

CHAPTER

1

การเขียนโปรแกรมภาษาจาวาขั้นพื้นฐาน : ตอนที่ 1

(Basic Java Programming : Part I)

1

แนะนำภาษาจาวา (Introduction to Java Language)

1. ประวัติและความเป็นมาของภาษาจาวา

- 1) จาวาเป็นภาษาโปรแกรมเชิงวัตถุซึ่งถูกพัฒนาโดย เจมส์ กอสลิง (James Gosling) และวิศวกรจากบริษัทซันไมโครซิสเต็มส์ (Sun Microsystems)
- 2) ภาษาจาวาถูกพัฒนาขึ้นในปี พ.ศ. 2534 (ค.ศ. 1991) โดยเป็นส่วนหนึ่งของโครงการกรีน (Green Project) สำเร็จและออกใช้งานในปี พ.ศ. 2538 (ค.ศ. 1995) โดยซันไมโครซิสเต็มส์ แต่ปัจจุบันได้ รับการผนวกกิจการต่างๆ เข้ากับบริษัทโอราเคิล (Oracle Corporation) แล้ว
- 3) ภาษาจาวาพัฒนามาจากภาษาซี (C) และภาษาซีพลัสพลัส (C++) แต่มีโครงสร้างที่ง่ายกว่า โดยมีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัสที่มีการทำงานซับซ้อนกว่า
- 4) เริ่มแรกนั้นภาษาจาวามีชื่อว่าภาษาโอ๊ค (Oak) ซึ่งตั้งชื่อตามต้นโอ๊คที่อยู่ใกล้ที่ทำงานของเจมส์ กอสลิง แต่มีปัญหาทางลิขสิทธิ์จึงเปลี่ยนชื่อมาเป็น "จาวา" ซึ่งเป็นชื่อของพันธุ์กาแฟโดยได้มาจากการสุ่มชื่อ (Random) คำที่เจมส์ กอสลิงชื่นชอบซึ่งคำว่าจาวาเป็นหนึ่งในนั้น (เจมส์มีรสนิยมนชอบดื่มกาแฟ)
- 5) โลโก้ (Logo) ของจาวาเป็นรูปกาแฟที่อยู่ภายในถ้วยที่ยังร้อนอยู่ และสัญลักษณ์นำโชค (Mascot) ของจาวาคือรูปเมลดิกาแฟจูกี้แดงที่ชื่อว่า Duke
- 6) ภาษาจาวามีทั้งหมด 22 รุ่น โดยรุ่นปัจจุบันคือ Java SE Development Kit 22 (JDK 22) ซึ่งเริ่มออกใช้งานเมื่อวันที่ 16 เมษายน ค.ศ. 2024
- 7) ภาษาจาวาเป็นภาษามัธยมใหม่ที่น่าสนใจแนวคิดการเขียนโปรแกรมแบบเชิงวัตถุ หรือที่เรียกกันว่า Object-Oriented Programming (OOP)
- 8) ภาษาจาวาและทรัพยากรของจาวาเป็น Freeware และ Open Source สามารถดาวน์โหลดได้จากเว็บไซต์อย่างเป็นทางการของโอราเคิล
- 9) ภาษาจาวาโดยทั่วไปจะถูกคอมไพล์ (Compile) ให้เป็นไบต์โค้ด (Bytecode) แล้วทำการประมวลผลหรือรัน (Run) ด้วยจาวาเวอร์ชวลแมชีน (Java Virtual Machine: JVM)
- 10) การเริ่มต้นเขียนโปรแกรมด้วยภาษาจาวาจะต้องติดตั้งเครื่องมือ 2 ส่วนหลักคือ Java SE และ Eclipse



2. องค์ประกอบหลักของโปรแกรมภาษาจาวา

```
1 public class HelloJava {  
2     public static void main(String[] args) {  
3         System.out.println("Hello Java");  
4     }  
5 }
```

ตัวอย่างของโปรแกรมภาษาจาวาแบบขั้นพื้นฐาน (โปรแกรมอย่างง่าย)

องค์ประกอบของโปรแกรมภาษาจาวามี 3 ส่วนหลัก ได้แก่ คลาส (Class) เมทอด (Method) และคำสั่ง (Statement)

2 การประมวลผลข้อมูล (Data Processing)

1. ประเภทข้อมูลหรือชนิดตัวแปร (Data Types)

1) ประเภทข้อมูลแบบดั้งเดิม (Primitive Data Types) แบ่งออกเป็น 4 กลุ่ม ได้แก่

(1) ตัวเลขจำนวนเต็ม (Integer) คือ จำนวนเต็มบวก จำนวนเต็มลบ จำนวนเต็มศูนย์

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
byte	8 bits	-128 ถึง 127	0
short	16 bits	-32,768 ถึง 32,767	0
int	32 bits	-2,147,483,648 ถึง 2,147,483,647	0
long	64 bits	-9,223,372,036,854,775,808 ถึง 9,223,372,036,854,775,807	0L

(2) ตัวเลขจำนวนจริง (Floating Point/Real Number) คือ จำนวนที่มีจุดทศนิยม

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
float	32 bits	-3.40292347E+38 ถึง 3.40292347E+38	0.0F
double	64 bits	-1.79769313486231570E+308 ถึง 1.79769313486231570E+308	0.0D

(3) ตัวอักขระ (Character) คือ ตัวอักษร (Letter) หรือตัวเลข (Digit) หรือสัญลักษณ์ (Symbol) เพียง 1 ตัวเท่านั้น

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
char	16 bits	'\u0000' ถึง '\uFFFF' (0 - 65535) เช่น 'A' หรือ 'a' โดยจะครอบด้วยเครื่องหมาย '...'	'\u0000'

(4) ตัวตรรกะ (Boolean) คือ ค่าความจริง ซึ่งมีเพียงสองค่าคือ จริง (True) หรือ เท็จ (False)

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
boolean	ไม่ระบุ	true หรือ false	false

2) ประเภทข้อมูลแบบอ้างอิง (Reference Data Types) แบ่งออกเป็น 1 กลุ่มได้แก่ (ความจริงแล้วยังมีมากกว่านี้)

(1) สตริง (String) หรือข้อความ คือ กลุ่มของอักขระหรือสายอักขระตั้งแต่ 0 ตัวขึ้นไปมาเรียงต่อกัน

ชนิดตัวแปร	ขนาดตัวแปร	ช่วงของข้อมูล	ค่าเริ่มต้น
String	ไม่ระบุ	เช่น "Java Chula" หรือ "1" หรือ "F50" หรือ "" (เรียกว่า Empty String) โดยจะครอบด้วยเครื่องหมาย "..."	null

3) การเรียงลำดับขนาดและศักดิ์ศรีของประเภทข้อมูล (เฉพาะตัวเลข) มีดังนี้

byte	short	int	long	float	double
เล็ก					ใหญ่

2. ค่าของตัวแปรตามประเภทข้อมูล

1) ค่าของจำนวนเต็ม

ประเภทข้อมูล	รูปแบบการเขียน / ตัวอย่างค่าที่จัดเก็บ
int (ฐานสิบ)	เขียนเลขจำนวนเต็มทั่วไป เช่น <code>int dec = 26;</code>
int (ฐานแปด)	เขียน 0 (ศูนย์) นำหน้าเลขจำนวนเต็ม เช่น <code>int oct = 032;</code>
int (ฐานสิบหก)	เขียน 0x หรือ 0X นำหน้าเลขจำนวนเต็ม เช่น <code>int hex = 0x1A;</code>

2) ค่าของจำนวนจริง

ประเภทข้อมูล	รูปแบบการเขียน / ตัวอย่างค่าที่จัดเก็บ
double (รูปทั่วไป)	เขียนเลขจำนวนจริงทั่วไป เช่น <code>double d = 1.3;</code>
double (รูป $a \times 10^b$)	เขียนค่า a ในรูปของเลขจำนวนเต็มหรือเลขจำนวนจริงทั่วไป ตามด้วยอักษร e หรือ E และตามด้วยค่า b ในรูปของเลขจำนวนเต็ม (เท่านั้น) เช่น <ul style="list-style-type: none"> <code>double s = 93.478e8;</code> มีค่าเท่ากับ 93.478×10^8 <code>double t = 631E-5;</code> มีค่าเท่ากับ 631×10^{-5}

3) ค่าของอักขระและสตริง

ประเภทข้อมูล	รูปแบบการเขียน / ตัวอย่างค่าที่จัดเก็บ
char (รูปทั่วไป)	เขียนอักขระทั่วไปล้อมด้วย Single Quote เช่น <code>char c = 'a';</code>
String (รูปทั่วไป)	เขียนสายอักขระล้อมด้วย Double Quote เช่น <code>String s = "Bank";</code>
String (รูปอ็อบเจกต์)	สร้างอ็อบเจกต์จากคลาส String เช่น <code>String t = new String("CU");</code>

โจทย์ข้อที่ 1 [ระดับง่าย] จงพิจารณาคำสั่งต่อไปนี้ถูก (✓) หรือผิด (✗) พร้อมบอกเหตุผล (20 คะแนน)

- ☐ `integer x;`
- ☐ `int x = 1.5;`
- ☐ `byte x = 1000;`
- ☐ `int null = 13;`
- ☐ `Int y = 2564;`
- ☐ `float cal = 15.76392;`
- ☐ `String tab = "\tab";`
- ☐ `char c = 'Tuksin';`
- ☐ `String String = 'Prayut';`
- ☐ `double i, j = 0, final;`
- ☐ `String 3Type = "One Two Three";`
- ☐ `short ____$abcdefghijklmnopqrstuvwxyz0123456789$____;`
- ☐ `boolean flag = 1;`
- ☐ `String num = "" + 2021 + 2022;`

- 15) `String ตัวเลข = "\\t";`
- 16) `char newLine = '\\n';`
- 17) `String _ = "はじめまして";`
- 18) `long LONG = 10F;`
- 19) `int ch = 0xBEEF;`
- 20) `int integer = 10e1;`

3. การรับเข้าข้อมูล (Input the Data)

การรับเข้าข้อมูลในภาษาจาวามีอยู่หลายวิธีด้วยกัน แต่ในเอกสารชุดนี้จะนำเสนอการรับเข้าข้อมูลที่นิยมใช้กันมากในปัจจุบัน นั่นคือการรับเข้าข้อมูลด้วยคลาส `Scanner` ซึ่งมีขั้นตอนการใช้งานดังต่อไปนี้

1) การนำเข้าคลาส `Scanner` และการประกาศ และการสร้างตัวอ่านจากแป้นพิมพ์ ดังตัวอย่างต่อไปนี้

```

Scanner <ชื่อตัวอ่าน> = new Scanner(System.in);

1  import java.util.Scanner;
2  public class Test {
3      public static void main(String[] args) {
4          Scanner kb = new Scanner(System.in);
5          ...
6      }
7  }

```

ชื่อตัวอ่านจะชื่อว่าอะไรก็ได้ แต่ต้องมีรูปแบบการประกาศแบบนี้เสมอ (ต้องท่องจำให้ขึ้นใจ)

ตัวอ่านชื่อว่า kb

System.in หมายถึง แป้นพิมพ์ (Keyboard)

2) การรับค่าข้อมูลด้วยตัวอ่านจากแป้นพิมพ์ ตัวอ่านจากแป้นพิมพ์ที่สร้างขึ้นสามารถรับค่าข้อมูลประเภทจำนวนเต็ม จำนวนจริง และสตริง โดยใช้คำสั่งและเมทอดดังต่อไปนี้

(1) คำสั่งรับค่าจำนวนเต็มประเภท `int` โดยใช้เมทอด `nextInt()`

```
int <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.nextInt();
```

หลังชื่อเมทอด `nextInt` ต้องตามด้วยวงเล็บเสมอ ห้ามลืมโดยเด็ดขาด

ตัวอย่าง `int num = kb.nextInt();` (รับค่า `int` จากตัวอ่าน `kb` มาเก็บไว้ที่ตัวแปร `num`)

(2) คำสั่งรับค่าจำนวนจริงประเภท `double` โดยใช้เมทอด `nextDouble()`

```
double <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.nextDouble();
```

ตัวอย่าง `double d = kb.nextDouble();` (รับค่า `double` จากตัวอ่าน `kb` มาเก็บไว้ที่ตัวแปร `d`)

(3) คำสั่งรับค่าข้อมูลประเภท `String` โดยใช้เมทอด `nextLine()` และเมทอด `next()`

```
String <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.nextLine();
```

```
String <ชื่อตัวแปร> = <ชื่อตัวอ่าน>.next();
```

