

2110101 : การทำโปรแกรมคอมพิวเตอร์

ระบบคอมพิวเตอร์เบื้องต้น

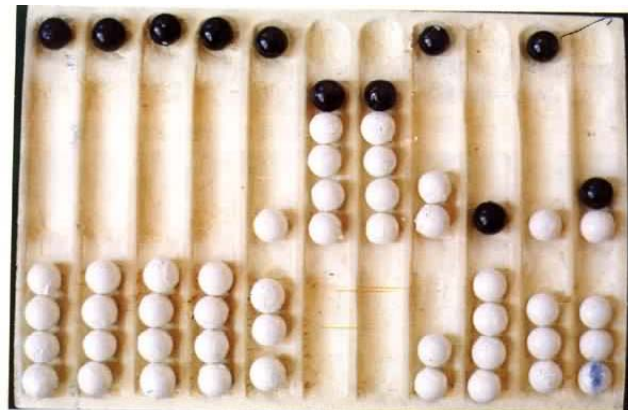
ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

- รู้จักคอมพิวเตอร์
- ประยุกต์คอมพิวเตอร์
- แก้ปัญหาทางวิศวกรรมด้วยคอมพิวเตอร์
- **ทำโปรแกรมคอมพิวเตอร์**

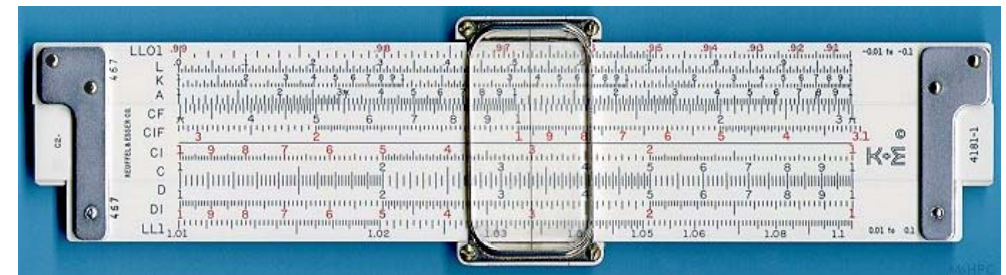
2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (25/05/52)

2

300 B.C. : Babylonians



1632 : Slide Rule



1642 : Blaise Pascal (Pascaline)

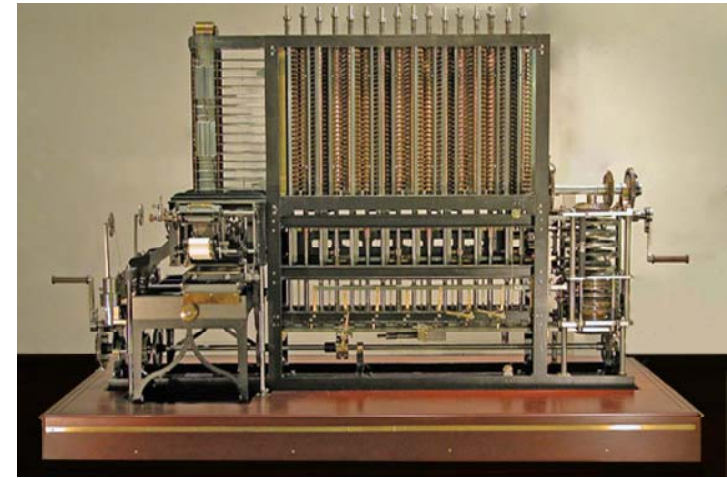


Blaise Pascal
1623-1662



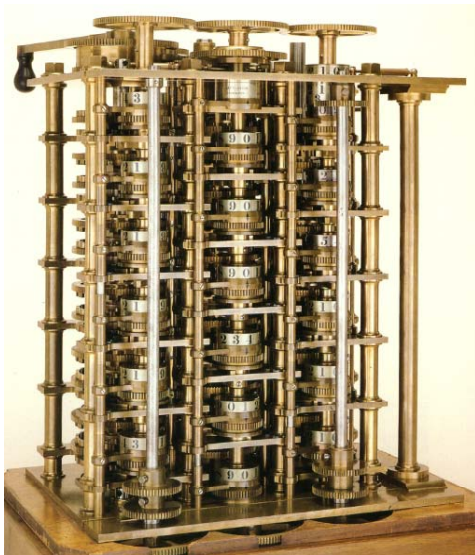
Mechanical calculator
Add and subtract

1822 : Charles Babbage (Difference Engine)



