

# ใบงานการทดลองที่ 3

## เรื่อง อาเรย์ สตริง และฟังก์ชัน ในภาษาจาวา

### 1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจการโปรแกรมเชิงวัตถุร่วมกับอาเรย์และสตริง
- 1.2. รู้และเข้าใจการโปรแกรมเชิงวัตถุร่วมกับฟังก์ชัน

### 2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

### 3. ทฤษฎีการทดลอง

- 3.1. โครงสร้างข้อมูลแบบ “อาเรย์” มีลักษณะเป็นอย่างไร ? มีองค์ประกอบอะไรบ้าง ? อธิบายพร้อมยกตัวอย่างประกอบ

เป็น โครงสร้างข้อมูลที่มีสมาชิกเหมือนกัน (Memory) ในชุดๆ แต่จะจัดเก็บด้วยจำนวนช่องข้อมูลขนาดคงที่  
ที่แต่ละช่องจะเก็บข้อมูลชนิดเดียวกัน และอยู่ในตำแหน่งที่ต่อเนื่องกัน เริ่มนับที่ตำแหน่งที่ 0-ก  
ตัวอย่างประกอบ String[] God = { "Jong", "TheJong", "DevilJong" } ;

- 3.2. การเข้าถึงแต่ละ Element ของอาเรย์สามารถทำได้อย่างไร ? อธิบายพร้อมยกตัวอย่างประกอบ

เข้าถึง (Access) โดยใช้ในโครงสร้าง Array สามารถใช้ Index หมายเลขของช่องข้อมูลที่ต้องการ  
(Index) ไปเข้าถึงข้อมูลได้ Access หมายเลขของช่องข้อมูลที่ต้องการ  
ตัวอย่าง String[] God = { "Jong", "TheJong", "DevilJong" } ;  
System.out.println(God[0]) ;

- 3.3. คำสั่ง length เกี่ยวข้องกับอาเรย์อย่างไร ? อธิบายพร้อมยกตัวอย่างประกอบ

คำสั่ง length สามารถใช้เพื่อหาจำนวนสมาชิกใน Array ได้  
ตัวอย่าง String[] God = { "Jong", "TheJong", "DevilJong" } ;  
System.out.println(God.length) ;

- 3.4. จงยกตัวอย่างประกอบในการวนรอบเพื่อแสดงค่าภายในตัวแปรอาเรย์ตั้งแต่ค่าแรกจนถึงค่าสุดท้าย

ตัวอย่าง String[] God = { "Jong", "TheJong", "DevilJong" } ;  
for (i=0 ; i < 3 ; i++) {  
    System.out.println(i) ;  
}

- 3.5. จงยกตัวอย่างการใช้งานคำสั่ง for each เพื่อแสดงค่าภายในตัวแปรอาเรย์

```
String[] God = {"Jong", "TheJong", "DevilJong"};
for (i=0; i < 3; i++) {
    System.out.println(i);
}
```

3.6. เหตุใดจึงต้องมีคำสั่ง import java.util.Arrays ; ในส่วนต้นของไฟล์ ?

เพื่อใช้กับ class ชื่อ Arrays ซึ่งอยู่ใน Library ชื่อ "java.util"

3.7. คำสั่ง Arrays.copyOf( \_\_\_\_, \_\_\_\_, \_\_ ); มีหน้าที่ทำอะไร ?

เพื่อคัดลอกข้อมูลจาก Array ต้นฉบับ ไปสร้าง Array ตัวใหม่โดยที่ Array ตัวใหม่จะแยกออกมา  
โดยสละพื้นที่ของในตัวของ Array ต้นฉบับ

3.8. จงยกตัวอย่างการประกาศ String และกำหนดค่าคำว่า "Hello World" ในภาษาจาวา

String TheJone = "Hello World"

3.9. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง toUpperCase() ในภาษาจาวา

toUpperCase() ใช้เปลี่ยนตัวอักษรทุกตัวใน String ให้เป็นตัวพิมพ์ใหญ่ Ex. String TheJone = "Hello World";  
System.out.println (txt.toUpperCase());

3.10. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง toLowerCase() ในภาษาจาวา

toLowerCase() ใช้เปลี่ยนตัวอักษรทุกตัวใน String ให้เป็นตัวพิมพ์เล็ก String TheJone = "Hello World";  
System.out.println (txt.toLowerCase());

3.11. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง indexOf() ในภาษาจาวา

คือ เมื่อดูว่าตัวที่ระบุใน String นั้นมีอยู่ใน String หรือไม่ Ex. String myStr = "Hello planet earth, you are a great planet.";  
System.out.println (myStr.indexOf("e", 5));

3.12. จงอธิบายความแตกต่างระหว่างการเชื่อม String แบบปกติและแบบใช้คำสั่ง concat()

การเชื่อม String แบบปกติคือการนำ String มาเชื่อมกันโดยใช้ + หรือ concat() ซึ่งจะได้ String ใหม่ที่เชื่อม  
String firstName = "SonRai", lastName = "SingRun" System.out.println (firstName.concat(lastName));

3.13. หากต้องการแสดงสัญลักษณ์พิเศษภายในตัวแปร String ควรทำอย่างไร ?

