為 **zip** 檔再上傳，zip 檔名不限，然而程式碼請以 **programming\_hw1.py** 做繳交。