ตัวอย่าง `String s1 = kb.nextLine();` (รับค่า `String` ทั้งบรรทัด จากตัวอ่าน `kb` มาเก็บไว้ที่ตัวแปร `s1`)

`String s1 = kb.next();` (รับค่า `String` คำแรก จากตัวอ่าน `kb` มาเก็บไว้ที่ตัวแปร `s2`)

4. การส่งออกข้อมูล (Output the Data)

การส่งออกข้อมูลหรือการแสดงผลข้อมูลสามารถทำได้หลายช่องทาง แต่ช่องทางที่พื้นฐานที่สุดนั้นก็คือจอภาพ (Monitor/Screen) โดยมีรายละเอียดดังนี้

1) การแสดงผลออกทางจอภาพแบบไม่ขึ้นบรรทัดใหม่ โดยใช้เมทอด `print()`

```
System.out.print(...);
```

System.out หมายถึงจอภาพ (Monitor)

2) การแสดงผลออกทางจอภาพแบบขึ้นบรรทัดใหม่ โดยใช้เมทอด `println()`

```
System.out.println(...);
```

3) สิ่งที่สามารถระบุและแสดงผลภายในวงเล็บของเมทอด `print()` และ `println()` ได้นั้นมีดังนี้

- (1) ตัวแปร เช่น `System.out.print(num);` (ตัวแปร `num` ต้องมีการประกาศไว้ก่อนหน้านี้อยู่แล้ว)
- (2) ตัวเลข เช่น `System.out.print(13);`
- (3) นิพจน์ เช่น `System.out.print(x * y + (10 - z));`
- (4) ข้อความหรือสตริง เช่น `System.out.print("Java");`
- (5) ข้อ 1-4 ปะปนกัน เช่น `System.out.print("Money = " + (x * y + 92.4) + " Baht");`

ใช้เครื่องหมาย + เป็นตัวเชื่อมแต่ละช่วงเข้าด้วยกัน

5. ตัวดำเนินการ (Operators) แบ่งออกเป็น 5 กลุ่มหลักดังต่อไปนี้

1) ตัวดำเนินการทางคณิตศาสตร์ (Arithmetic Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
+	การบวก	$x + y$ หรือ $a + b + c$
-	การลบ	$x - y$ หรือ $a - b - c$
*	การคูณ	$x * y$ หรือ $a * b * c$
/	การหาร	x / y หรือ $a / b / c$
%	การหารเอาแต่เศษ (Modulo)	$x \% y$ หรือ $a \% b \% c$

2) ตัวดำเนินการเพิ่มค่าและลดค่า (Increment and Decrement Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
<code>x++</code>	ใช้งานตัวแปร <code>x</code> ก่อนเพิ่มค่าขึ้นอีก 1	ให้ <code>x = 2</code> , <code>y = 1</code> จะได้ว่า <code>x++ + y</code> เท่ากับ 3
<code>++x</code>	เพิ่มค่าขึ้นอีก 1 ก่อนใช้งานตัวแปร <code>x</code>	ให้ <code>x = 2</code> , <code>y = 1</code> จะได้ว่า <code>++x + y</code> เท่ากับ 4
<code>x--</code>	ใช้งานตัวแปร <code>x</code> ก่อนลดค่าลงอีก 1	ให้ <code>x = 2</code> , <code>y = 1</code> จะได้ว่า <code>x-- + y</code> เท่ากับ 3
<code>--x</code>	ลดค่าลงอีก 1 ก่อนใช้งานตัวแปร <code>x</code>	ให้ <code>x = 2</code> , <code>y = 1</code> จะได้ว่า <code>--x + y</code> เท่ากับ 2

3) ตัวดำเนินการเปรียบเทียบค่า (Comparison/Relational Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
<code>==</code>	เท่ากัน	<code>x == y</code> (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)

สัญลักษณ์	ความหมาย	ตัวอย่าง
<code>!=</code>	ไม่เท่ากัน	<code>x != y</code> (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
<code><</code>	น้อยกว่า	<code>x < y</code> (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
<code>></code>	มากกว่า	<code>x > y</code> (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
<code><=</code>	น้อยกว่าหรือเท่ากัน	<code>x <= y</code> (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)
<code>>=</code>	มากกว่าหรือเท่ากัน	<code>x >= y</code> (คำตอบเป็นค่าจริงหรือเท็จเท่านั้น)

4) ตัวดำเนินการทางตรรกศาสตร์ (Logical Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
<code>&&</code>	และ (AND)	<code>(x >= y) && (x != 0)</code>
<code> </code>	หรือ (OR)	<code>(x == y) (x == z)</code>
<code>!</code>	นิเสธ (NOT)	<code>!(x <= y)</code>

5) ตัวดำเนินการกำหนดค่า (Assignment Operators) ได้แก่

สัญลักษณ์	ความหมาย	ตัวอย่าง
<code>=</code>	เท่ากับ	<code>x = y</code> นำค่า <code>y</code> เก็บไว้ใน <code>x</code>
<code>+=</code>	<code>x += y</code> มีค่าเท่ากับ <code>x = x + y</code>	<code>x += 14</code> จะได้ว่า <code>x = x + 14</code>
<code>-=</code>	<code>x -= y</code> มีค่าเท่ากับ <code>x = x - y</code>	<code>x -= 1</code> จะได้ว่า <code>x = x - 1</code>
<code>*=</code>	<code>x *= y</code> มีค่าเท่ากับ <code>x = x * y</code>	<code>x *= y--</code> จะได้ว่า <code>x = x * y--</code>
<code>/=</code>	<code>x /= y</code> มีค่าเท่ากับ <code>x = x / y</code>	<code>x /= 5.8</code> จะได้ว่า <code>x = x / 5.8</code>
<code>%=</code>	<code>x %= y</code> มีค่าเท่ากับ <code>x = x % y</code>	<code>x %= 7</code> จะได้ว่า <code>x = x % 7</code>

หมายเหตุ ตัวดำเนินการทั้ง 5 กลุ่มนี้สามารถใช้งานได้โดยการพิมพ์เครื่องหมายของตัวดำเนินการจากทางแป้นพิมพ์

6. คลาสมาตรฐานทางคณิตศาสตร์ (Mathematic Class)

ยังมีตัวดำเนินการบางประเภทที่ไม่สามารถพิมพ์สัญลักษณ์ผ่านทางแป้นพิมพ์ได้ ดังนั้นจึงต้องเรียกใช้งานตัวดำเนินการเหล่านี้ผ่านทางเมทอดของคลาสมาตรฐานทางคณิตศาสตร์แทน ซึ่งมีดังต่อไปนี้

ลำดับ	ประเภทข้อมูลที่คืนค่า	รูปแบบคำสั่ง	ความหมายและหน้าที่
1.	double	<code>Math.PI</code>	ค่าคงที่ PI ซึ่งเท่ากับ 3.141592653589793...
2.	double	<code>Math.toRadians(d)</code>	เปลี่ยนองศาเป็นเรเดียน
3.	double	<code>Math.toDegrees(r)</code>	เปลี่ยนเรเดียนเป็นองศา
4.	double	<code>Math.sin(r)</code>	คำนวณค่าไซน์ (Sine) (ใช้มุมเรเดียน)
5.	double	<code>Math.cos(r)</code>	คำนวณค่าโคไซน์ (Cosine) (ใช้มุมเรเดียน)
6.	double	<code>Math.tan(r)</code>	คำนวณค่าแทนเจนต์ (Tangent) (ใช้มุมเรเดียน)
7.	double	<code>Math.sqrt(x)</code>	คำนวณค่ารากที่สอง

ลำดับ	ประเภทข้อมูลที่คืนค่า	รูปแบบคำสั่ง	ความหมายและหน้าที่
8.	int, double	Math.abs(x)	คำนวณค่าสัมบูรณ์ (Absolute)
9.	double	Math.exp(x)	คำนวณค่าเลขยกกำลัง e^x
10.	double	Math.pow(x, y)	คำนวณค่าเลขยกกำลัง x^y
11.	int, double	Math.max(x, y)	หาค่ามากกว่าระหว่าง x กับ y
12.	int, double	Math.min(x, y)	หาค่าน้อยระหว่าง x กับ y
13.	double	Math.random() สุ่มตัวเลขช่วง [0.0, 1.0)	int x = (int) (Math.random() * (b-a+1)) + a สร้างตัวเลขสุ่มจำนวนเต็มในช่วง a ถึง b

7. กฎการพิจารณาลำดับของตัวดำเนินการ (เฉพาะที่ใช้กันบ่อยๆ)

ลำดับ	ชื่อตัวดำเนินการ / ตัวดำเนินการ		ตัวอย่างการทำงาน
1.	วงเล็บ (Parentheses)	(...)	$x - (2 + y)$ จะทำ $(2 + y)$ ก่อนนำ x มาลบออก
2.	เรียกเมทอด (Method Call)	()	$a + \text{Math.sqrt}(4)$ จะทำ $\text{Math.sqrt}(4)$ ก่อน
3.	ค่าบวก (Unary Plus)	+	$2 * +a$ จะทำ $+a$ ก่อนนำ 2 ไปคูณกับคำตอบ
	ค่าลบ (Unary Minus)	-	$-a / 3$ จะทำ $-a$ ก่อนนำคำตอบไปหาร 3
	นิเสธ (Logical NOT)	!	$!a b$ จะทำ $!a$ ก่อนทำ $ $
4.	คูณ (Multiplication)	*	$x * 3 + 2$ จะทำ $x * 3$ ก่อนนำคำตอบไปบวก 2
	หาร (Division)	/	$3 - 5 / x$ จะทำ $5 / x$ ก่อนนำ 3 มาลบออก
	หารเอาแต่เศษ (Modulo)	%	$y \% 2 + 1$ จะทำ $y \% 2$ ก่อนนำคำตอบไปบวก 1
5.	บวก (Addition)	+	$x + 17 > 19$ จะทำ $x + 17$ ก่อนทำ $>$
	ลบ (Subtraction)	-	$x - 9.5 = 42.5$ จะทำ $x - 9.5$ ก่อนทำ $=$
6.	การเปรียบเทียบค่า : น้อยกว่า (<), น้อยกว่าเท่ากับ (<=), มากกว่า (>), มากกว่าเท่ากับ (>=)		
7.	การเปรียบเทียบค่า : เท่ากัน (==), ไม่เท่ากัน (!=)		
8.	และ (Logical AND)	&&	$a b \&\& c$ จะทำ $\&\&$ ก่อน $ $
9.	หรือ (Logical OR)		$x = d > 3 d <= 15$ จะทำ $ $ ก่อนทำ $=$
10.	เท่ากับ (Assignment)	=	จะถูกพิจารณาหลังสุด

หมายเหตุ ในกรณีที่ตัวดำเนินการมีลำดับความสำคัญเท่ากัน ให้พิจารณาการทำงานจากซ้ายไปขวา

8. การเปลี่ยนประเภทข้อมูล (Type Conversions)

การเปลี่ยนประเภทข้อมูลสามารถทำได้ 2 วิธีดังต่อไปนี้

- 1) **เปลี่ยนแบบอัตโนมัติ** โดยทั่วไปคอมไพเลอร์ของภาษาจาวาจะปรับเปลี่ยนประเภทข้อมูลให้อัตโนมัติตามความเหมาะสม ซึ่งเป็นไปตามความสัมพันธ์ทั้ง 12 รูปแบบที่ได้กล่าวไปในหัวข้อที่แล้ว ดังตัวอย่างต่อไปนี้

```
double a = 34 + 1.0F; //a = 35.0
float b = 4L / 3; //b = 1.0F
long c = 100; //c = 100L
```

```
int d = 'A'; //d = 65
char e = 65; //e = 'A'
String f = 15 //Error
```

2) เปลี่ยนโดยการบังคับตามทีระบุ (Casting) จะเป็นการบังคับหรือกำหนดประเภทข้อมูลที่ต้องการจะให้ เป็น โดยการใส่วงเล็บชื่อประเภทข้อมูลหน้าข้อมูลที่จะเปลี่ยน ดังรูปแบบคำสั่งและตัวอย่างต่อไปนี้

(<ประเภทข้อมูล>) <ข้อมูล/ตัวแปร>

ใช้กับข้อมูลประเภทตัวเลข (เต็ม/จริง) และอักขระเท่านั้น

```
byte b = (byte) (5L * 2); //b = 10
int x = (int) 5.989777; //x = 5
```

```
int ten = (int) "10"; //Error
boolean f = (boolean) 1; //Error
```