Compute tables of numbers, such as logarithm tables

1833 : Charles Babbage (Analytical Engine)



- The first automatic computing machine
- It has central processing unit, memory storage, software instructions, punch card inputs, and printed outputs
 - 50 decimal digit calculations
 - Memory of 1000 digits
- Operated by steam power !
- Babbage failed to build the engine

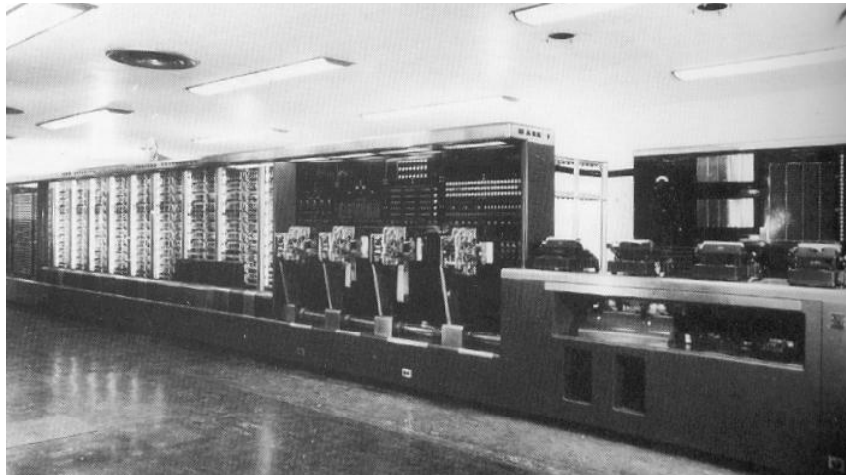
1890 : Herman Hollerith (Hollerith desk)



เครื่องอ่านบัตรเจาะรู
เพื่อนับข้อมูลทางสถิติ
สำมะโนครัวประชากร

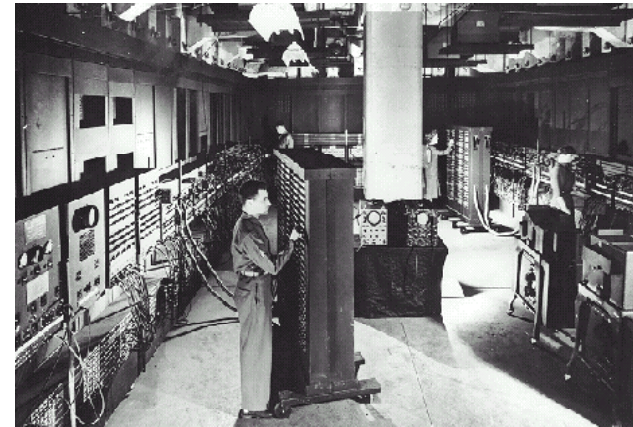


1940 : Harvard Mark I



An electro-mechanical computer was constructed out of switches, relays, rotating shafts, and clutches , weighed 5 tons, ran non-stop for 15 years.

1946 : ENIAC (used for H-bomb research)



Use **vacuum tubes** for fast switching (1000 times faster than relays). 30 tons, 15000 square feet, 18000 vacuum tubes, 140 KW power dissipation

คอมพิวเตอร์ในแต่ละยุค

- หลอดสุญญากาศ 40 K ops/sec
- ทรานซิสเตอร์ 200 K ops/sec
- วงจรรวมความจุน้อย 1 M ops/sec
- วงจรรวมความจุมาก (LSI) 10 M ops/sec
- วงจรรวมความจุสูงมาก (VLSI) 1000 M ops/sec



