

ใบงานการทดลองที่ 3

เรื่อง อาร์เรย์ สตริง และฟังก์ชัน ในภาษาจาวา

1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจการโปรแกรมเชิงวัตถุร่วมกับอาร์เรย์และสตริง
- 1.2. รู้และเข้าใจการโปรแกรมเชิงวัตถุร่วมกับฟังก์ชัน

2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

3. ทฤษฎีการทดลอง

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

เป็นโครงสร้างข้อมูลที่มีการจองพื้นที่หน่วยความจำ (memory) เป็นชุดๆ แต่ละชุดประกอบด้วยจำนวนช่อง
ข้อมูลหลายช่องพื้นที่แต่ละช่องข้อมูลจะเก็บข้อมูลชนิดเดียวกันและอยู่ในตำแหน่งที่ต่อเนื่องกันไปตามลำดับ
Ex.int[] number=10, -20,30,40,60);

- 3.2. การเข้าถึงแต่ละ Element ของอาร์เรย์สามารถทำได้อย่างไร ? อธิบายพร้อมยกตัวอย่างประกอบ

Element ของอาร์เรย์ ลำดับแรกจะเป็น 0 เสมอลำดับของ element ของอาร์เรย์โดยส่วนมากจะเป็น
ค่าตัวเลขจำนวนเต็ม ex.num[0]={1,2,3}

- 3.3. คำสั่ง length เกี่ยวข้องกับอาร์เรย์อย่างไร ? อธิบายพร้อมยกตัวอย่างประกอบ
เป็นคำสั่งในการหาความยาวของอาร์เรย์

- 3.4. จงยกตัวอย่างประกอบในการวนรอบเพื่อแสดงค่าภายในตัวแปรอาร์เรย์ตั้งแต่ค่าแรกจนถึงค่าสุดท้าย

```
Int[] cars={0,1,2,3};  
for(int i=0; i < cars.length; i++) {  
    System. Out print (cars[i]+",");  
} // end for
```

- 3.5. จงยกตัวอย่างการใช้งานคำสั่ง for each เพื่อแสดงค่าภายในตัวแปรอาร์เรย์

3.5. จงยกตัวอย่างการใช้งานคำสั่ง for each เพื่อแสดงค่าภายในตัวแปรอาร์เรย์

```

Int sum=0;
int[]a= {91,75, 43,22,18,64};
Int [] a={91,75,43,22,18,64};
for(int i=0;i<a.length;i++)
{
    Sum+=a[i];
    System.out.print(i+ " ");
}
//end foreach
System.out.println("");
System.out.print("Sum="+Sum);

```

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

ช่วยในการใช้ฟังก์ชันของอาเรย์ได้ง่ายขึ้นโดยไม่ต้องพิมพ์เอง

3.7. คำสั่ง Arrays.copyOf(_____ , _____); มีหน้าที่ทำอะไร ?

เปลี่ยนขนาดของอาร์เรย์อย่างเช่นอาร์เรย์เดิมคือ 10 สามารถเปลี่ยนเป็น 20 ได้

3.8. จงยกตัวอย่างการประกาศ String และกำหนดค่าคำว่า “Hello World” ในภาษาจาวา

```
String e="Hello World";  
System.out.print(e);
```

3.9. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง toUpperCase() ในภาษาจาวา

จะเปลี่ยนบทบาทเป็นตัวพิมพ์ใหญ่

```
String e = "Hello World";  
System.out.print(e.toUpperCase());
```

3.10. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง `toLowerCase()` ในภาษาจาวา

จะเปลี่ยนบทบาทเป็นพิมพ์เล็ก

```
String e="Hello World";  
System.out.println(e.toLowerCase());
```