คือ ใช้ \ เพื่อแทนสัญลักษณ์พิเศษภายใน String

3.14. จงอธิบายและยกตัวอย่างประกอบการสร้างฟังก์ชันในภาษาจาวา

คือ print (แสดงค่าในคอนโซล) (พิมพ์ค่าที่ส่งไปแสดง ตามด้วย ตัว \n)

```
Ex. static void myMethod() {  
    System.out.println("SongRai Is The God");  
}
```

### 3.15. อธิบายข้อแตกต่างระหว่าง Pass by value และ Pass by reference

Pass by Value เป็น argument ของฟังก์ชัน ซึ่งส่งค่าที่ ค่าในฟังก์ชันจะไม่ส่งกลับไปสู่ค่าก่อนหน้าฟังก์ชัน

Pass by Reference เป็นส่งตัวชี้ (variable) เป็น argument ของฟังก์ชัน ซึ่งส่งค่าที่ส่งค่ากลับมาในฟังก์ชันและส่งค่าที่ส่งกลับไป

### 3.16. ความแตกต่างระหว่างการประกาศฟังก์ชันแบบ void กับแบบ int, double, float, string คืออะไร ?

เป็นประเภทของ return ของฟังก์ชัน ซึ่งโดยส่วนมากไม่ได้ หรือ primitive datatype หรือ reference type ก็ได้ และหากไม่ได้ระบุค่าส่งกลับ จะใช้ void

### 3.17. โครงสร้างข้อมูลแบบ Stack แตกต่างกับ Array อย่างไร ?

Stack เป็นโครงสร้างข้อมูลเชิงเส้น (Linear) ซึ่งทำงานด้วยหลักการ Last In First Out (LIFO) ส่วน Array เป็นโครงสร้างข้อมูลเชิงเส้น (Linear) ซึ่งทำงานด้วยหลักการ First In First Out (FIFO) และ Array สามารถเข้าถึงข้อมูลได้ทุกตำแหน่ง ในขณะที่ Stack สามารถเข้าถึงข้อมูลได้เฉพาะตำแหน่งที่เพิ่งใส่เข้าไปเท่านั้น

### 3.18. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง Push ในโครงสร้างข้อมูลแบบ Stack

push() เป็นคำสั่งที่เพิ่ม Element ลงไปในส่วนบนสุดของ Stack  
Ex. public void push(String param) {  
 Node n = new Node();  
 n.data = param; n.next = begin;  
 begin = n;  
}

### 3.19. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง Pop ในโครงสร้างข้อมูลแบบ Stack

pop() เป็นคำสั่งที่ลบ Element ออกจากส่วนบนสุดของ Stack  
Ex. public String pop() {  
 Node temp = (Optional.ofNullable(begin).orElseThrow(() -> new NoSuchElementException()));  
 begin = begin.next; return temp.data;  
}

### 3.20. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง isEmpty ในโครงสร้างข้อมูลแบบ Stack

isEmpty() เป็นคำสั่งที่ตรวจสอบว่า Stack ว่างหรือไม่  
Ex. stack.add("The Song");  
System.out.println(stack.isEmpty());

### 3.21. อธิบายพร้อมยกตัวอย่างประกอบความหมายของคำว่า Stack overflow

ฟังก์ชันที่เรียกซ้ำกันมากเกินไป หรือเรียกซ้ำกันจนเกิน Stack Overflow  
หรือฟังก์ชันที่เรียกซ้ำกันมากเกินไป (หรือเรียกซ้ำกันจนเกิน) ฟังก์ชันที่เรียกซ้ำกันมากเกินไป  
อาจส่งผลให้โปรแกรมทำงานช้าลง หรืออาจเกิดข้อผิดพลาดได้