Apple Computers



70 กว่าปีที่ผ่านมามี

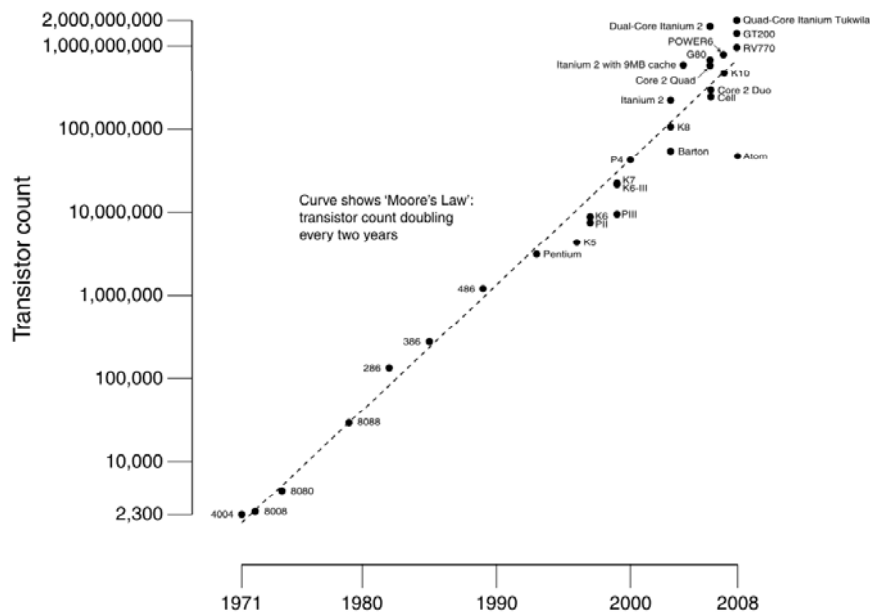
- เล็กลง (ร้อยล้านทรานซิสเตอร์ปลายนิ้วก้อย)
- เร็วขึ้น (พันล้านคำสั่งในวินาที)
- จุข้อมูลมากขึ้น (แสนล้านตัวอักษรในฝ่ามือ)
- กินไฟน้อยลง (ใช้ไฟทั้งอำเภอลดเหลือเท่าทีวี)
- ราคาถูกลง (RAM จากไบต์ละบาทเป็นล้านไบต์บาท)

Moore's Law : เทคโนโลยีทางคอมพิวเตอร์มีประสิทธิภาพมากขึ้นสองเท่าทุกๆ 18 เดือน

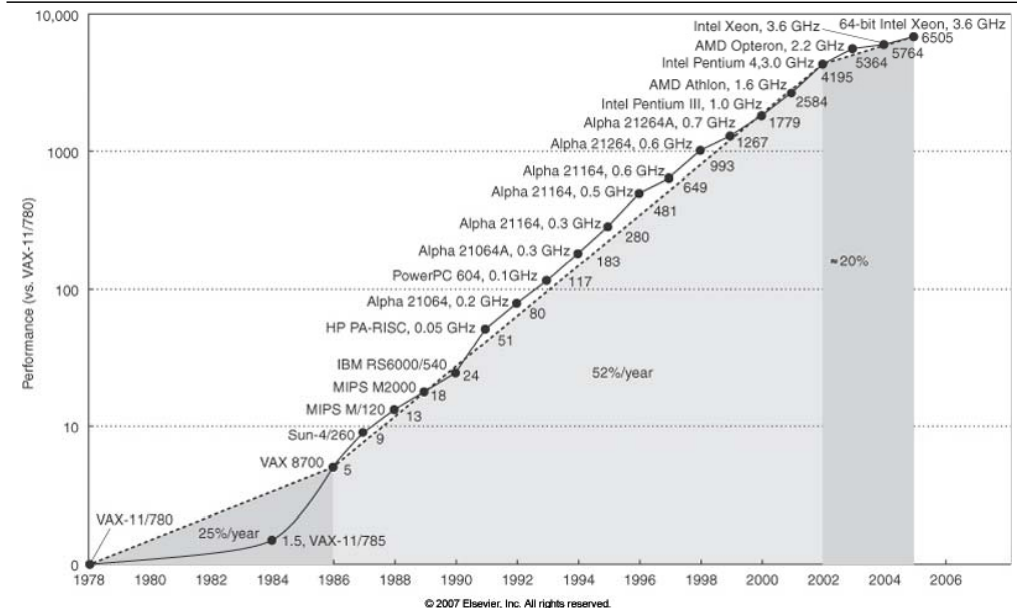
Game Consoles

- 1983 NES (Famicom) 8-bit CPU, 1.79 MHz, 2KB, \$160
- 1989 Sega Genesis 16-bit CPU, 7.67MHz, 64KB, \$200
- 1991 Super NES 16-bit CPU, 3.58MHz, 128KB, \$200
- 1994 Sega Saturn 32-bit CPU, 28.63MHz, 2MB
- 1994 PlayStation 1 32-bit CPU, 33.86MHz, 2MB
- 1996 Nintendo 64 64-bit CPU, 93.75MHz, 4MB
- 1999 Sega dreamcast 128-bit CPU, 200MHz, 16MB
- 2000 PlayStation 2 64-bit CPU, 299MHz, 32MB
- 2001 Gamecube 32/64-bit CPU, 485MHz, 43MB
- 2001 Xbox 32-bit CPU 733MHz, 64MB
- 2005 Xbox 360 3.2GHz, 512MB
- 2006 Wii 512MB 32/64-bit CPU, 729MHz, 88MB
- 2006 PlayStation 3 32/64-bit CPU, 3.2GHz, 256MB