3.11. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง indexOf() ในภาษาจาวา

เป็นคำสั่งในการหาข้อความที่อยู่ในรูปแบบ string

```
String e = "Hello World" ;
System.out.println(e.toLowerCase());
System.out.println(e.toUpperCase());
System.out.println(e.indexOf("benjaporn"));
If(e.indexOf("Id") != -1) {
```

```
System.out.println(e.indexOf("Id"));
System.out.println("FOUND");
```

```
{ else {  
    System.out.println("Not FOUND");  
} //end if
```

3.12. จงอธิบายความแตกต่างระหว่างการเชื่อม String แบบปกติและแบบใช้คำสั่ง concat()

concat() เป็นการนำ String มารวมกัน

3.13. หากต้องการแสดงสัญลักษณ์พิเศษภายในตัวแปร String ควรทำอย่างไร ?

ต้องใช้เครื่องหมาย\ก่อนตามด้วยสัญลักษณ์พิเศษ เช่น \'\",\\

3.14. จงอธิบายและยกตัวอย่างประกอบการสร้างฟังก์ชันในภาษาจาวา

ฟังก์ชันหรือในจาวาเรียกว่าเมทอด (method) มีความสำคัญต่อการเขียนโปรแกรมมากเพราะช่วยในการนำส่วนของโปรแกรมที่มีการทำงานซ้ำๆ สามารถนำมาใช้ด้วยโค้ดสั้นๆ ช่วยลดความซ้ำซ้อนของโค้ดภายในโปรแกรมและทำให้อ่านง่ายขึ้น

```
Package hello worldproject;
```

```
Public class lab {
```

```
    public static void main (string[] args ) {
```

```
        int a;
```

```
        float b;
```

```
        String c;
```

```
        Test d =new Test();
```

```
        d.DodgerBlue();
```

```
        System.out.println("Answer="+d.Answer);
```

```
        d.Answer=99;
```

```
        System.out.println("Answer="+d.Answer);
```

```
        f();
```

```
        abc();
```

```
    }//end method
```

```
    public static int f() {
```

```
        System.out.println("Hello World");
```

```
        return 1;
```

```
    }//end function
```

```
    public static void abc() {
```

```
        System.out.println("Hello World");
```

```
    }//end function
```

```
}//end class
```

3.15. อธิบายข้อแตกต่างระหว่าง Pass by value และ Pass by reference

Pass by value เป็นการผ่านข้อมูลโดยผ่านการ return

Pass by reference มีในภาษาซีเท่านั้นคือการส่งผ่านข้อมูลผ่านตัวแปรแบบ pointer และแบบ string

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

Void มี return

Int double, float, string ไม่มี return

3.17. โครงสร้างข้อมูลแบบ Stack แตกต่างกับ Array อย่างไร ?

Array จะเหมือนกับช่องหลายๆ ช่องแล้วแต่มีตัวที่มี
มากขึ้นน้อยส่วน stack จะเหมือนกับกล่องวางต่อกัน

3.18. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง Push ในโครงสร้างข้อมูลแบบ Stack

เป็นการเพิ่มข้อมูลเข้าไปใน stack เมื่อ stack เต็มสถานะที่เกิดเป็น stack overflow

3.19. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง Pop ในโครงสร้างข้อมูลแบบ Stack

เป็นการนำเอาข้อมูลบนสุดออกก่อนทำงานแบบ FILO

3.20. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง isEmpty ในโครงสร้างข้อมูลแบบ Stack

isEmpty จะ return ค่าออกมาแค่ true, false เพื่อตรวจว่ามีข้อมูลใน stack ไหมถ้ามีจะเป็น true ถ้าไม่มีจะเป็น false

3.21. อธิบายพร้อมยกตัวอย่างประกอบความหมายของคำว่า Stack overflow

เมื่อ stack เต็มแล้วแต่ยัง push ข้อมูลต่อแต่เพราะมัน push ต่อไม่ได้แล้วเกิด stack overflow