โจทย์ข้อที่ 2 [ระดับง่าย] จงระบุประเภทข้อมูลที่เป็นผลลัพธ์สุดท้ายและคำตอบจากนิพจน์ต่อไปนี้ (11 คะแนน)

ข้อ	นิพจน์	ประเภทข้อมูลสุดท้าย	คำตอบ
1.	(byte) (3 + 5.4) / 7		
2.	(3 + 5.4) / (float) 7		
3.	(float) ((3 + 2) / 7)		
4.	(5 + 4) + ((3 + 2L) - 1)		
5.	!("Java" == "Jaba")		
6.	(13 / 5) + (2 / 3)		
7.	4E1 / 2		
*8.	char c = 67 + (int) Math.PI		
9.	!false true && false !true		
10.	'A' >= 'a'		
11.	++5 + 6-- * 7++ - Math.sin(0.5)		

โจทย์ข้อที่ 3 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับอุณหภูมิในหน่วยองศาเซลเซียส (°C) เข้ามาทางแป้นพิมพ์แล้วแปลงค่านั้นเป็นหน่วยองศาฟาเรนไฮต์ (°F) และเคลวิน (K) ตามลำดับ โดยกำหนดให้ $K = ^\circ C + 273.15$ และ $^{\circ}F = (1.8 \times ^\circ C) + 32$ (10 คะแนน)

```
Enter Celsius: 50.0
Kelvin: 323.15
Fahrenheit: 122.0
```

โจทย์ข้อที่ 4 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลข

จำนวนจริง 7 จำนวนเข้ามาทางแป้นพิมพ์ โดยให้พิมพ์รับค่าตัวแปรทั้งเจ็ด พร้อมกัน (คั่นด้วยการเว้นวรรค) ดังตัวอย่าง แล้วทำการคำนวณหาเฉลี่ยของตัวเลขทั้ง 7 ตัวนั้นพร้อมทั้งแสดงผลที่ได้ออกทางจอภาพให้สวยงาม (10 คะแนน)

```
Number: 1 2 3 4 5 6 7
Average Number: 4.0
```


โจทย์ข้อที่ 5 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับค่ามุมในหน่วยองศาจากทางแป้นพิมพ์ เพื่อคำนวณหาค่าไซน์ (Sine) พร้อมทั้งแสดงผลลัพธ์ออกทางจอภาพให้เหมือนกับตัวอย่าง (10 คะแนน)

```
Enter Degree: 30.0
sin(30.0): 0.5

Enter Degree: 65.5
sin(65.5): 0.90996...
```

โจทย์ข้อที่ 6 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับค่าพิกัดของจุดสองจุดใด ๆ เข้ามาทางแป้นพิมพ์ซึ่งประกอบไปด้วยจุด (x_1, y_1) และ จุด (x_2, y_2) แล้วทำการคำนวณหาระยะห่างระหว่างจุดสองจุดนั้นตามสมการต่อไปนี้ (10 คะแนน)

$$\text{ระยะห่างระหว่างจุด} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

โจทย์ข้อที่ 7 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็ม 5 จำนวนเข้ามาทางแป้นพิมพ์ โดยให้รับค่าตัวแปรทั้งหมดภายในบรรทัดเดียวกัน แล้วทำการคำนวณหาค่าพิสัย (Range) และค่ากึ่งกลางพิสัย (Middle Range) ของตัวเลขทั้ง 5 ค่านั้น แล้วแสดงผลลัพธ์ออกทางจอภาพ (10 คะแนน)

```
Number: 19 8 16 25 6
Maximum: 25
Minimum: 6
Range: 19
Middle Range: 15.5
```

โจทย์ข้อที่ 8 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อสลับค่าตัวเลขจำนวนเต็มสองจำนวนที่รับเข้ามาทางแป้นพิมพ์เก็บไว้ในตัวแปร a และ b ตามลำดับ โดยที่ค่าของตัวแปร a หลังการสลับค่าจะเท่ากับค่าเก่าของตัวแปร b และค่าของตัวแปร b หลังการสลับค่าจะเก็บค่าเก่าของตัวแปร a พร้อมทั้งแสดงผลลัพธ์ออกทางจอภาพตามตัวอย่างต่อไปนี้ (10 คะแนน)

```
Enter a: 13
Enter b: 21
Value of a: 21
Value of b: 13
```

ตัวเลข 13 และ 21 คือค่าที่รับเข้ามา (จะเป็นตัวเลขจำนวนจริงอะไรก็ได้)

ผลลัพธ์ที่ได้จะต้องสลับค่าภายในตัวแปรกันจริงๆ ไม่ใช่แค่แสดงผลลัพธ์เพื่อสลับค่าเฉยๆ)

โจทย์ข้อที่ 9 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็ม 3 จำนวนเข้ามาทางแป้นพิมพ์เก็บไว้ในตัวแปร a, b และ c ตามลำดับ (ให้รับค่าภายในบรรทัดเดียวกัน) แล้วทำการเรียงลำดับตัวเลขนั้นจากมากไปหาน้อย พร้อมทั้งแสดงผลลัพธ์ที่ได้ขึ้นบนจอภาพตามตัวอย่างที่กำหนดให้ (10 คะแนน)

```
Enter Number: 23 14 19
Sorted Number: 14 < 19 < 23
```

โจทย์ข้อที่ 10 [ระดับยาก] ธนาคารสวนสุนันทาแห่งประเทศไทยได้จ้างให้นักศึกษาชั้นปีที่ 1 ของคณะวิศวกรรมศาสตร์ และคณะวิทยาศาสตร์เทคโนโลยี ซึ่งกำลังเรียนวิชาการเขียนโปรแกรมด้วยภาษาจาวาเบื้องต้น เพื่อสร้างเครื่องคำนวณจำนวนเงิน (Money Machine) โดยเมื่อผู้ใช้ระบุจำนวนเงินเข้ามา โปรแกรมจะทำการคำนวณว่าจะต้องใช้ธนบัตรใบละ 1000 บาท 500 บาท 100 บาท 50 บาท และ 20 บาท อย่างละกี่ใบ และเหรียญ 10 บาท 5 บาท 2 บาท และ 1 บาท อย่างละกี่เหรียญ โดยมีตัวอย่างของผลการทำงานดังต่อไปนี้ (15 คะแนน)

```
Money: 2897
1000 Baht : 2
500 Baht  : 1
100 Baht   : 3
50 Baht    : 1
20 Baht    : 2
10 Baht    : 0
5 Baht     : 1
2 Baht     : 1
1 Baht     : 0
```

3

การตัดสินใจ (Decisions)

1. คำสั่งตัดสินใจแบบ 1 เงื่อนไข

คำสั่งตัดสินใจแบบ 1 เงื่อนไขนี้จะใช้คำสั่ง `if` เพียงอย่างเดียวในการทำงาน โดยมีรูปแบบดังนี้

```
1  if (Condition) {
2      Statement;
3  }
```

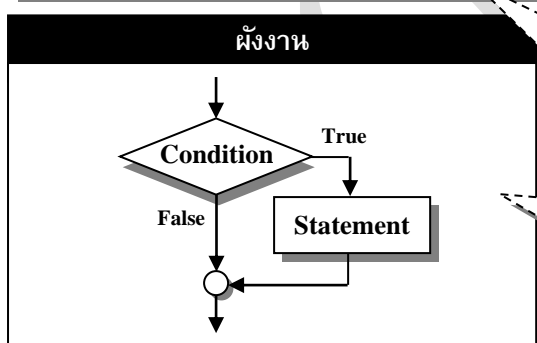
เงื่อนไข (Condition) จะเป็นนิพจน์ตรรกะ (Boolean Expression) ที่แสดงถึงค่าความจริงว่าเมื่อใดจะทำการในคำสั่ง `if` หรือเมื่อใดไม่ทำ

ภายในคำสั่ง `if` สามารถมีคำสั่งได้มากกว่า 1 คำสั่ง แต่ถ้ามีเพียง 1 คำสั่ง ไม่ต้องมีเครื่องหมายปีกกาเปิดปิดก็ได้ (ถ้ากลัวสับสนก็ให้ใส่เครื่องหมายปีกกาทุกครั้งเมื่อเขียนคำสั่ง `if`)

พิจารณาเงื่อนไขของคำสั่ง `if` ว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจะเข้าทำคำสั่งภายใน `if` ทุกคำสั่ง ถ้าเป็นเท็จจะไม่เข้าทำ

ตัวอย่าง

ถ้า `score = 120` จะแสดงผลลัพธ์ได้เป็น ABC
 ถ้า `score = 100` จะแสดงผลลัพธ์ได้เป็น ABC
 ถ้า `score = 60` จะแสดงผลลัพธ์ได้เป็น BC



```
1  if(score >= 100) {
2      System.out.print("A");
3  }
4  System.out.print("B");
5  System.out.print("C");
```

2. คำสั่งตัดสินใจแบบ 2 เงื่อนไข

คำสั่งตัดสินใจแบบ 2 เงื่อนไขนี้จะใช้คำสั่ง `if-else` ในการทำงาน โดยมีรูปแบบดังนี้

```

1  if (Condition) {
2      Statement1;
3  } else {
4      Statement2;
5  }

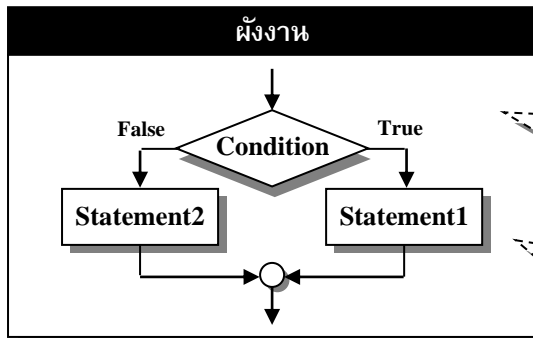
```

ถึงแม้ว่าจะมี 2 เงื่อนไข ก็ตามแต่ให้ระบุเงื่อนไขไว้ที่คำสั่ง if เท่านั้น โดยไม่ต้องระบุเงื่อนไขที่คำสั่ง else (หลังคำสั่ง else ห้ามใส่เงื่อนไข (Condition) โดยเด็ดขาด)

ภายในคำสั่ง if และ else สามารถมีคำสั่งได้มากกว่า 1 คำสั่ง แต่ถ้ามีเพียง 1 คำสั่งไม่ต้องมีเครื่องหมายปีกกาเปิดปิดก็ได้

คำสั่ง if-else ไม่สามารถเข้าทำคำสั่งภายในพร้อมกันได้ โดยจะเข้าทำเพียงกรณี if หรือ else กรณีใดกรณีหนึ่งเท่านั้น

พิจารณาเงื่อนไขของคำสั่ง if ว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจะเข้าทำคำสั่งภายใน if ทุกคำสั่ง แต่ถ้าเป็นเท็จจะเข้าทำคำสั่งภายใน else ทุกคำสั่ง



```

1  if(score >= 100) {
2      System.out.print("A");
3  } else {
4      System.out.print("B");
5  }
6  System.out.print("C");

```

ตัวอย่าง

ถ้า score = 120 จะแสดงผล AC

ถ้า score = 100 จะแสดงผล AC

ถ้า score = 60 จะแสดงผล BC

3. คำสั่งตัดสินใจแบบมากกว่า 2 เงื่อนไข (หลายเงื่อนไข)

คำสั่งตัดสินใจแบบมากกว่า 2 เงื่อนไขนี้จะใช้คำสั่ง if-else ซ้อน if-else ในการทำงาน โดยมีรูปแบบดังนี้

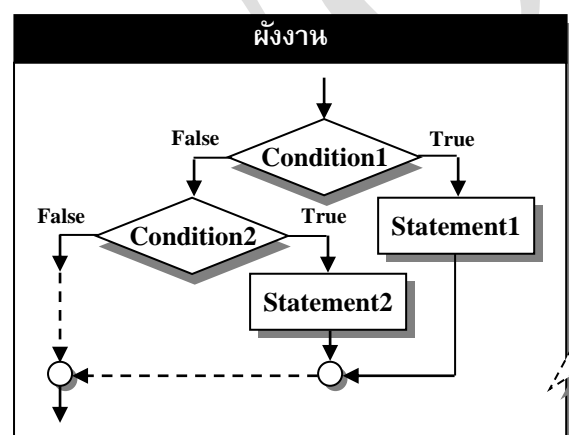
```

1  if (Condition1) {
2      Statement1;
3  } else if (Condition2) {
4      Statement2;
5  } else if (Condition3) {
6      ...
7  } else {
8      StatementN;
9  }