จำนวนทรานซิสเตอร์



ประสิทธิภาพของหน่วยประมวลผล



คอมพิวเตอร์

- เครื่องประมวลผล จัดเก็บ และสื่อสารข้อมูล
 - ประมวลผลข้อมูลได้รวดเร็ว
 - จัดเก็บข้อมูลได้มากมาย
 - สื่อสารกันได้อย่างไร้พรมแดน
- การใช้งาน
 - บันทึกลง
 - เกมส์ ดูนั่ง ฟังเพลง ...
 - จัดเก็บและค้นคืน
 - WWW, ระบบทะเบียนราษฎร์ ห้องสมุด ...
 - ธุรกิจ
 - email ระบบบุคลากร, วิเคราะห์ข้อมูล, วางแผนทรัพยากร ...
 - ...



คุณลักษณะของคอมพิวเตอร์

- เชื่อสัจย์ ทำตามคำสั่ง
- ไม่บ่น ทำแต่เรื่องซ้ำๆ ซากๆ
- จำได้มาก จำได้แม่น
- ทำงานรวดเร็ว
- ไม่ฉลาด สอนอย่างไรก็ทำอย่างนั้น



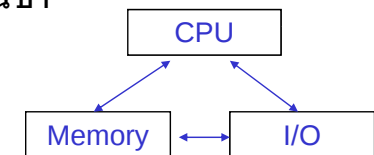
องค์ประกอบของคอมพิวเตอร์

- hardware เพื่อประมวลผล จัดเก็บ และสื่อสารข้อมูล
- software เพื่อสั่งให้ hardware ทำงานตามขั้นตอนที่กำหนดไว้



Hardware

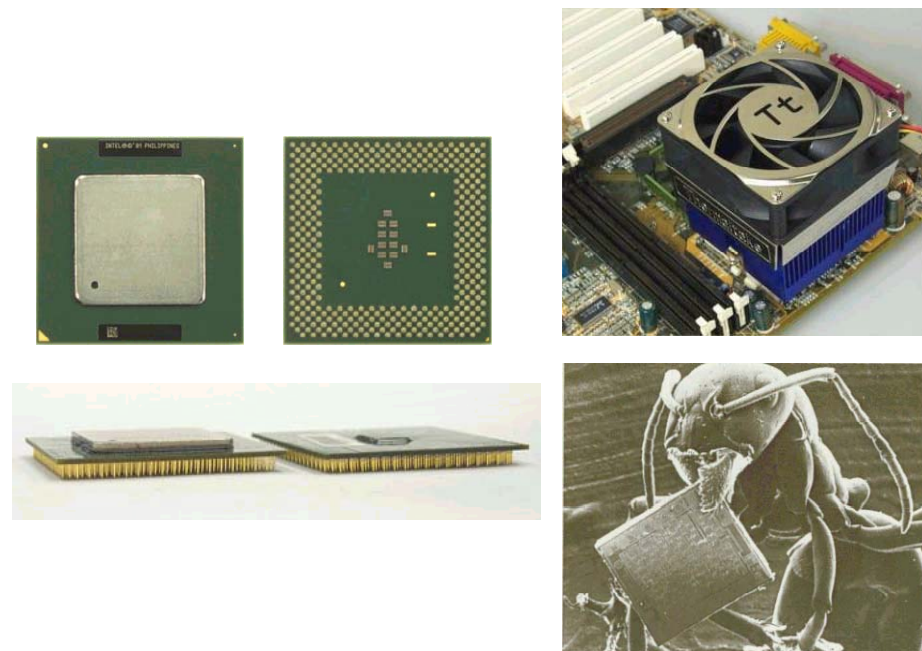
- Central Processing Unit (CPU)
 - เพื่อประมวลผลทางคณิตศาสตร์, ตรรกะ, การทดสอบจริงเท็จ, การทำงานซ้ำ
- Memory
 - เพื่อจัดเก็บและค้นคืนข้อมูล
- I/O
 - เพื่อรับส่งสื่อสารข้อมูล (จอภาพ แป้นพิมพ์ เมาส์ โมเด็ม เครื่องพิมพ์ กล้อง ...)