#### 4. ลำดับขั้นการปฏิบัติการ

4.1. จงแก้โจทย์ปัญหาดังต่อไปนี้

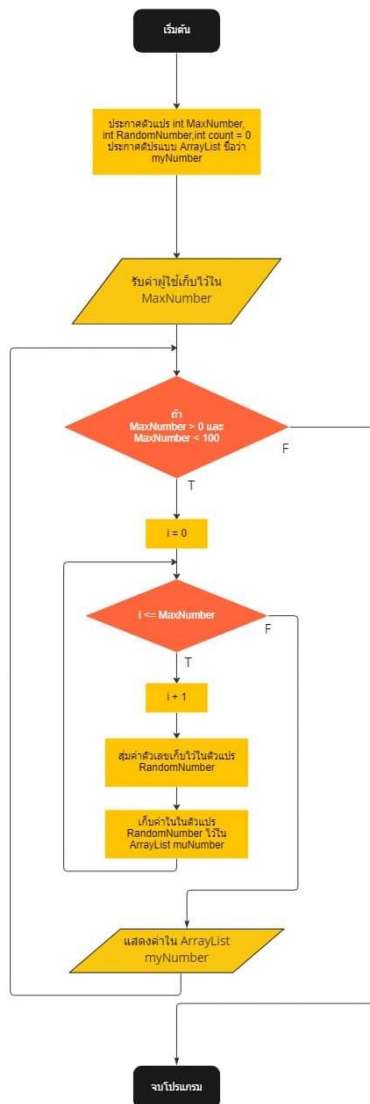
4.1.1. จงเขียนโปรแกรมเพื่อสุ่มค่าเข้าไปในอาเรย์ 1 มิติ ตามจำนวนค่าที่รับจากผู้ใช้ โดยค่าที่ถูกสุ่มจะต้องเป็นตัวเลขจำนวนเต็มที่อยู่ระหว่าง 0 ถึง 99 เท่านั้น

Test case 1	Test case 2
Please enter your random value : <b>8</b> ----- Array[0] = 94 Array[1] = 32 Array[2] = 46 Array[3] = 18 Array[4] = 27 Array[5] = 5 Array[6] = 31 Array[7] = 17	Please enter your random value : <b>12</b> ----- Array[0] = 56 Array[1] = 27 Array[2] = 13 Array[3] = 15 Array[4] = 65 Array[5] = 29 Array[6] = 11 Array[7] = 92 Array[8] = 95 Array[9] = 47 Array[10] = 58 Array[11] = 62

4.1.2. ผังงานแสดงกระบวนการทำงานและโค้ดโปรแกรม(ที่ตรงตามผังงาน)

ผังงาน	โค้ดโปรแกรม

### LAB3.1



```

1* import java.util.Scanner; // Import the Scanner class
4
5 public class Lab0301 {
6
7     public static void main(String[] args) {
8         int MaxNumber;
9         int RandomNumber;
10        int count = 0;
11
12        //-----class Import-----//
13
14        Random rand = new Random(); //instance of random class
15        ArrayList<Integer> myNumbers = new ArrayList<Integer>(); //สำหรับเก็บค่า ArrayList
16        //-----End class Import-----//
17
18        System.out.print("Please enter your random value : ");
19        Scanner myObj = new Scanner(System.in); // Create a Scanner object
20        MaxNumber = myObj.nextInt(); // รับค่าผู้ใช้ใส่มา
21
22
23        RandomNumber = rand.nextInt(MaxNumber); //สุ่มตัวเลข
24        System.out.println("-----");
25
26
27
28        if(MaxNumber > 0 && MaxNumber < 100) {
29            for(int i=0; i <= MaxNumber; i++) {
30                RandomNumber = rand.nextInt(MaxNumber); //สุ่มตัวเลขอีกจำนวน
31                myNumbers.add(RandomNumber);
32            } // End forLoop
33
34            for(int j: myNumbers ) {
35                System.out.println("Array["+count+"]"+" = " + j);
36                count++;
37            } // End forLoop
38        } else {
39            System.out.println("Please Enter Number 0-99");
40        } //End if
41    } //END Void main
42
43 }
44
  
```

```

Please enter your random value : 8
-----
Array[0] = 5
Array[1] = 1
Array[2] = 6
Array[3] = 3
Array[4] = 2
Array[5] = 5
Array[6] = 4
Array[7] = 7
Array[8] = 1
  