4. ลำดับขั้นการปฏิบัติการ

4.1. จงแก้โจทย์ปัญหาดังต่อไปนี้

4.1.1. จงเขียนโปรแกรมเพื่อสุ่มค่าเข้าไปในอาเรย์ 1 มิติ ตามจำนวนค่าที่รับจากผู้ใช้ โดยค่าที่ถูกสุ่มจะต้องเป็นตัวเลขจำนวนเต็มที่อยู่ระหว่าง 0 ถึง 99 เท่านั้น

| Test case 1 | Test case 2 |
|---|---|
| <p>Please enter your random value : 8</p> <p>-----</p> <p>Array[0] = 94 Array[1] = 32 Array[2] = 46 Array[3] = 18 Array[4] = 27 Array[5] = 5 Array[6] = 31 Array[7] = 17</p> | <p>Please enter your random value : 12</p> <p>-----</p> <p>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</p> |

4.1.2. ผังงานแสดงกระบวนการทำงานและโค้ดโปรแกรม(ที่ตรงตามผังงาน)

| ผังงาน | โค้ดโปรแกรม |
|--|---|
| <pre> graph TD Start([Start]) --> Init[n,r=0,Array[]] Init --> Scan[/Scan n/] Scan --> Alloc[[]Array=new int[n]] Alloc --> Decision{i<n?} Decision -- Yes --> Random[random ค่า r เก็บในArray] Random --> Print[/print Array[]=ค่าArray/] Print --> Increment[i=i+1] Increment --> Decision Decision -- No --> End([End]) </pre> | <pre> lab3_1.java x 1 package lab3; 2 import java.util.Random; 3 import java.util.Scanner; 4 5 public class lab3_1 { 6 public static void main(String[] args) { 7 int numValues = getNumValuesFromUser(); 8 int[] values = generateRandomValues(numValues); 9 printValues(values); 10 } 11 12 public static int getNumValuesFromUser() { 13 Scanner scanner = new Scanner(System.in); 14 System.out.print("Please enter your random value: "); 15 return scanner.nextInt(); 16 } 17 18 public static int[] generateRandomValues(int numValues) { 19 Random random = new Random(); 20 int[] values = new int[numValues]; 21 for (int i = 0; i < numValues; i++) { 22 values[i] = random.nextInt(100); 23 } 24 return values; 25 } 26 27 public static void printValues(int[] values) { 28 for (int i = 0; i < values.length; i++) { 29 System.out.println("Array[" + i + "] = " + values[i]); 30 } 31 } 32 } </pre> |

