



## Task Lab05 ข้อ 3 (Lab05\_3)

### แบบฝึกปฏิบัติการครั้งที่ 5

#### Encapsulation and Constructor

#### จุดประสงค์

เมื่อผ่านปฏิบัติการนี้แล้ว นักเรียนจะสามารถ

1. เข้าใจคลาส (Class) และอ็อบเจกต์ (Object)
2. เข้าใจการเข้าถึงข้อมูลและเมทอดสมาชิกของคลาส
3. เข้าใจการใช้งานของ Get Method, Set Method, Constructor Method , Instance Method และ Class Method
4. เข้าใจการใช้งานเมทอดสมาชิกของคลาสประเภท Private

#### การส่งงาน

เข้าสู่เว็บ [grader.cs.science.cmu.ac.th](http://grader.cs.science.cmu.ac.th) และ login ด้วย user และ password ที่แจกให้ทาง email

- เลือกเมนู Course > 65-204114 > เลือกข้อหรือ Task ที่ต้องการส่งงาน
- Upload ไฟล์ .java ที่มีชื่อเดียวกันกับชื่อ Task เช่น Lab05\_3.java
- ให้เขียน comment เป็นรหัสนักศึกษาและชื่อไว้ด้านบนไฟล์

#### คำสั่ง

จงเขียนโปรแกรมเพื่อสร้างคลาส Stack โดยมีรายละเอียดดังนี้

5	10	7		
---	----	---	--	--



top

← ตัวอย่างกรณีมีข้อมูลในสแตก 3 ตัว ค่า top จะเท่ากับ 2

- เป็นคลาสสำหรับสร้างและจัดการกับ stack
- มี member data คือ
  - maxSize เป็นเลขจำนวนเต็ม เก็บค่าจำนวนสมาชิกที่สแตกสามารถเก็บได้
  - item[] เก็บข้อมูลสมาชิกของ stack ซึ่งจะเป็นเลขจำนวนเต็มไม่เกิน maxSize จำนวน
  - top เก็บข้อมูล index ของสมาชิกตัวสุดท้าย
- มี member method คือ
  - Stack(maxSize) เป็น constructor method ที่จะกำหนดให้ค่า top เป็น -1 โดยเป็นค่าในตอนเริ่มต้น เพื่อแสดงว่า stack ว่าง และกำหนดค่า maxSize เพื่อระบุจำนวนสมาชิกที่สแตกสามารถเก็บได้
  - void push(int x)
    - เป็นการเพิ่มข้อมูล x ลงใน stack ในตำแหน่ง top+1 และเพิ่มค่า top
    - ทั้งนี้การ push จะทำได้เมื่อ stack ไม่เต็ม (ตรวจสอบจากการเรียกใช้ `isFull()` )
    - กรณีที่ stack เต็มให้แสดงข้อความว่า "full"
  - void pop()
    - เป็นการนำข้อมูล 1 จำนวนออกจาก stack ในตำแหน่ง top และลดค่า top ลง 1
    - พิมพ์ข้อมูลที่ pop ออกมาได้ เช่น pop 7
    - ทั้งนี้การ pop จะทำได้เมื่อ stack ไม่เป็น stack ว่าง (ตรวจสอบจากการเรียกใช้ `isEmpty()` )

- กรณี stack ว่างให้แสดงข้อความว่า “empty”
  - void getTop() เพื่อ กรณีไม่มีข้อมูลให้แสดงข้อความ empty
    - เป็นการพิมพ์ข้อมูลที่อยู่ตำแหน่ง top ของ stack 1 จำนวน
    - ทั้งนี้ getTop() จะทำได้เมื่อ stack ไม่เป็น stack ว่าง (ตรวจสอบจากการเรียกใช้ isEmpty() )
    - กรณี stack ว่างให้แสดงข้อความว่า “empty”
  - int getSize() ให้คืนค่าจำนวนข้อมูลในสแตก
  - boolean isEmpty() เป็น **private method** ของคลาสสแตกสำหรับการตรวจสอบว่า stack ว่างหรือไม่ (stack จะว่างเมื่อมีสมาชิกอยู่ใน stack 0 ตัว)
 

