

Overloading คือ การเขียนชุดของ method ที่มีชื่อเหมือนกัน แต่มีการทำงานที่แตกต่างกัน  
ซึ่งถือว่าเป็นคุณลักษณะพิเศษหนึ่งของการเขียนโปรแกรมเริ่มต้น คือ polymorphism

ให้ศึกษาและทดลองพิมพ์ Code ตัวอย่างการสร้างเมธอดจากตัวอย่างต่อไปนี้

### 1.1

```
Class A {  
    int x = 10, y = 5;  
    void methodA (int a, int b) {  
        a = x;  
        b = y;  
    }  
}  
class TestPassByValue {  
    public static void main (String args[]) {  
        int var1, var2;  
        A objA = new A();  
        var1 = 5;  
        var2 = 5;  
        System.out.println("var1(before) = " + var1);  
        System.out.println("var2(before) = " + var2);  
        objA.methodA(var1, var2);  
        System.out.println("var1(after) = " + var1);  
        System.out.println("var2(after) = " + var2);  
    }  
}
```

จากโปรแกรมดังกล่าวผลลัพธ์ของโปรแกรมคือ

var1(before) = 5      var1(after) = 5  
var2(before) = 5      var2(after) = 5

ให้อธิบายการทำงานของการใช้คำสั่ง `objA.methodA(var1, var2);`

### 1.2

```
class A {  
    int x = 10, y = 10;  
    void methodA (B objB) {  
        objB.var1 = x;  
        objB.var2 = y;  
    }  
}  
class B {  
    int var1, var2;  
}  
class TestPassByValue1 {  
    public static void main(String args[]) {  
        B objB = new B();  
        A objA = new A();  
        objB.var1 = 5;  
        objB.var2 = 5;  
        System.out.println("var1(before) = "+objB.var1);  
        System.out.println("var2(before) = "+objB.var2);  
    }  
}
```

ชื่อ-นามสกุล นพดล ภูมิพล รหัสประจำตัวนักศึกษา 620220302  
ปีการศึกษา 2567

```
objA.methodA(objB);
System.out.println("var1(after)="+objB.var1);
System.out.println("var2(after)="+objB.var2);
```

จากโปรแกรมดังกล่าวผลลัพธ์ของโปรแกรมคือ

var1(before) : 5 var1(after) : 10  
var2(before) : 5 var2(after) : 10

ให้อธิบายการทำงานของการใช้คำสั่ง objA.methodA(objB); *ผ่านอินซีร์ฟาร์ Pass By Reference*

2. จงอธิบายว่าเหตุใดโปรแกรมด้านล่างจึง Compiles ไม่ผ่าน *因为 args ที่ objB ไม่ได้กำหนด*

| โปรแกรม   | เหตุผล   |
|---|--|
| <pre>public class TestProgram{     public static void main(String[] args){         int i=f(2, 3);     }     public static int f(int a, int b){         return Math.pow(a, b) + Math.pow(b, a);     } }</pre>  | <i>return type Math.pow<br/>= double ไม่ f ที่ int</i> |
| <pre>public class Test {     public static method1(int n, m) {         n += m;         method2(3.4);     }      public int method2(int n) {         if (n &gt; 0) return 1;         else if (n == 0) return 0;         else if (n &lt; 0) return -1;     } }</pre>                              | <i>Test Math(String[]) args</i>                        |
| <pre>public class Test {     public static void main(String[] args) {         nPrintln (5, "Welcome to Java!");     }     public static void nPrintln(String message, int n) {         int n = 1;         for (int i = 0; i &lt; n; i++)             System.out.println(message);     } }</pre> | <i>Pass parameter<br/>ไม่ถูกต้อง</i>                   |

3. จงเขียน method header สำหรับ method m() ในแต่ละข้อต่อไปนี้

a. int i=m(1, 1);

static int

b. float f=m(Math.exp(5));

static double

c. static String

d. static Csstudent

e. static void

## 4. จงอธิบาย output ที่ได้จาก main() พังก์ชันต่อไปนี้

| โปรแกรม  | ผล                                |
|--|-----------------------------------|
| <pre>public class TestProgram{     public static void main(String[] args) {         System.out.println(g("A"));     }     public static String f(){         System.out.println("A");         return "A";     }     public static String g(String s){         return f()+s;     } }</pre>   | A<br>AA                           |
| <pre>public class Test {     public static void main(String[] args) {         int max = 0;         max(1, 2, max);         System.out.println(max);     }     public static void max(         int value1, int value2, int max) {         if (value1 &gt; value2)             max = value1;         else             max = value2;     } }</pre>  | ○                                 |
| <pre>public class Test {     public static void main(String[] args) {         int i = 0;         while (i &lt;= 4) {             method1(i);             i++;         }         System.out.println("i is " + i);     }     public static void method1(int i) {         do {             if (i % 3 != 0)                 System.out.print(i + " ");             i--;         }         while (i &gt;= 1);         System.out.println();     } }</pre> | 1<br>2 1<br>2 1<br>4 2 7<br>i iss |