| | |
|--|--|
| | |
|--|--|

| | |
|--|--|
| | |
|--|--|

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

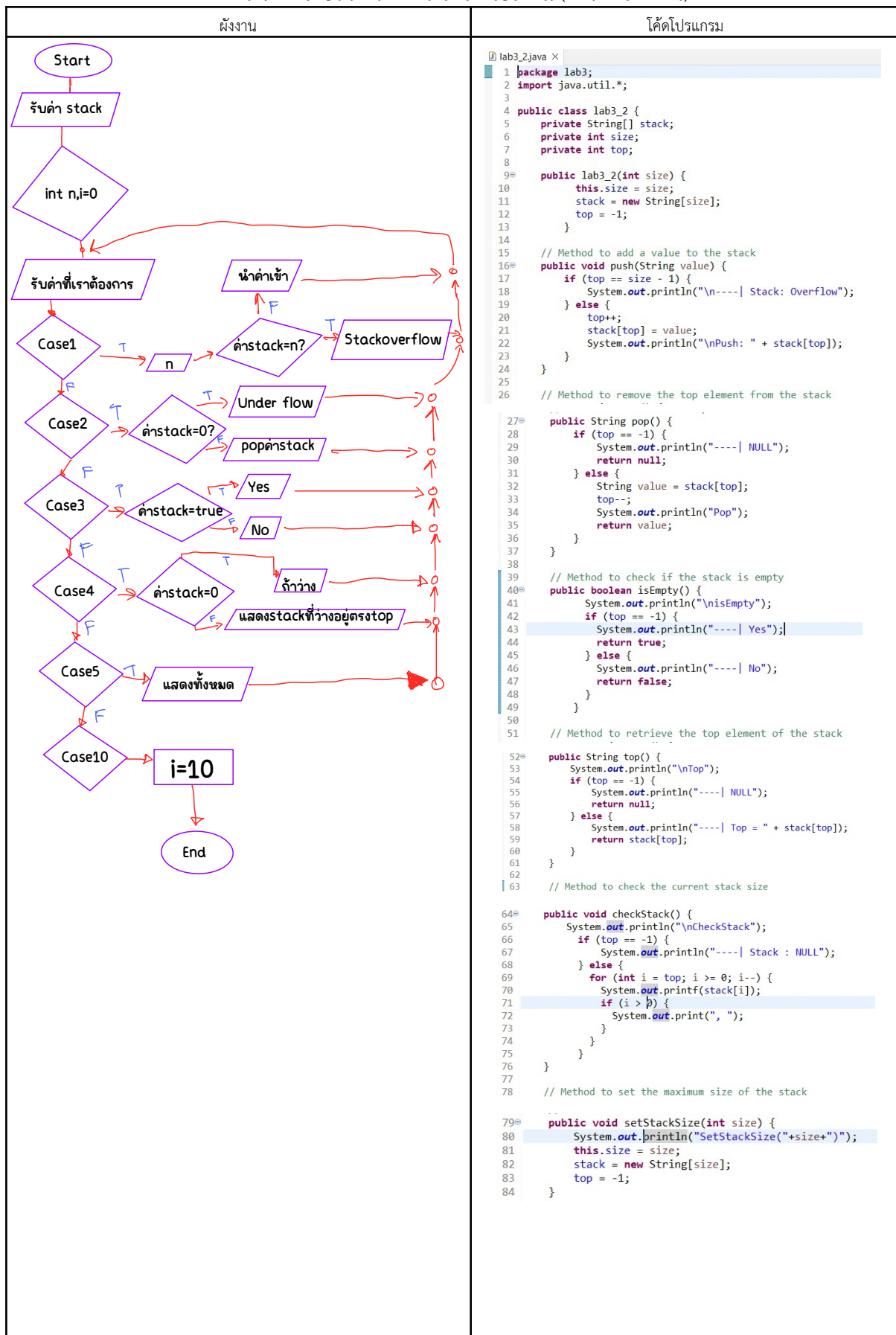
คำสั่ง CheckStack() ; เพื่อตรวจสอบค่าภายใน Stack ทั้งหมด

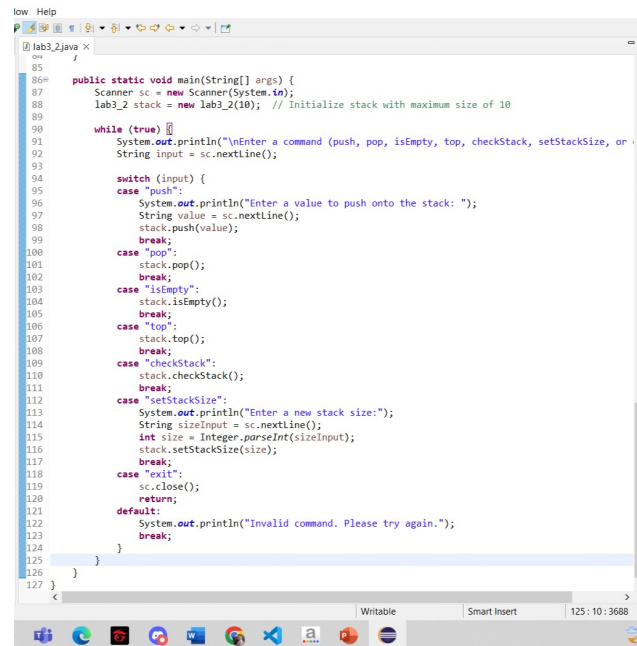
คำสั่ง SetStackSize(int size) ; เพื่อกำหนดขนาดเริ่มต้นของ Stack

| 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. ฟังก์ชันแสดงกระบวนการทำงานและโค้ดโปรแกรม(ที่ตรงตามฟังก์ชัน)