```

จะระบุเงื่อนไข ที่คำสั่ง if หรือ else if เท่านั้น แต่จะไม่ระบุเงื่อนไขที่คำสั่ง else โดยเด็ดขาด และสามารถตัดคำสั่ง else ออกไปได้

ถ้ามีหลายเงื่อนไขที่เป็นจริงจะเข้าทำเพียงกรณีแรกที่พบว่าเป็นจริงเท่านั้น (กรณีที่เหลือจะไม่เข้าทำอีกถึงแม้ว่าจะเป็นจริง)



พิจารณาเงื่อนไขของคำสั่ง if ว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจะเข้าทำคำสั่งภายในคำสั่ง if ทุกๆ คำสั่ง ถ้าเป็นเท็จจะพิจารณาเงื่อนไขของคำสั่ง else if ว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจะเข้าทำคำสั่งภายในคำสั่ง else if ทุกๆ คำสั่ง ซึ่งจะทำแบบนี้ไปเรื่อยๆ จนถึงคำสั่งสุดท้าย ถ้าไม่มีเงื่อนไขใดเลยเป็นจริงจะเข้าทำภายในคำสั่ง else ทุกๆ คำสั่ง (เลือกทำเพียงกรณีเดียวเท่านั้น)

หมายเหตุ

- 1) เงื่อนไขหรือนิพจน์ตรรกะจะเป็นค่าความจริง (Boolean) ที่มีค่าเป็น true หรือ false อย่างใดอย่างหนึ่งเท่านั้น
- 2) ตัวดำเนินการที่ใช้คือตัวดำเนินการเปรียบเทียบค่า (Comparison) และตัวดำเนินการตรรกศาสตร์ (Logical) เท่านั้น

โจทย์ข้อที่ 11 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาโดยให้เลือกใช้คำสั่ง if, if-else หรือ if-else if-else ตามเงื่อนไขที่กำหนดให้ต่อไปนี้ และให้รับค่าตัวแปรที่ใช้ในเงื่อนไขแต่ละข้อจากแป้นพิมพ์ (50 คะแนน)

- 1) ถ้าตัวแปร score มีค่าน้อยกว่า 50 ให้แสดงคำว่า "F" ขึ้นบนจอภาพ (5 คะแนน)
- 2) ถ้าตัวแปร i มีค่าน้อยกว่าหรือเท่ากับ 10 ให้เพิ่มค่าตัวแปร i นั้นขึ้นอีก 1 พร้อมทั้งแสดงค่าของตัวแปร i นั้นขึ้นบนจอภาพ (5 คะแนน)
- 3) ถ้าตัวแปร x เป็นตัวเลขบวก (Positive Number) ให้แสดงรากที่สองของตัวแปร x ขึ้นบนจอภาพ (5 คะแนน)
- 4) ถ้าตัวแปร num เป็นเลขคู่ (Even Number) ให้แสดงข้อความว่า "Even Number" ขึ้นบนจอภาพ (5 คะแนน)
- 5) ถ้าตัวแปร gpa มีค่ามากกว่าหรือเท่ากับ 2.00 ให้แสดงข้อความว่า "Pass" ขึ้นบนจอภาพ ถ้าไม่เช่นนั้นให้แสดงข้อความว่า "Retire" ขึ้นบนจอภาพ (5 คะแนน)
- 6) ถ้าตัวแปร score ว่ามีค่ามากกว่าหรือเท่ากับ 60 ให้แสดงเกรด "S" ขึ้นบนจอภาพ ถ้าไม่เช่นนั้นให้แสดงเกรด "C" ขึ้นบนจอภาพ (5 คะแนน)
- 7) ถ้าตัวแปร n เป็นเลขคี่ให้แสดงค่าตัวเลขนั้นตามด้วยข้อความว่า "is odd number" ขึ้นบนจอภาพ ถ้าเป็นเลขคู่ให้แสดงค่าตัวเลขนั้นตามด้วยข้อความว่า "is even number" ขึ้นบนจอภาพ (5 คะแนน)
- 8) ถ้าตัวแปร x มีค่ามากกว่าหรือเท่ากับ 0 ให้แสดงผลค่ารากที่สองของตัวแปร x ขึ้นบนจอภาพ ถ้าไม่เช่นนั้นให้แสดงค่าของตัวเลขนั้นตามด้วยข้อความว่า "is negative value" ขึ้นบนจอภาพ (5 คะแนน)
- 9) ถ้าตัวแปร num เป็นจำนวนเต็มบวกให้แสดงข้อความว่า "Positive Number" ขึ้นบนจอภาพ ถ้าเป็นจำนวนเต็มลบให้แสดงข้อความว่า "Negative Number" ขึ้นบนจอภาพ ถ้าเป็นจำนวนเต็มศูนย์ให้แสดงข้อความว่า "Zero Number" ขึ้นบนจอภาพ (5 คะแนน)
- 10) ถ้าตัวแปร n เท่ากับ 1 ให้แสดงข้อความ "High" ถ้าเท่ากับ 2 ให้แสดงข้อความ "Medium" ถ้าเท่ากับ 3 ให้แสดงข้อความ "Low" ถ้าไม่เช่นนั้นให้แสดงข้อความ "Unknown" (5 คะแนน)

โจทย์ข้อที่ 12 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับค่าตัวเลขจำนวนเต็ม 2 ค่าเข้ามาทางแป้นพิมพ์เก็บไว้ในตัวแปร x และ y แล้วหาค่าผลต่าง (Difference) ของตัวแปรทั้งสองตัวนั้น พร้อมทั้งแสดงผลลัพธ์ขึ้นบนจอภาพให้สวยงาม โดยห้ามใช้เมทอด abs (...) ในการประมวลผลครั้งนี้ (10 คะแนน)

```
Enter Number x: 43
Enter Number y: 76
Difference: 33
```

โจทย์ข้อที่ 13 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับชื่อเล่น (Nickname) จากแป้นพิมพ์และรับตัวเลขจำนวนเต็มอีก 1 จำนวนจากแป้นพิมพ์ แล้วให้พิจารณาว่าเลขจำนวนเต็มที่ได้รับจากแป้นพิมพ์ดังกล่าวนั้นมีค่าเป็น 1-4 หรือไม่ ถ้ามีค่าเป็น 1 ให้แสดงข้อความว่า “Good Morning” แล้วตามด้วยชื่อเล่นขึ้นบรรทัดใหม่ ถ้ามีค่าเป็น 2 ให้แสดงข้อความว่า “Good Afternoon” แล้วตามด้วยชื่อเล่นขึ้นบรรทัดใหม่ ถ้ามีค่าเป็น 3 ให้แสดงข้อความว่า “Good Evening” แล้วตามด้วยชื่อเล่นขึ้นบรรทัดใหม่ ถ้ามีค่าเป็น 4 ให้แสดงข้อความว่า “Good Night” แล้วตามด้วยชื่อเล่นขึ้นบรรทัดใหม่ แต่ถ้าค่าไม่ได้อยู่ในช่วง 1-4 ให้แสดงข้อความว่า “Hello” แล้วตามด้วยชื่อเล่นขึ้นบรรทัดใหม่ (10 คะแนน)

```
Enter Nickname: Bank
Enter Number: 2
Good Afternoon, Bank

Enter Nickname: Petch
Enter Number: -1
Hello, Petch
```

โจทย์ข้อที่ 14 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับค่าตัวแปร score จากทางแป้นพิมพ์ แล้วพิจารณาค่าในตัวแปร score ซึ่งใช้ในการแสดงผลการเรียนดังรายละเอียดต่อไปนี้ (10 คะแนน)

- ถ้า score มากกว่า 100 ให้แสดงข้อความว่า “Overflow Error”
- ถ้า score อยู่ในช่วง [80, 100] ให้แสดงผลการเรียนเป็น “A”
- ถ้า score อยู่ในช่วง [70, 80) ให้แสดงผลการเรียนเป็น “B”
- ถ้า score อยู่ในช่วง [60, 70) ให้แสดงผลการเรียนเป็น “C”
- ถ้า score อยู่ในช่วง [50, 60) ให้แสดงผลการเรียนเป็น “D”
- ถ้า score อยู่ในช่วง [0, 50) ให้แสดงผลการเรียนเป็น “F”
- ถ้า score น้อยกว่า 0 ให้แสดงข้อความว่า “Underflow Error”

โจทย์ข้อที่ 15 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับวันเดือนปีเกิดของนิสิตคณะวิศวกรรมศาสตร์เข้ามาทางแป้นพิมพ์หนึ่งคน เพื่อใช้คำนวณและแสดงค่าระดับความเกเรียน (Noob Levels) ที่ประกอบไปด้วยระดับที่ 1 เกเรียน (Noob) ระดับที่ 2 เกเรียนเรียกพ่อ (Father Noob) ระดับที่ 3 เกเรียนเทพ (God Noob) และระดับที่ 4 เกเรียนเมพ (Super God Noob) โดยให้รับข้อมูลทั้งสามค่าอยู่ภายในบรรทัดเดียวกัน และแต่ละค่าแบ่งด้วยช่องว่างอย่างน้อยหนึ่งช่องว่าง เช่น "1 12 2537" หรือ "04 08 2542" เป็นต้นและให้ใช้สูตรในการคำนวณหาค่าระดับความเกเรียนตามที่กำหนดให้ดังต่อไปนี้ (10 คะแนน)

$$\text{ระดับความเกเรียน} = ((\sqrt{\text{day}} + \sqrt{\text{month}} + \sqrt{\text{year}}) \% 4) + 1$$

โจทย์ข้อที่ 16 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับค่าตัวเลขจำนวนเต็มสองค่าเข้ามาทางแป้นพิมพ์เก็บไว้ในตัวแปร x และ y ตามลำดับ แล้วให้คำนวณหาคำตอบของผลหารระหว่าง x (ตัวตั้ง) หารด้วย y (ตัวหาร) โดยถ้าค่าของตัวตั้งและตัวหารเป็นศูนย์ทั้งคู่ให้แสดงผลลัพธ์ว่า “nan” ถ้าค่าของตัวตั้งมากกว่าศูนย์และตัวหารเป็นศูนย์ให้แสดงผลลัพธ์ว่า “Infinity” ถ้าค่าของตัวตั้งน้อยกว่าศูนย์และตัวหารเป็นศูนย์ให้แสดงผลลัพธ์ว่า “-Infinity” ถ้าค่าของทั้งสองเป็นค่าอื่น ๆ ให้แสดงผลหาร (Quotient) และเศษที่ได้จากการหาร (Remainder) ดังแสดงตามตัวอย่างที่กำหนดให้ (10 คะแนน)

Enter x: -92	Enter x: 9
Enter y: 0	Enter y: 5
-Infinity	Quotient: 1
	Remainder: 4

โจทย์ข้อที่ 17 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มสองค่าเข้ามาทางแป้นพิมพ์เก็บไว้ในตัวแปร x และ y ตามลำดับ แล้วตรวจสอบว่า x มากกว่า y หรือไม่ ถ้าใช่ให้สลับค่า x และ y นั้น แต่ถ้าไม่เช่นนั้นให้เพิ่มค่า x และ y ขึ้นอีก 1 พร้อมทั้งแสดงค่าตัวแปรทั้งสองขึ้นบนจอภาพ (หมายเหตุ ค่า x และ y จะสลับค่ากันจริง ๆ ไม่ใช่เพียงแค่การแสดงผลแบบสลับชื่อตัวแปร) (10 คะแนน)

โจทย์ข้อที่ 18 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนจริงจากทางแป้นพิมพ์สองตัวเข้ามาเก็บไว้ในตัวแปร a และ b ตามลำดับ แล้วทำการแสดงผลลัพธ์ว่าจำนวนใดที่มีค่ามากกว่าหรือน้อยกว่าหรือเท่ากันนอกจอภาพ พร้อมทั้งแสดงค่าของตัวเลขจำนวนเต็มที่อยู่กึ่งกลางระหว่างตัวเลขทั้งสองขึ้นบนจอภาพอีกด้วย โดยห้ามใช้เมทอด $\max(\dots)$, $\min(\dots)$ ในการประมวลผล (10 คะแนน)

Enter a: 13.0
Enter b: 22.0
13.0 < 22.0 (Mid:17)
Enter a: 8.0
Enter b: 8.0
8.0 = 8.0 (Mid:8)