Central Processing Unit

- ทำงานตามชุดคำสั่งที่กำหนดให้
- ทำงานได้รวดเร็ว (พันล้านคำสั่งในวินาที)
- ควบคุมการทำงานของส่วนอื่นๆ
- ชุดคำสั่ง
 - เป็นรหัสฐานสอง
 - เก็บในหน่วยความจำ
 - เปลี่ยนแปลงได้
- ยี่ห้อ
 - Intel Pentium, Intel Atom, Intel StrongARM, AMD Xeon, IBM PowerPC, ...

CPU : Microprocessor



Memory

- เก็บชุดคำสั่งการทำงานของ CPU
- เก็บข้อมูลสำหรับการประมวลผล
- แบ่งเป็น
 - main memory
 - ที่เก็บชั่วคราว
 - เร็วแต่ราคาแพง (10^{-8} วินาที ล้านไบต์บาท)
 - RAM (random access memory)
 - secondary storage
 - ที่เก็บถาวร
 - ช้ากว่าแต่ราคาถูก (HD : 0.01 วินาที ร้อยล้านไบต์บาท)
 - hard disk, CD-ROM, DVD-ROM, flash drive, ...

Memory



Input/Output Devices

- อุปกรณ์รับข้อมูล
 - แป้นพิมพ์ เมาส์ แป้นกดเกมส์ กล้อง สแกนเนอร์ โมเด็ม ไมโครโฟน เครื่องอ่านบาร์โค้ด เครื่องอ่านบัตร
- อุปกรณ์ส่งหรือแสดงข้อมูล
 - จอภาพ เครื่องพิมพ์ โมเด็ม ลำโพง



หน่วยต่างๆ ที่น่าสนใจ

- bit - หน่วยเล็กสุดแทนสองสถานะ (binary digit)
- Byte - 8 bits โดยทั่วไป 32 bits
- word - จำนวน bytes ที่เครื่องประมวลผลได้สะดวก
- CPU : 900MHz, 2.5GHz
- RAM : 512MB, 4GB
- Harddisk : 500GB, 1TB, 7200rpm
- Printer : 600dpi, 10ppm
- Wi-Fi : 802.11b (11Mbps), 802.11g (54Mbps)
- LAN : 10/100/1000 Mbps

K – Kilo = 1,000

M – Mega = 1000K

G – Giga = 1000M

T – Tera = 1000G

B – Byte

b – bit

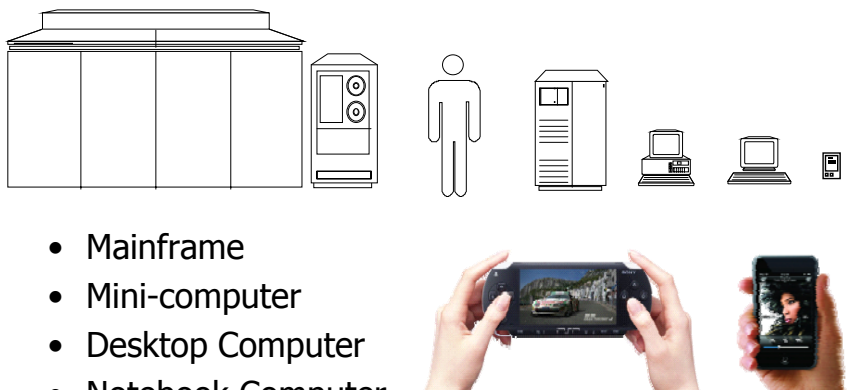
bps – bits per second

ppm – pages per minute

rpm – rounds per minute

dpi – dots per inch

ประเภท : ขนาดและน้ำหนัก



- Mainframe
- Mini-computer
- Desktop Computer
- Notebook Computer
- Personal Digital Assistant

Reliability

Type	Platform	Outages/year
Mainframe	IBM S/390 (Sysplex)	10 min.
	Tandem Nonstop	1.7 Hrs.
Mini	Digital VAX	18.9 Hrs.
	Unix-based	23.8 Hrs.
PC	Windows-NT based	224.5 Hrs.