The screenshot shows a Java IDE window titled 'lab3_2.java'. The code implements a stack with a maximum size of 10. It includes methods for push, pop, isEmpty, top, checkStack, and setStackSize. The main method uses a Scanner to read user input and a switch statement to handle different commands. The IDE interface includes a toolbar at the top, a status bar at the bottom with 'Writable' and 'Smart Insert' indicators, and a taskbar at the very bottom showing various application icons.

```
low Help
lab3_2.java
85
86 public static void main(String[] args) {
87     Scanner sc = new Scanner(System.in);
88     lab3_2 stack = new lab3_2(10); // Initialize stack with maximum size of 10
89
90     while (true) {
91         System.out.println("\nEnter a command (push, pop, isEmpty, top, checkStack, setStackSize, or
92         String input = sc.nextLine();
93
94         switch (input) {
95             case "push":
96                 System.out.println("Enter a value to push onto the stack: ");
97                 String value = sc.nextLine();
98                 stack.push(value);
99                 break;
100             case "pop":
101                 stack.pop();
102                 break;
103             case "isEmpty":
104                 stack.isEmpty();
105                 break;
106             case "top":
107                 stack.top();
108                 break;
109             case "checkStack":
110                 stack.checkStack();
111                 break;
112             case "setStackSize":
113                 System.out.println("Enter a new stack size:");
114                 String sizeInput = sc.nextLine();
115                 int size = Integer.parseInt(sizeInput);
116                 stack.setStackSize(size);
117                 break;
118             case "exit":
119                 sc.close();
120                 return;
121             default:
122                 System.out.println("Invalid command. Please try again.");
123                 break;
124         }
125     }
126 }
127 }
```

| | |
|--|--|
| | |
|--|--|

5. สรุปผลการปฏิบัติการ

ศึกษาคำสั่งพื้นฐานของอาร์เรย์และใช้คำสั่งพื้นฐานมาแก้โจทย์ปัญหา

.....

.....

.....

.....

.....

6. คำถามท้ายการทดลอง

6.1. ข้อควรระวังในการใช้งาน Array ในภาษาจาวาคืออะไร ?

เมื่อประกาศตัวแปรอาร์เรย์และกำหนดขนาดของอาร์เรย์แล้วอาจเขียนหรืออ่านเกินกว่าสมาชิกของอาร์เรย์ ได้โดย คอมไพเลอร์จะไม่แจ้งเตือนแต่การเขียนข้อมูลเกินกว่าขนาดของอาร์เรย์ที่จองไว้จะทำให้ข้อมูลที่เขียนลงไปนั้นจะไปทับข้อมูลอื่นหรือ ไปทับส่วนของโปรแกรมอาจจะทำให้การทำงานของโปรแกรมเสียหายได้

.....

.....

.....

.....

6.2. ข้อควรระวังในการใช้งาน String ในภาษาจาวาคืออะไร ?

ระวังเรื่องสัญลักษณ์พิเศษ

.....

.....

.....

.....

6.3. ฟังก์ชันในภาษาจาวาไม่สามารถใช้งานแบบ Pass by reference ในภาษาซีได้ คุณมีแนวทางการแก้ไขปัญหานี้ได้อย่างไร ?

ในภาษา java จะใช้ pass by value

.....

.....

.....

6.4. โครงสร้างข้อมูลแบบ Stack แตกต่างกับโครงสร้างข้อมูลแบบ Array อย่างไร ?

Array จะเหมือนกับตารางส่วน stack จะเหมือนกล่อง

.....

.....

.....