โจทย์ข้อที่ 19 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อใช้ในการคำนวณค่าความต้านทานรวมของวงจรไฟฟ้าที่ประกอบไปด้วยตัวต้าน 3 ตัว ซึ่งต่อแบบอนุกรมหรือต่อแบบขนานอย่างใดอย่างหนึ่ง โดยจะต้องรับค่าของตัวต้านทานทั้ง 3 ตัว ที่เป็นจำนวนเต็มเข้ามาทางแป้นพิมพ์ทีละค่าเก็บไว้ในตัวแปร $r1$, $r2$ และ $r3$ ตามลำดับ พร้อมทั้งรับจำนวนเต็มอีกหนึ่งค่าเก็บไว้ในตัวแปร $types$ เพื่อใช้แทนรูปแบบการต่อวงจร โดยถ้า $types$ เป็น 1 โปรแกรมจะคำนวณค่าความต้านทานรวมแบบอนุกรม ถ้า $types$ เป็น 2 โปรแกรมจะคำนวณค่าความต้านทานรวมแบบขนาน ถ้า $types$ เป็นค่าอื่น ๆ ให้แสดงข้อความว่า

Enter R1: 4
Enter R2: 2
Enter R3: 2
Enter Type (1 or 2): 2
R Total is 0.8
Enter R1: 11
Enter R2: 9
Enter R3: 5
Enter Type (1 or 2): 5
Incorrect Circuit Type

"Incorrect Circuit Types" โดยให้ใช้สูตรที่กำหนดให้ในการคำนวณ และให้ใช้ตัวแปร rTotal เก็บผลลัพธ์ของความต้านทานรวมที่คำนวณได้ พร้อมทั้งแสดงผลพร้อมออกทางจอภาพดังตัวอย่างต่อไปนี้ (10 คะแนน)

$$\text{แบบอนุกรม } R_{\text{total}} = R_1 + R_2 + R_3$$

$$\text{แบบขนาน } \frac{1}{R_{\text{total}}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}$$

(ห้ามประกาศตัวแปรเพิ่มเติมนอกเหนือจากที่โจทย์กำหนดไว้โดยเด็ดขาด และจะต้องแสดงผลที่ได้ตามตัวอย่างที่โจทย์กำหนดมา)

โจทย์ข้อที่ 20 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มจากทางแป้นพิมพ์ 4 จำนวน โดยให้รับค่าอยู่ภายในบรรทัดเดียวกันคั่นด้วยเว้นวรรคดังตัวอย่าง จากนั้นให้เรียงลำดับ (Sorting) ตัวเลขทั้ง 4 จำนวนนั้นจากค่าน้อยสุดไปหาค่ามากที่สุด พร้อมทั้งแสดงผลที่ได้ขึ้นบนจอภาพให้สวยงามดังตัวอย่างต่อไปนี้ (10 คะแนน)

Enter 4 Number: 12 9 1 10

Sorted Number: 1 9 10 12

Enter 4 Number: 87 34 136 25

Sorted Number: 25 34 87 136

โจทย์ข้อที่ 21 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับหมายเลขเดือนและปี ค.ศ. เข้ามาทางแป้นพิมพ์ โดยให้คำนวณและคืนค่าจำนวนวันในเดือนและปี ค.ศ. ที่ระบุ ดังตัวอย่างต่อไปนี้

- ถ้าระบุหมายเลขเดือนเป็น 12 ปี ค.ศ. เป็น 2015 จะได้จำนวนวันเป็น 31
- ถ้าระบุหมายเลขเดือนเป็น 2 ปี ค.ศ. เป็น 2016 จะได้จำนวนวันเป็น 29
- ถ้าระบุหมายเลขเดือนเป็น 2 ปี ค.ศ. เป็น 2017 จะได้จำนวนวันเป็น 28
- ถ้าหมายเลขเดือนไม่ถูกต้องให้แสดงข้อความ "Incorrect Month"

Enter Month: 2

Enter Year: 2016

Number of Day: 29

Enter Month: 13

Enter Year: 2017

Incorrect Month

โดยในการคำนวณจะต้องตรวจสอบปีอธิกสุรทิน (Leap Year) ซึ่งเป็นปีที่เดือนกุมภาพันธ์มี 29 วัน โดยมีวิธีการตรวจสอบคือ ปี ค.ศ. ใดที่หาร 4 ลงตัวแต่หาร 100 ไม่ลงตัวจะเป็นปีอธิกสุรทินหรือปี ค.ศ. ใดที่หาร 400 ลงตัวก็จะเป็นปีอธิกสุรทินเช่นกัน (10 คะแนน)

โจทย์ข้อที่ 22 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับหมายเลขวัน เดือน ปี ค.ศ. เข้ามาทางแป้นพิมพ์ แล้วให้คำนวณว่าวันเดือนปีดังกล่าวนั้นเป็นวันอะไรในสัปดาห์ (จันทร์ อังคาร พุธ ฯลฯ) พร้อมทั้งแสดงผลขึ้นบนจอภาพให้สวยงาม โดยกำหนดให้ใช้สูตรดังต่อไปนี้ในการคำนวณ (10 คะแนน)

ชื่อตัวแปร	ความหมาย	สูตร / เงื่อนไข
d	วัน (Day)	ใช้ตามค่า d ที่รับมา
m	เดือน (Month)	$m = \begin{cases} m + 12, m < 3 \\ m, m \geq 3 \end{cases}$
y	ปี (Year)	$y = \begin{cases} y - 1, m < 3 \\ y, m \geq 3 \end{cases}$
c	ค่าคงที่	$c = \left\lfloor \frac{y}{100} \right\rfloor$
k	ค่าคงที่	$k = y \% 100$
w	หมายเลขวันในหนึ่งสัปดาห์	$w = \left(d + \left\lfloor \frac{26(m+1)}{10} \right\rfloor + 5c + \left\lfloor \frac{c}{4} \right\rfloor + k + \left\lfloor \frac{k}{4} \right\rfloor \right) \% 7$

โดยค่าของตัวแปร w ที่คำนวณได้จะมีค่าอยู่ในช่วง 0 ถึง 6 โดยกำหนดให้ 0 = "SAT", 1 = "SUN", 2 = "MON", 3 = "TUE", 4 = "WED", 5 = "THU" และ 6 = "FRI" ตามลำดับ

Enter Day: 2
Enter Month: 9
Enter Year: 2015
WED

4

การวนซ้ำ (Iterations)

1. คำสั่งวนซ้ำด้วยคำสั่ง while มี 3 ส่วนหลักดังรูปแบบคำสั่งต่อไปนี้

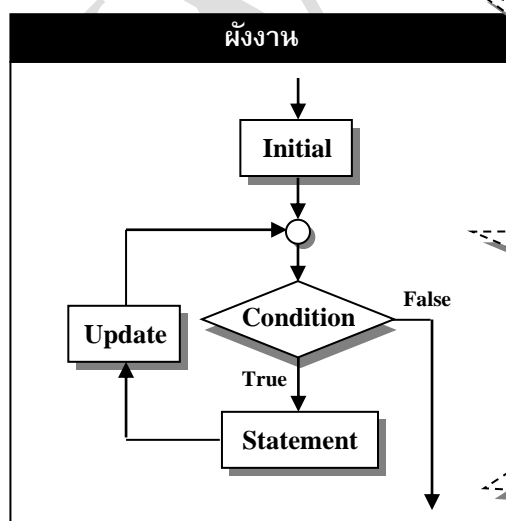
```

1 Initial;
2 while(Condition) {
3     Statement(s);
4     Update;
5 }
```

① คำตั้งต้น (Initial) เป็นการกำหนดค่าเริ่มต้นสำหรับการวนซ้ำ

② เงื่อนไข (Condition) จะเป็นนิพจน์ตรรกะ (Boolean Expression) ที่แสดงถึงค่าความจริงว่าเมื่อไรจะวนซ้ำ เมื่อไรจะสิ้นสุดการวนซ้ำ

③ การเปลี่ยนรอบ (Update) เป็นการเปลี่ยนรอบการวนซ้ำ (ถ้าไม่ระบุส่วนนี้จะเกิดการวนซ้ำแบบไม่รู้จบ หรือ Infinite Loop)



ภายในคำสั่ง while ถ้ามีเพียง 1 คำสั่งไม่ต้องมีเครื่องหมายปีกกาเปิดปิดก็ได้

พิจารณาเงื่อนไขของคำสั่ง while ว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจะเข้าทำคำสั่งทุกๆ คำสั่ง ภายในคำสั่ง while และจะวนทำงานแบบนี้ไปเรื่อยๆ จนกว่าเงื่อนไขของคำสั่ง while จะเป็นเท็จ ก็จะสิ้นสุดการวนซ้ำ


```

1 int i = 1;
2 while (i <= 5) {
3     System.out.print(i);
4     i++;
5 }

```

กำหนดค่าเริ่มต้นให้ตัวแปร `i` เท่ากับ 1 แล้วทำการวนซ้ำเพื่อ แสดงค่า `i` ตั้งแต่ 1 ถึง 5 โดยในแต่ละรอบการวนซ้ำจะเพิ่มจำนวนรอบ (เพิ่มค่า `i`) ขึ้นทีละ 1 (โดยใช้คำสั่ง `i++`) ดังนั้นผลลัพธ์ที่ได้คือ 12345

2. คำสั่งวนซ้ำด้วยคำสั่ง `do-while` มี 3 ส่วนหลักดังรูปแบบคำสั่งต่อไปนี้

```

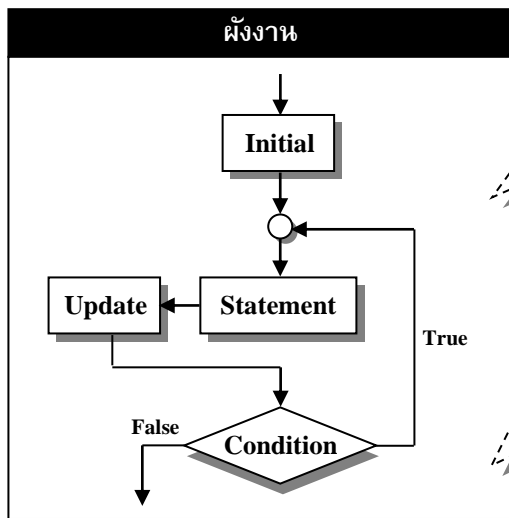
1 Initial;
2 do {
3     Statement(s);
4     Update;
5 } while (Condition);

```

หลังคำสั่ง `do` ไม่มีเงื่อนไข (Condition) แต่หลังคำสั่ง `while` จะมีเงื่อนไข

หลังจบคำสั่ง `while` ต้องมีเครื่องหมาย Semicolon ปิดท้ายเสมอ

ภายในคำสั่ง `do` ถ้ามีเพียง 1 คำสั่งไม่ต้องมีเครื่องหมายปีกกาเปิดปิดก็ได้



เข้าทำคำสั่งทุกๆ คำสั่ง ภายในคำสั่ง `do` หนึ่งรอบก่อนเสมอ แล้วจึงพิจารณาเงื่อนไขของคำสั่ง `while` ว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจะวนทำงานแบบนี้ไปเรื่อยๆ จนกว่าเงื่อนไขของคำสั่ง `while` จะเป็นเท็จ ก็จะสิ้นสุดการวนซ้ำ

เปรียบเทียบคำสั่ง `while` และคำสั่ง `do-while`

คำสั่ง `while` จะตรวจสอบเงื่อนไขก่อนการวนซ้ำ (ตรวจสอบเงื่อนไขก่อนแล้วค่อยทำงาน)

คำสั่ง `do-while` จะตรวจสอบเงื่อนไขหลังการวนซ้ำ (ทำงานก่อนแล้วค่อยตรวจสอบเงื่อนไข)

```

1 int i = 1;
2 do {
3     System.out.print(i);
4     i++;
5 } while(i <= 5);

```

กำหนดค่าเริ่มต้นให้ตัวแปร `i` เท่ากับ 1 แล้วเข้าทำในคำสั่ง `do` เพื่อแสดงค่า `i` ในรอบที่หนึ่งแล้วเพิ่มค่า `i` ขึ้นอีก 1 จากนั้นจึงค่อยตรวจสอบเงื่อนไขในคำสั่ง `while` แล้ววนซ้ำแสดงค่า `i` ตั้งแต่ 1 ถึง 5 (แต่ละรอบจะมีการเพิ่มค่า `i` ขึ้นทีละ 1) ดังนั้นผลลัพธ์ที่ได้คือ 12345