หากไม่มีข้อมูล (สแตกว่าง) ให้คืนค่า true หากไม่ว่างให้คืนค่า false
  - boolean isFull() เป็น **private method** ของคลาสสแตก สำหรับตรวจสอบว่า stack เต็มหรือไม่ (stack จะเต็มเมื่อมีสมาชิกอยู่ใน stack เท่ากับ maxSize ตัว)
 

หากสแตกเต็มให้คืนค่า true หากไม่เต็มให้คืนค่า false
  - void show() เป็นการพิมพ์ข้อมูลทั้งหมดใน stack ทีละค่า โดยคั่นด้วยช่องว่าง 1 ช่อง โดยไม่มีการนำข้อมูลออกจาก stack แต่หาก stack ว่าง ให้แสดงข้อความว่า “empty” ดังนี้
- ```
public void show(){
    if (isEmpty()){
        for(int i=0;i<=top;i++)
            System.out.print(item[i]+ " ");
        System.out.println();
    }
    else
        System.out.println("empty");
}
```

จึงเขียนโปรแกรมเชิงวัตถุเพื่อรับเมนูคำสั่ง และทำงานตามคำสั่ง จนกว่าผู้ใช้จะเลือกคำสั่งเป็น 6 จึงจะจบโปรแกรม

### ข้อมูลเข้า

บรรทัดแรก เป็นเลขจำนวนเต็ม N แทนขนาดของสแตกที่จะสร้าง โดย  $1 \leq N \leq 10000$

บรรทัดถัดไป จะเป็นเลขจำนวนเต็ม 1-2 ตัวต่อหนึ่งบรรทัดซึ่งเป็นการเรียกใช้งาน operation ต่างๆ และเรียกจบการทำงาน โดยมี 6 เป็นข้อมูลสุดท้ายเสมอ มีหน้าที่การทำงานดังนี้

| Operation | หน้าที่                                                                                                         |
|-----------|-----------------------------------------------------------------------------------------------------------------|
| 1 x       | เรียก push(x) ของ class stack เพื่อทำการเพิ่มข้อมูล x เข้าไปในสแตก<br>กรณีสแตกเต็มให้แสดงข้อความ full           |
| 2         | เรียก pop() เพื่อลบข้อมูลที่ตำแหน่งบนสุดออกจาก Stack และพิมพ์ข้อมูลที่ลบนั้น<br>กรณีไม่มีข้อมูลให้แสดงว่า empty |
| 3         | เรียก getTop() เพื่อพิมพ์ข้อมูลที่อยู่ตำแหน่งบนสุดออกจาก Stack กรณีไม่มีข้อมูลให้แสดงข้อความ empty              |
| 4         | เรียก getSize() เพื่อพิมพ์จำนวนข้อมูลในสแตก                                                                     |
| 5         | เรียก show() เพื่อพิมพ์ข้อมูลทั้งหมดในสแตก                                                                      |
| 6         | จบการทำงานของโปรแกรม                                                                                            |

### ข้อมูลออก

แต่ละบรรทัดเป็นการแสดงผลตามการดำเนินการ (Operation) ที่ได้รับ

ตัวอย่าง Input และ Output

| ตัวอย่างที่ | Input                                                                                    | Output                                                             |
|-------------|------------------------------------------------------------------------------------------|--------------------------------------------------------------------|
| 1           | 5<br>5<br>1 1<br>1 2<br>1 3<br>5<br>1 4<br>1 5<br>5<br>6                                 | empty<br>1 2 3<br>1 2 3 4 5                                        |
| 2           | 5<br>1 1<br>1 2<br>1 3<br>5<br>1 4<br>1 5<br>1 3<br>5<br>4<br>6                          | 1 2 3<br>full<br>1 2 3 4 5<br>5                                    |
| 3           | 5<br>1 1<br>1 2<br>1 3<br>5<br>3<br>2<br>1 4<br>1 5<br>1 3<br>5<br>2<br>2<br>2<br>5<br>6 | 1 2 3<br>3<br>pop 3<br>1 2 4 5 3<br>pop 3<br>pop 5<br>pop 4<br>1 2 |
| 4           | 10<br>1 1<br>1 2<br>5<br>2<br>3<br>2<br>2<br>6<br>6                                      | 1 2<br>pop 2<br>1<br>pop 1<br>empty                                |