```

--	--

#### 4.2. จงแก้ไขข้อปัญหาดังต่อไปนี้

##### 4.2.1. จงเขียนฟังก์ชันการจัดการโครงสร้างข้อมูลแบบ Stack พร้อมจำลองการทำงานโดยการเรียกใช้คำสั่งพื้นฐานดังต่อไปนี้

คำสั่ง Push( String Value ) ; เพื่อนำข้อมูลเข้าไปเก็บไว้ใน Stack

คำสั่ง Pop( ) ; เพื่อนำข้อมูลบนสุดออกจาก Stack

คำสั่ง isEmpty( ) ; เพื่อตรวจสอบข้อมูลใน Stack ว่ามีอยู่หรือไม่

คำสั่ง Top( ) ; เพื่อตรวจสอบข้อมูลที่อยู่บนสุด

คำสั่ง CheckStack( ) ; เพื่อตรวจสอบค่าภายใน Stack ทั้งหมด

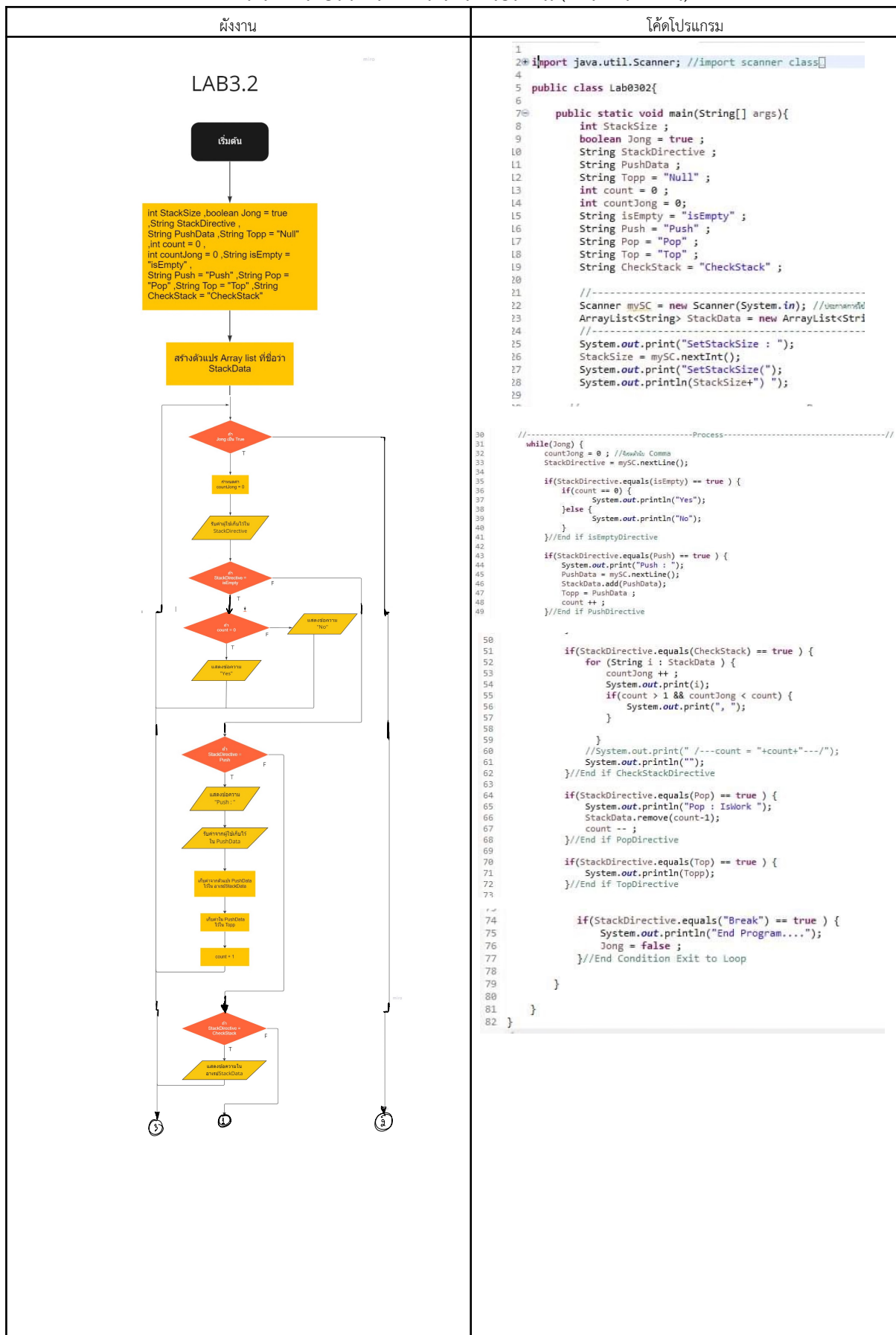
คำสั่ง SetStackSize( int size ) ; เพื่อกำหนดขนาดเริ่มต้นของ Stack