3. คำสั่งวนซ้ำด้วยคำสั่ง `for` มี 3 ส่วนหลักเช่นเดียวกับคำสั่ง `while` ดังรูปแบบต่อไปนี้

```

1 for (Initial; Condition; Update) {
2     Statement(s);
3 }

```

[1] ค่าตั้งต้น (Initial) [2] เงื่อนไข (Condition) และ [3] การเปลี่ยนรอบ (Update) ทั้ง 3 ส่วนนี้จะอยู่ภายในส่วนหัวของคำสั่ง `for` ทั้งหมด

```

1 for (int i = 1; i <= 5; i++) {
2     System.out.print(i);
3 }

```

4. คำสั่งวนซ้ำด้วยคำสั่ง `while (true)` มีรูปแบบดังต่อไปนี้

```

1 while (true) {
2     Statement(s);
3     if (Condition) break;
4 }

```

คำสั่ง `while (true)` มีเงื่อนไข (Condition) เป็นจริงตลอดกาล ดังนั้นจะวนซ้ำแบบไม่รู้จบ การออกจากคำสั่ง `while (true)` ทำได้เพียงวิธีเดียวเท่านั้น คือ ใช้คำสั่ง `if (...) break;`

```

1 Scanner kb = new ...;
2 while (true) {
3     int x = kb.nextInt();
4     if (x == 0) break;
5     x = x + 1;
6     if (x == 2) continue;
7     x = x + 1;
8 }

```

- คำสั่ง **break** ใช้ในการออกจาก Loop
- คำสั่ง **continue** ใช้ในการวกกลับไปเริ่มต้น Loop

โจทย์ข้อที่ 23 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาโดยใช้คำสั่ง **while**, **do-while**, **for** และ **while-true** ตามเงื่อนไขที่กำหนดให้ดังต่อไปนี้ (60 คะแนน)

- 1) แสดงคำว่า "Java" ขึ้นบนจอภาพ 100 ครั้ง (20 คะแนน)
- 2) แสดงตัวเลข 100 ถึง 1 (นับถอยหลัง) ออกทางจอภาพ (20 คะแนน)
- 3) แสดงเลขคู่ 100 ตัวแรกออกทางจอภาพ โดยเริ่มต้นที่ตัวเลข 78 เป็นต้นไป (20 คะแนน)

โจทย์ข้อที่ 24 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อคำนวณหาผลบวกของตัวเลขตั้งแต่ 1 จนถึงค่า n ที่รับเข้ามาผ่านทางแป้นพิมพ์ แล้วเก็บคำตอบไว้ในตัวแปร sm พร้อมทั้งแสดงค่าของตัวแปร sm ออกทางจอภาพ (10 คะแนน)

โจทย์ข้อที่ 25 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อคำนวณหาค่าแฟคทอเรียล (Factorial) ของตัวแปร n ที่รับเข้ามาผ่านทางแป้นพิมพ์ โดยที่ $n! = 1 \times 2 \times \dots \times n$ แล้วเก็บคำตอบไว้ในตัวแปร fc พร้อมทั้งแสดงค่าของตัวแปร fc ออกทางจอภาพ (10 คะแนน)

โจทย์ข้อที่ 26 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อคำนวณหาค่าเลขยกกำลังของ a^b โดยที่ a เป็นเลขฐานและ b เป็นเลขชี้กำลัง และตัวแปร a และ b เป็นค่าที่รับเข้ามาผ่านทางแป้นพิมพ์ โดยให้เก็บคำตอบไว้ในตัวแปร pw พร้อมทั้งแสดงค่าของตัวแปร pw ออกทางจอภาพ กำหนดให้ b มีค่าตั้งแต่ 0 ขึ้นไป และห้ามใช้เมธอด `pow(...)` ในการคำนวณ (10 คะแนน)

หมายเหตุ นิยามของ a^b คือ a คูณกัน b ตัว

โจทย์ข้อที่ 27 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อคำนวณหาค่าผลคูณของ $a \times b$ โดยที่ตัวแปร a และ b เป็นค่าที่รับเข้ามาผ่านทางแป้นพิมพ์ และให้เก็บคำตอบไว้ในตัวแปร $m1$ พร้อมทั้งแสดงค่าของตัวแปร $m1$ ออกทางจอภาพ โดยห้ามใช้เครื่องหมายคูณในการคำนวณ (10 คะแนน)

หมายเหตุ นิยามของ $a \times b$ คือ a บวกกัน b ตัว

โจทย์ข้อที่ 28 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อหาค่าเฉลี่ยของตัวเลขจำนวนเต็ม ที่ได้จากการสุ่ม (Random) ทั้งสิ้น n จำนวน (รับ n จากทางแป้นพิมพ์) ซึ่งตัวเลขแต่ละตัวมีค่าอยู่ในช่วง 1 ถึง 100 พร้อมทั้งแสดงค่าเฉลี่ยที่คำนวณได้ออกทางจอภาพให้สวยงาม (10 คะแนน)

โจทย์ข้อที่ 29 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มเข้ามาทางแป้นพิมพ์ 50 ตัว แล้วนับจำนวนของเลขคู่และเลขคี่ว่ามีอย่างละกี่จำนวนในตัวเลขทั้ง 50 ตัวนั้น พร้อมทั้งแสดงผลการนับที่ได้ออกทางจอภาพให้สวยงาม (10 คะแนน)

โจทย์ข้อที่ 30 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มหนึ่งจำนวนเข้ามาเก็บไว้ในตัวแปร n แล้วให้แสดงตัวเลขจำนวนเต็มทุกตัวระหว่าง 1 ถึง n ที่หาร n นั้นลงตัวขึ้นบนจอภาพ (ไม่รวม 1 และ n) พร้อมทั้งแสดงด้วยว่าตัวเลขจำนวนเต็มที่มากที่สุดที่หาร n นั้นลงตัวคือค่าใด (10 คะแนน)

โจทย์ข้อที่ 31 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มใด ๆ 100 จำนวนทางแป้นพิมพ์ แล้วหาค่าสูงสุด (Maximum) และค่าต่ำสุด (Minimum) ของตัวเลขทั้งหมดนั้น พร้อมทั้งแสดงผลลัพธ์ที่ได้ออกทางจอภาพ (ไม่อนุญาตให้ใช้ฟังก์ชันใด ๆ ในการประมวลผล) (10 คะแนน)

โจทย์ข้อที่ 32 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มหนึ่งจำนวนทางแป้นพิมพ์ แล้วตรวจสอบว่าตัวเลขดังกล่าวเป็นจำนวนเฉพาะ (Prime Number) หรือไม่ พร้อมทั้งแสดงผลลัพธ์ที่ได้ดังตัวอย่าง ซึ่งจำนวนเฉพาะคือจำนวนที่ค่า 1 และตัวมันเองเท่านั้นที่หารลงตัว (10 คะแนน)
(ดูตัวอย่างการแสดงผลในหน้าถัดไป)

Number: 17
17 is a prime number

Number: 45
45 is not a prime number

โจทย์ข้อที่ 33 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็มเก็บไว้ในตัวแปร n และหาคำตอบของสมการฟีโบแนคชี (Fibonacci) ซึ่งมีรูปแบบของสมการคือ $F_n = F_{n-1} + F_{n-2}$ โดยกำหนดให้ค่าของ $F_0 = 0$ และค่าของ $F_1 = 1$ ส่วนรายละเอียดอื่นๆ แสดงไว้ตามตัวอย่าง โดยให้แสดงผลลัพธ์ที่ได้ออกทางจอภาพให้สวยงาม (10 คะแนน)

$F_2 = F_1 + F_0 \rightarrow 1 + 0 = 1$
 $F_3 = F_2 + F_1 \rightarrow 1 + 1 = 2$
 $F_4 = F_3 + F_2 \rightarrow 2 + 1 = 3$
 $F_5 = F_4 + F_3 \rightarrow 3 + 2 = 5$
 ...

โจทย์ข้อที่ 34 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนเต็ม 2 จำนวนเข้ามาทางแป้นพิมพ์ แล้วหาค่า ห.ร.ม. (GCD) ของสองจำนวนนั้น พร้อมทั้งแสดงผลลัพธ์ขึ้นบนจอภาพ (10 คะแนน)

การหาค่า ห.ร.ม. ของ a และ b
ทำได้โดยการตรวจสอบว่า a % b เท่ากับ 0 หรือไม่ ถ้าใช่ค่า GCD ก็คือ b แต่ถ้าไม่ใช่ให้สลับค่า a ให้เท่ากับ b และค่า b ให้เท่ากับ a % b แล้ววนตรวจสอบในรอบถัดไปเรื่อย ๆ

โจทย์ข้อที่ 35 [ระดับง่าย-ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อแสดงรูปภาพเรขาคณิตดังต่อไปนี้และกำหนดให้ใช้คำสั่ง 4 คำสั่งที่กำหนดให้เท่านั้นในการแสดงผล ได้แก่ `System.out.print("*")` ; , `System.out.println("*")` ; , `System.out.print(" ")` ; และ `System.out.println()` ; (70 คะแนน)

ข้อ	รูปภาพตัวอย่าง	ข้อ	รูปภาพตัวอย่าง	ข้อ	รูปภาพตัวอย่าง	ข้อ	รูปภาพตัวอย่าง
1.	Enter n: 11 ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** *****	2.	Enter n: 11 * ** *** **** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** *****	3.	Enter n: 11 ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** *****	4.	Enter n: 11 * ** *** **** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** *****
5.	Enter n: 11 ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** *****	6.	Enter n: 11 ***** * * * * * * * * * * * * * * *****	7.	Enter n: 11 ***** ** * * * * * * * * * * * * * *****	8.	Enter n: 11 ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** ***** *****

5 สตริง (String)

1. นิยามและข้อมูลเบื้องต้นของสตริง

- 1) สตริง (String) หรือ สายอักขระ คือ ตัวอักขระ (Character) ตั้งแต่ศูนย์ตัวขึ้นไปมาเรียงต่อกันอยู่ภายใต้เครื่องหมาย Double Quote เช่น "Hello", "F", "@123", "คำถึง หลวงกระโทก", " ", "" เป็นต้น
- 2) เรียก "" (เครื่องหมาย Double Quote ติดกัน) ว่า สตริงว่าง (Empty String / Null String) ซึ่งเป็นสตริงที่มีความยาวเป็นศูนย์ (หรือมีจำนวนอักขระเป็นศูนย์)
- 3) การประกาศตัวแปรประเภทสตริงทำได้ดังนี้

```
String <ชื่อสตริง> = "<ค่าสตริง>";
```

เช่น String s = "Hello SSRU";

คำว่า String จะต้องเขียนขึ้นต้นด้วย
อักษร S แบบพิมพ์ใหญ่เสมอ

ค่าของสตริงต้องล้อมด้วยเครื่องหมาย "..." เสมอ

- 4) การนำเข้าสตริงจากแป้นพิมพ์ จะใช้เมทอดของคลาส Scanner ซึ่งได้กล่าวไปแล้วในบทที่ 1 โดยประกอบไปด้วยเมทอดดังต่อไปนี้
 - (1) nextLine() ใช้ในการรับสตริงทีละบรรทัด
 - (2) next() ใช้ในการรับสตริงทีละช่วงหรือทีละคำ (เริ่มนับคำแรกจากทางซ้ายมือ)
- 5) สตริงมีตัวดำเนินการพื้นฐานดังต่อไปนี้

รูปแบบ	ข้อมูลที่ 1	กลุ่มของตัวดำเนินการ	ข้อมูลที่ 2	ประเภทข้อมูลสุดท้าย
1.	String	=	String	String
2.	String	+	ทุกประเภท	String (การต่อสตริง)

รูปแบบ	ข้อมูลที่ 1	กลุ่มของตัวดำเนินการ	ข้อมูลที่ 2	ประเภทข้อมูลสุดท้าย
3.	ทุกประเภท	+	String	String (การต่อสตริง)
4.	String	== !=	String	boolean

2. โครงสร้างของสตริง

- 1) สตริงเป็น คลาสมาตรฐาน ในภาษาจาวา โดยโครงสร้างภายในเขียนด้วย อาร์เรย์ของตัวอักษร (Array of Character) ดังตัวอย่าง

(1) สตริงในรูปแบบทั่วไป `String s = "Java";`

(2) สตริงในรูปแบบอาร์เรย์ `char s[] = {'J', 'a', 'v', 'a'};`

หรืออาจจะเขียนสตริงในรูปแบบของอ็อบเจกต์ดังนี้
`String s = new String("Java");`

- 2) ตำแหน่ง (Index) ของสตริงจะ เริ่มนับที่ 0 แต่ ความยาว (Length) ของสตริงจะ เริ่มนับที่ 1 ดังตัวอย่าง

`String s = "JAVA SSRU!.";`

0	1	2	3	4	5	6	7	8	9	10
J	A	V	A		S	S	R	U	!	.

หมายเลขตำแหน่ง (Index)

สตริงยาว 11 อักขระ แต่ตำแหน่งสูงสุดคือ 10

3. การดำเนินการกับสตริง

เนื่องจากสตริงเป็น ตัวแปรที่ถูกสร้างขึ้นจากคลาส String ดังนั้นการดำเนินการส่วนใหญ่ของสตริงจะต้องกระทำภายใต้ เมทอดของสตริงหรือเมทอดที่เกี่ยวข้องกับสตริง โดยมีวิธีการเรียกใช้เมทอดดังนี้

`<ชื่อสตริง>.<ชื่อเมทอด>([พารามิเตอร์])`

เช่น `s.length()`, `n.charAt(4)` เป็นต้น

พารามิเตอร์ คือ ค่าที่อยู่ในวงเล็บของเมทอด

- 1) เมทอดหาความยาวของสตริง (String Length)

`<สตริง>.length()`

จะคืนค่า จำนวนเต็ม ซึ่งเป็นความยาวของสตริง

ตัวอย่างเช่น

```
String name = "Suda Rukchula";
int len = name.length(); //len = 13
```

- 2) เมทอดตัดช่องว่างทางซ้ายหรือขวาของสตริงออกไป (String Trim)

`<สตริง>.trim()`

จะคืนค่าสตริงที่เหมือนกับสตริงเดิมแต่ตัดช่องว่าง (Space) ทางซ้ายหรือขวาของสตริงออกไป

ตัวอย่างเช่น

```
String m = " Hello JAVA ";
String n = m.trim(); //n = "Hello JAVA"
```

- 3) เมทอดแปลงสตริงให้เป็นอักษรตัวพิมพ์ใหญ่ (String to Upper Case)

`<สตริง>.toUpperCase()`

จะคืนค่าสตริงที่เหมือนกับสตริงเดิมแต่ตัวอักษร อังกฤษทุกตัวจะเปลี่ยนเป็นตัวพิมพ์ใหญ่

ตัวอย่างเช่น

```
String m = "Hello! JaVa 2013";
m = m.toUpperCase(); //m = "HELLO! JAVA 2013"
```

4) เมทีออดแปลงสตริงให้เป็นอักษรตัวพิมพ์เล็ก (String to Lower Case)

```
<สตริง>.toLowerCase()
```

ตัวอย่างเช่น

```
String m = "$Hello JAVA #2556";
m = m.toLowerCase(); //m = $hello java #2556
```

คืนค่าสตริงที่เหมือนกับสตริงเดิมแต่ตัวอักษร
อังกฤษทุกตัวจะเปลี่ยนเป็นตัวพิมพ์เล็ก

5) เมทีออดตัดสตริงย่อย (Substring)

(1) การตัดสตริงย่อยตั้งแต่ตำแหน่งเริ่มต้นจนถึงก่อนตำแหน่งสิ้นสุด

```
<สตริง>.substring(<ตำแหน่งเริ่ม>, <ตำแหน่งจบ>)
```

ตัวอย่างเช่น

```
String name = "Chulalongkorn University";
String s1 = name.substring(9, 16); //s1 = "korn Un"
String s2 = name.substring(0, 5) + "Thailand"; //s2 = "ChulaThailand"
```

เอาหน้าแต่ไม่เอาหลัง

คืนค่าสตริงที่เป็นสตริงย่อยตั้งแต่ตำแหน่ง
เริ่มต้น จนถึงก่อนตำแหน่งสิ้นสุด

(2) การตัดสตริงย่อยตั้งแต่ตำแหน่งเริ่มต้นเป็นต้นไปจนจบสตริง

```
<สตริง>.substring(<ตำแหน่งเริ่ม>)
```

ตัวอย่างเช่น

```
String name = "Chulalongkorn University";
String s1 = name.substring(14); //s1 = "University"
String s2 = name.substring(20); //s2 = "sity"
```

คืนค่าสตริงที่เป็นสตริงย่อยตั้งแต่ตำแหน่ง
เริ่มต้นที่ระบุ จนถึงตำแหน่งสุดท้าย (จบสตริง)

6) เมทีออดหาตัวอักขระ ณ ตำแหน่งที่ระบุในสตริง (Character At)

```
<สตริง>.charAt(<ตำแหน่ง>)
```

ตัวอย่างเช่น

```
String m = "Hello JAVA";
char ch1 = m.charAt(0); //ch1 = 'H'
char ch2 = m.charAt(6); //ch2 = 'J'
```

คืนค่าอักขระในตำแหน่งที่ระบุ

7) เมทีออดหาตำแหน่งที่พบเป็นครั้งแรกของสตริงย่อยในสตริงหลัก (Index Of)

```
<สตริง>.indexOf(<สตริง>)
```

ตัวอย่างเช่น

```
String m = "Hello JAVA";
int i = m.indexOf("l"); //i = 2
int j = m.indexOf("lo"); //j = 3
int k = m.indexOf("F"); //k = -1
```

คืนค่าจำนวนเต็มที่เป็นตำแหน่งของสตริงย่อยใน
สตริงหลักที่พบเป็นครั้งแรก ถ้าไม่พบจะคืนค่า -1

8) เมทีออดหาตำแหน่งที่พบเป็นครั้งสุดท้ายของสตริงย่อยในสตริงหลัก (Last Index Of)

```
<สตริง>.lastIndexOf(<สตริง>)
```

ตัวอย่างเช่น

```
String m = "Hello JAVA";
int i = m.lastIndexOf("A"); //i = 9
int j = m.lastIndexOf("AV"); //j = 7
int k = m.lastIndexOf("b"); //k = -1
```

คืนค่าจำนวนเต็มที่เป็นตำแหน่งของสตริงย่อยใน
สตริงหลักที่พบเป็นครั้งสุดท้าย ถ้าไม่พบจะคืนค่า -1

เริ่มนับที่ตำแหน่งที่ 0 (ซ้ายไปขวา)

9) เมท็อดเปรียบเทียบการเท่ากันทุกประการของสตริง

`<สตริง>.equals(<สตริง>)`

คืนค่าความจริงที่เป็น true ถ้าสตริงเท่ากันทุกประการ หรือ false ถ้าสตริงไม่เท่ากันทุกประการ

ตัวอย่างเช่น

```
String s1 = "E-mail";
String s2 = "e-mail";
boolean flag = s1.equals(s2); //flag = false
```

เปรียบเทียบสตริง s1 กับสตริง s2

10) เมท็อดเปรียบเทียบการเท่ากันของสตริงโดยไม่สนใจตัวอักษรพิมพ์ใหญ่หรือพิมพ์เล็ก

`<สตริง>.equalsIgnoreCase(<สตริง>)`

คืนค่าความจริงที่เป็น true ถ้าสตริงเท่ากัน หรือ false ถ้าสตริงไม่เท่ากัน โดยไม่สนใจตัวอักษรพิมพ์ใหญ่พิมพ์เล็ก

ตัวอย่างเช่น

```
String s1 = "E-mail";
String s2 = "e-mail";
boolean flag = s1.equalsIgnoreCase(s2); //flag = true
```

11) เมท็อดเปรียบเทียบลำดับของสตริง (ใช้ค่า Unicode ในการเปรียบเทียบ)

`<สตริง>.compareTo(<สตริง>)`

คืนค่าที่น้อยกว่า 0 เมื่อสตริงตัวแรกมาก่อนสตริงตัวหลัง
คืนค่า 0 เมื่อสตริงตัวแรกและสตริงตัวหลังเท่ากัน
คืนค่าที่มากกว่า 0 เมื่อสตริงตัวแรกมาหลังสตริงตัวหลัง

ตัวอย่างเช่น

```
String s1 = "E-mail";
String s2 = "e-mail";
int flag = s1.compareTo(s2); //flag = -32
```

(เปรียบเทียบรหัสยูนีโค้ด (Unicode) ของอักขระทีละตัว โดยพิจารณาอักขระเป็นคู่ๆ เริ่มจากอักขระทางซ้ายไปขวาและจะคืนค่าความแตกต่างของรหัสยูนีโค้ดในอักขระคู่แรกที่มีความแตกต่างกัน)

12) การแปลงสตริงให้เป็นตัวเลขจำนวนเต็ม (String to Integer)

`Integer.parseInt(<สตริง>)`

คืนค่าจำนวนเต็มที่มีค่ามาจากสตริงที่เป็นตัวเลขจำนวนเต็ม (ห้ามมีอักขระอื่นๆ ที่ไม่ใช่ตัวเลขปะปน)

ตัวอย่างเช่น

```
String price = "500";
int p = Integer.parseInt(price); //p = 500
int len = p.length(); //Error
```

13) การแปลงสตริงให้เป็นตัวเลขจำนวนจริง (String to Double)

`Double.parseDouble(<สตริง>)`

คืนค่าจำนวนจริงที่มีค่ามาจากสตริงที่เป็นตัวเลขจำนวนจริงหรือจำนวนเต็ม

ตัวอย่างเช่น

```
String price = "500.0";
double p = Double.parseDouble(price); //p = 500.0
```

14) การแปลงตัวเลขจำนวนเต็มให้เป็นสตริง (Integer to String)

`Integer.toString(<จำนวนเต็ม>)`

คืนค่าสตริงที่มีค่าเหมือนกับตัวเลขจำนวนเต็มที่ระบุ (หรือใช้วิธี `<จำนวนเต็ม> + ""`)

ตัวอย่างเช่น

```
int price = 500;
String p = Integer.toString(price); //p = "500"
```


15) การแปลงตัวเลขจำนวนจริงให้เป็นสตริง (Double to String)

`Double.toString(<จำนวนจริง>)`

ตัวอย่างเช่น

```
double price = 500.0;
String p = Double.toString(price); //p = "500.0"
```

คืนค่าสตริงที่มีค่าเหมือนกับตัวเลขจำนวนจริงที่ระบุ
(หรือใช้วิธี `<จำนวนจริง> + ""`)

4. การใช้คำสั่งวนซ้ำเพื่อดำเนินการกับสตริง

เมื่อต้องการประมวลผลกับอักขระที่ปรากฏอยู่ในสตริงชุดใดชุดหนึ่งที่ละตัวสามารถทำได้โดยการใช้คำสั่งวนซ้ำ เช่น คำสั่ง `for` หรือคำสั่ง `while` วนเข้าไปยังอักขระที่ละตัวโดยมีวิธีการดังต่อไปนี้

```
1 String s = "ABCDEFGH";
2 for(int i = 0; i < s.length(); i++) {
3     String c = s.substring(i, i + 1);
4     System.out.println(c);
5 }
```

`i` ต้องเริ่มที่ 0 ไม่ใช่ 1 (วนตำแหน่งของสตริง)

ต้องเขียน `s.length()` (มีวงเล็บด้วย) และ
ต้องเป็นเครื่องหมาย < เท่านั้นไม่ใช่ <=

โจทย์ข้อที่ 36 [ระดับง่าย] จงเติมเต็มคำสั่งการประกาศและกำหนดค่าของตัวแปรต่อไปนี้ให้สมบูรณ์ พร้อมทั้งระบุค่าที่เก็บอยู่ในตัวแปรแต่ละข้อ (22 คะแนน)

```
String a = "I Love Java";
String b = "Chula ";
```

ข้อ	ประเภทตัวแปร	การประกาศและกำหนดค่าตัวแปร	ค่าที่เก็บในตัวแปร
1.		<code>var1 = b.length();</code>	
2.		<code>var2 = b.trim().length();</code>	
3.		<code>var3 = "\t".length();</code>	
4.		<code>var4 = "\n\t\u0000".length();</code>	
5.		<code>var5 = b.toUpperCase() + b.toLowerCase();</code>	
6.		<code>var6 = a.charAt(4);</code>	
7.		<code>var7 = a.indexOf("v") + "";</code>	
8.		<code>var8 = a.lastIndexOf("v");</code>	
9.		<code>var9 = a.indexOf("ava");</code>	
10.		<code>var10 = a.lastIndexOf("ava");</code>	
11.		<code>var11 = a.indexOf("java");</code>	
12.		<code>var12 = a.toLowerCase().indexOf("java");</code>	
13.		<code>var13 = a.substring(6, a.length()).trim() + b;</code>	
14.		<code>var14 = a.substring(0, 4);</code>	
15.		<code>var15 = a.toLowerCase().substring(2);</code>	