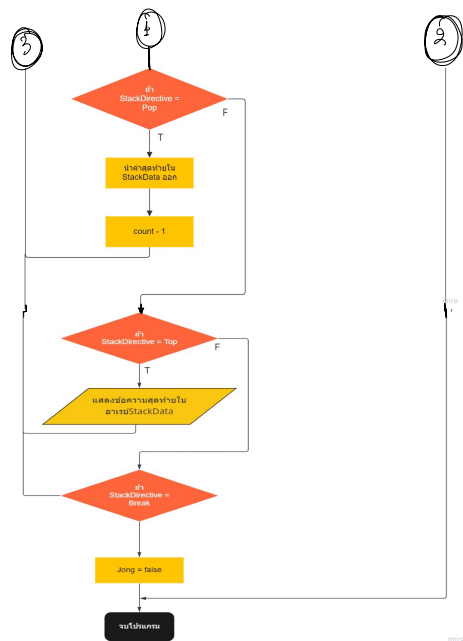
Test case
SetStackSize( 3 ) isEmpty ----  Yes Top ----  NULL Push : Hello CheckStack ----  STACK : Hello Push : Hi CheckStack ----  STACK : Hi, Hello Push : Test CheckStack ----  STACK : Test, Hi, Hello Top ----  Top = Test Pop CheckStack

```
----| STACK : Hi, Hello  
isEmpty  
----| No  
Push : OK  
CheckStack  
----| STACK : OK, Hi, Hello  
Push : RMUTL  
----| Stack Overflow  
CheckStack  
----| STACK : OK, Hi, Hello
```



#### 4.2.2. ผังงานแสดงกระบวนการทำงานและโค้ดโปรแกรม(ที่ตรงตามผังงาน)





```

SetStackSize : 77
SetStackSize(77)
isEmpty
---|Yes
Top
--|Null

```

```

Push
Push : TheJong
CheckStack
CheckStack
---|Stack : TheJong

```

```

Push : GodJong
CheckStack
---|Stack : TheJong, GodJong
Top
--|GodJong

```

```

Pop : IsWork
CheckStack
---|Stack : TheJong
isEmpty
---|No

```

```

Push : JongRaiIsRealGod
Push
Push : GodDevilJongRai
CheckStack
---|Stack : TheJong, JongRaiIsRealGod, GodDevilJongRai

```

--	--

## 5. สรุปผลการปฏิบัติการ

พอใช้ ArrayList ลงอีกที จอมง ทำให้ เกิดความซ้ำ ในที่ จึงพอ จอมง  
 สามารถ เก็บใน Array ต่อมา สามารถ จอ 100 ถึง 200 สิบ ซัด ตำแหน่ง ซัด ๆ ที่  
 อยู่ใน Array 15 สิบ ๆ 16 ซัด ซัด เสร็จ Array จอมง ๆ ๆ จอ จอมง  
 ตอนใน Array ทำให้ ซัด และ จอ ซัด ซัด

## 6. คำถามท้ายการทดลอง

6.1. ข้อควรระวังในการใช้งาน Array ในภาษาจาวาคืออะไร ?

1. ข้อมูลที่จะเก็บไว้ใน Array ต้องเป็นข้อมูลชนิดเดียวกัน

2. ต้องระวังขนาดของ Array ว่าจะเก็บข้อมูลได้

6.2. ข้อควรระวังในการใช้งาน String ในภาษาจาวาคืออะไร ?

String เป็น วัตถุ อันสืบ primitive datatype ไม่ใช่เป็นวัตถุ จะเก็บ  
 ข้อมูลไว้ใน heap memory วัตถุที่ไม่ได้ถูกใช้งานจะเก็บไว้ใน Garbage  
 Collection หรือ จอ จอ

6.3. ฟังก์ชันในภาษาจาวาไม่สามารถใช้งานแบบ Pass by reference ในภาษาซีได้ คุณมีแนวทางการ  
 แก้ไขปัญหานี้ได้อย่างไร ?

ไม่จำเป็น Pass by Reference ใดๆ แต่ทำแบบ Pass Reference Type by Value  
 หรือ mr copy Reference

6.4. โครงสร้างข้อมูลแบบ Stack แตกต่างกับโครงสร้างข้อมูลแบบ Array อย่างไร ?

Stack เป็นโครงสร้างข้อมูลเชิงเส้น ที่ใช้จัดเก็บข้อมูลแบบ后进先出 (Last In First Out)  
 Array เป็นโครงสร้างข้อมูลเชิงเส้น ที่ใช้จัดเก็บข้อมูลแบบ先进先出 (First In First Out)  
 ซึ่งข้อมูลจะเก็บใน Stack