ข้อ	ประเภทตัวแปร	การประกาศและกำหนดค่าตัวแปร	ค่าที่เก็บในตัวแปร
16.		<code>var16 = a.substring(7).equals("java");</code>	
17.		<code>var17 = "Java".equalsIgnoreCase("java");</code>	
18.		<code>var18 = !"".equals("").toUpperCase());</code>	
19.		<code>var19 = "Jaba".compareTo("Java");</code>	
20.		<code>var20 = "a".compareTo(a.charAt(8));</code>	
21.		<code>var21 = Integer.parseInt(100.0);</code>	
22.		<code>var22 = Double.parseDouble(100);</code>	

โจทย์ข้อที่ 37 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับเบอร์โทรศัพท์ที่ยาว 10 หลักทางแป้นพิมพ์ เช่น 0895993490 แล้วแทรกขีดกลาง (-) เป็น 089-599-3490 พร้อมทั้งแสดงผลลัพธ์บนจอภาพ (10 คะแนน)

โจทย์ข้อที่ 38 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับเลขประจำตัวนิสิตยาว 10 หลักจากทางแป้นพิมพ์ แล้วทำการแบ่งข้อมูลออกเป็นสัดส่วนตามตัวอย่างที่กำหนดให้ (10 คะแนน)

```
Enter ID: 6030079521
Year: 60
Level: 3
Sequence: 79
Check Sum: 5
Faculty: 21
```

โจทย์ข้อที่ 39 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับสตริงทางแป้นพิมพ์หนึ่งชุด แล้วแบ่งสตริงนั้นออกเป็น 2 ส่วนอย่างละเท่ากัน (หรือเกือบเท่ากัน) พร้อมทั้งแสดงผลลัพธ์ออกทางจอภาพ (10 คะแนน)

```
Enter String: ABCDEFG
ABC
DEFG
```

โจทย์ข้อที่ 40 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับประโยคจากแป้นพิมพ์แล้วทำการตัดช่องว่างทุกช่องในประโยคนั้นออกไปโดยให้เหลือเพียงอักขระอื่น ๆ ที่ไม่ใช่ช่องว่าง แล้วเก็บผลลัพธ์ที่ได้ไว้ในสตริงตัวใหม่ พร้อมทั้งแสดงผลออกทางจอภาพดังตัวอย่างต่อไปนี้ (10 คะแนน)

```
Enter Sentence: We love Python
New Sentence: WelovePython
```

โจทย์ข้อที่ 41 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับตัวเลขจำนวนจริงหนึ่งค่าจากแป้นพิมพ์แล้วปัดตำแหน่งทศนิยมให้เหลือเพียง 2 หลัก โดยให้ใช้การดำเนินการกับสตริงเท่านั้น (ห้ามใช้ฟังก์ชันอื่น ๆ) และแสดงผลลัพธ์ออกทางจอภาพให้สวยงาม โดยสมมุติว่าตัวเลขที่รับเข้ามานั้นมีตำแหน่งทศนิยมตั้งแต่ 2 หลักขึ้นไปเสมอ ยกตัวอย่างเช่น 215.4689 จะได้ผลลัพธ์เป็น 215.46 เป็นต้น (10 คะแนน)

โจทย์ข้อที่ 42 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อตัดค่าของวัน เดือน ปี จากตัวแปรสตริงชื่อ `s` ที่รับค่าวัน เดือน ปี จากแป้นพิมพ์ในรูปแบบของ "1/12/2020" หรือ "13/6/2016" เป็นต้น พร้อมทั้งแสดงผลลัพธ์ของวัน เดือน ปี ออกทางจอภาพที่ละบรรทัด (10 คะแนน)

```
Date: 13/6/2016
Day: 13
Month: 6
Year: 2016
```

โจทย์ข้อที่ 43 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับคำ (Word) จากแป้นพิมพ์มา 1 คำที่ลงท้ายด้วย "y" แล้วเปลี่ยนคำนั้นให้เป็นพหูพจน์ โดยถ้าคำนามที่ลงท้ายด้วยสระตามด้วย "y" ให้เติม "s" ต่อท้าย เช่น day จะได้เป็น days เป็นต้น แต่ถ้าคำนามที่ลงท้ายด้วยพยัญชนะตามด้วย "y" ให้เปลี่ยน "y" เป็น "i" แล้วเติม "es" เช่น strawberry จะได้เป็น strawberries เป็นต้น พร้อมทั้งแสดงออกทางจอภาพ (10 คะแนน)

โจทย์ข้อที่ 44 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับประโยคจากทางแป้นพิมพ์เข้ามาหนึ่งประโยค แล้วนับจำนวนอักขระทั้งหมดที่ปรากฏอยู่ในประโยคนั้น โดยจะไม่นับช่องว่างที่อยู่หัวและท้ายประโยค (แต่นับช่องว่างที่อยู่ระหว่างประโยค) และทำการดึงคำแรกของประโยค พร้อมทั้งแสดงผลลัพธ์ทั้งหมดออกทางจอภาพให้สวยงามดังตัวอย่างต่อไปนี้ (10 คะแนน)

```
Enter: Hi Wayne Rooney
Number of Char: 15
First Word: Hi
```

โจทย์ข้อที่ 45 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับคำ (Words) สองคำจากแป้นพิมพ์ภายในบรรทัดเดียวกันคั่นด้วย Comma แล้วทำการตรวจสอบว่า สองคำนี้คำใดมาก่อนกันเมื่อเรียงตามพจนานุกรม

```
Word: pythoN,python
pythoN is before python
```

```
Word: Python,Pythin
Python is after Pythin
```

```
Word: pythoN,pythoN
pythoN equals pythoN
```

ภาษาอังกฤษ (Dictionary) โดยให้แสดงผลลัพธ์ออกทางจอภาพดังตัวอย่างต่อไปนี้ (10 คะแนน)

โจทย์ข้อที่ 46 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับข้อความจากแป้นพิมพ์เข้ามา 2 ข้อความ แล้วตรวจสอบว่าข้อความใดที่มีความยาวมากกว่ากัน โดยให้นำข้อความที่สั้นกว่าแทรกเข้าไประหว่างกลางของข้อความที่ยาวกว่า พร้อมทั้งแสดงผลขึ้นบนจอภาพดังตัวอย่างต่อไปนี้ (10 คะแนน)

```
Message 1: abcdefghijklmn
Message 2: 1234567
          abcdefg1234567hijklmn
```

```
Message 1: 12345678
Message 2: ABCDEFGH
          1234ABCDEFGH5678
```

โจทย์ข้อที่ 47 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับคำ (Words) จำนวน 3 คำจากแป้นพิมพ์ โดยให้อยู่ภายในบรรทัดเดียวกันและคั่นด้วยช่องว่าง แล้วจัดเรียงคำทั้งสามตามพจนานุกรมภาษาอังกฤษ (Dictionary) พร้อมทั้งแสดงผลลัพธ์ออกทางจอภาพดังตัวอย่างต่อไปนี้ (10 คะแนน)

```
Enter Words: Niu Team Klar
Sorted Words: Klar Niu Team
```

```
Enter Words: Python Pythee Pythm
Sorted Words: Pythee Pythm Python
```

โจทย์ข้อที่ 48 [ระดับง่าย] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับข้อความเข้ามาทางแป้นพิมพ์แล้วนับว่ามีอักษร "F" หรือ "f" ปรากฏอยู่ในข้อความนั้นกี่ตัว พร้อมทั้งแสดงผลขึ้นบนจอภาพ (10 คะแนน)

```
Enter Message: FIFA Staff
Count F: 4
```

โจทย์ข้อที่ 49 [ระดับปานกลาง] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับประโยคจากแป้นพิมพ์แล้วทำการกลับ (Reverse) ประโยคนั้นแล้วเก็บไว้ในสตริงตัวใหม่ โดยเรียงอักขระทุกตัวจากหลังมาหน้า (ขวาไปซ้าย) พร้อมทั้งแสดงผลลัพธ์ที่ได้ออกทางจอภาพดังตัวอย่างต่อไปนี้ (10 คะแนน)

```
Enter Sentence: We love Python
Reverse Sentence: nohtyP evol eW
```

โจทย์ข้อที่ 50 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับบิตสตริง (Bit String) สองชุดเข้ามาทางแป้นพิมพ์แล้วทำการตรวจสอบว่าบิตสตริงทั้งสองชุดนั้นเป็นคู่อินเวอร์ส (Inverse Pair) กันหรือไม่ โดยนิยามของ

คู่อินเวอร์สคือบิตสตริงสองชุดใด ๆ ที่มีความยาวเท่ากันและตำแหน่งบิตที่ตรงกันคูใดคู่หนึ่งจะมีค่าตรงกันข้ามกันเสมอระหว่างบิต 0 และบิต 1 เช่น บิตสตริง 110100 มีคู่อินเวอร์สเป็น 001011 เป็นต้น (10 คะแนน)

```
Enter Bit String 1: 1111100011
Enter Bit String 2: 0010110010
Not Inverse Pair
```

```
Enter Bit String 1: 010111100
Enter Bit String 2: 101000011
Inverse Pair
```

โจทย์ข้อที่ 51 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาที่สมบูรณ์เพื่อรับข้อความใด ๆ เข้ามาทางแป้นพิมพ์ แล้วนับอักขระที่ปรากฏในข้อความว่าประกอบไปด้วยตัวเลข (Digit) ตัวอักษร (Letter) และสัญลักษณ์ (Symbol) อย่างละกี่ตัว โดยที่ตัวเลขได้แก่ 0-9 ตัวอักษรได้แก่ A-Z และ a-z และสัญลักษณ์ได้แก่อักขระอื่น ๆ ที่ไม่ใช่ตัวเลขและตัวอักษร พร้อมแสดงผลที่ได้ขึ้นบนจอภาพดังตัวอย่างต่อไปนี้ (10 คะแนน)

```
Message: WE2get4#8+$Time9
Digit: 4
Letter: 9
Symbol: 3
```

โจทย์ข้อที่ 52 [ระดับยาก] จงเขียนโปรแกรมภาษาจาวาให้สมบูรณ์เพื่อรับข้อความหนึ่งข้อความจากแป้นพิมพ์ แล้วนำมาตรวจสอบว่าข้อความดังกล่าวเป็นข้อความแบบพาลินโดรม (Palindrome) หรือไม่ โดยพาลินโดรมคือข้อความที่อ่านจากข้างหน้าไปข้างหลังเหมือนกับอ่านจากข้างหลังไปข้างหน้า เช่น "level", "civic", "deified", "2552", "A Toyota", "I prefer pi", "Never odd or even" เป็นต้น ทั้งนี้จะไม่สนใจช่องว่างและตัวอักษรพิมพ์ใหญ่หรือพิมพ์เล็กในการอ่านแต่อย่างใด (10 คะแนน)

```
Text: level
It is palindrome
```

```
Text: I prefer Java
It is not palindrome
```

```
Text: I prefer pi
It is palindrome
```

โจทย์ข้อที่ 53 [ระดับเทพ] ในเดือนพฤษภาคม 2558 ได้มีการเผยแพร่คลิปวิดีโอและโค้ดลับของลัทธิชาตานลงในเว็บไซต์ GadgetZZ.com โดยส่วนหนึ่งของโค้ดลับคือรหัสสมอร์ส (Morse Code) ที่ถอดออกมาเป็นข้อความได้ว่า "RED LIPS LIKE TENTH" เมื่อทำการตรวจสอบข้อความอย่างละเอียดพบว่า เป็นคำสลับอักษรหรือที่เรียกว่า แอนาแกรม (Anagram) ของข้อความ "KILL THE PRESIDENT" ซึ่งยังคงเป็นปริศนามาจนถึงปัจจุบัน เพื่อช่วยในการคลี่คลายปริศนาดังกล่าวไม่มากนักน้อย จงเขียนโปรแกรมเพื่อรับข้อความใด ๆ 2 ชุดเข้ามาทางแป้นพิมพ์ แล้วทำการตรวจสอบว่าข้อความดังกล่าวเป็นแอนาแกรมของกันและกันหรือไม่ พร้อมทั้งแสดงผลที่ได้ขึ้นบนจอภาพ (10 คะแนน)