ตัวอย่าง : คอมพิวเตอร์

- CPU : Intel® Core 2 Duo Processor 2.6 GHz

- Memory

- 2GB DDR2-667
- HD 320 GB SATA, 7200RPM
- Diskette drive : 1.44MB
- DVD-RW 16X SuperMulti Function



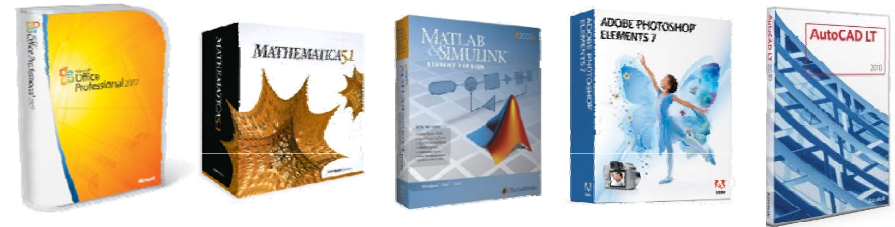
20,000 บาท

- I/O

- NVIDIA® GeForce® 7100 graphics
- Monitor : LCD 19" 1280 x 1024
- Sound card : Audio Codec 7.1 Surround Sound
- Gigabit Ethernet LAN on Board
- 10 USB 2.0 ports (4 Front 6 Rear)
- Internal 56K fax-modem

Software

- ชุดคำสั่งต่างๆ ซึ่งสั่งให้ hardware ทำงานตามขั้นตอนที่กำหนดไว้
- เราเก็บ software มากมายไว้ใน secondary storage (harddisk, CD-ROM, DVD-ROM)
- ระบบนำ (หรือ load) software ที่ต้องการลงใน main memory เพื่อใช้งาน



Software

- System software

- Operating system (Windows XP, Vista, Linux, Mac OS X)
- System tools (antivirus, zip/unzip, backup/restore, ...)
- Development tools (compiler, debugger, ...)
- ...

- Application software

- Office (email, word processor, spreadsheet, ...)
- Network applications (browser, chat, ...)
- Entertainment (movies, musics, games, ...)
- ...

ขั้นตอนการพัฒนาซอฟต์แวร์

หัวข้อ

- องค์ประกอบของโปรแกรม
- คำสั่งแสดงผล
- การแปลโปรแกรม
- โปรแกรมที่อ่านง่าย
- วงจรการพัฒนาโปรแกรม

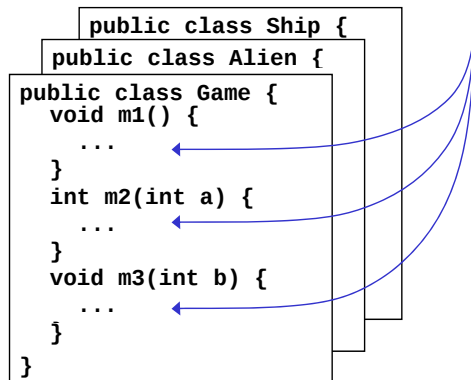
2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (25/05/52)

2

องค์ประกอบของโปรแกรม

- หนึ่งโปรแกรม ประกอบด้วยหลายคลาส
- หนึ่งคลาส ประกอบด้วยหลายเมทอด
- หนึ่งเมทอด ประกอบด้วยหลายคำสั่ง

```
public class Ship {  
    public class Alien {  
        public class Game {  
            void m1() {  
                ...  
            }  
            int m2(int a) {  
                ...  
            }  
            void m3(int b) {  
                ...  
            }  
        }  
    }  
}
```



คำสั่งมีหลายแบบ

- เตรียมที่เก็บ `double n, c = 0;`
- คำนวณ `n = len * Math.sin(r);`
- ตัดสินใจ `if (n == 0) return;`
- ทำซ้ำ `while(n > 0) {
 n = n/2;
 c++;
}`
- แสดงผล `System.out.println(c);`
- และอื่น ๆ ...

ช่วงแรก

- หนึ่งโปรแกรม → หนึ่งคลาส → หนึ่งเมทอด

```
public class ชื่อคลาส {

    public static void main(String[] args) {

        คำสั่ง ;
        คำสั่ง ;
        คำสั่ง ;
        ...
    }
}
```

สั่งคลาสใดทำงาน จะ
เริ่มทำคำสั่งในเมทอด
main ของคลาสนั้น

ทุกคำสั่งต้อง
ปิดท้ายด้วย ;

System.out.print, println

- แสดงผลเป็นข้อความทางจอภาพ

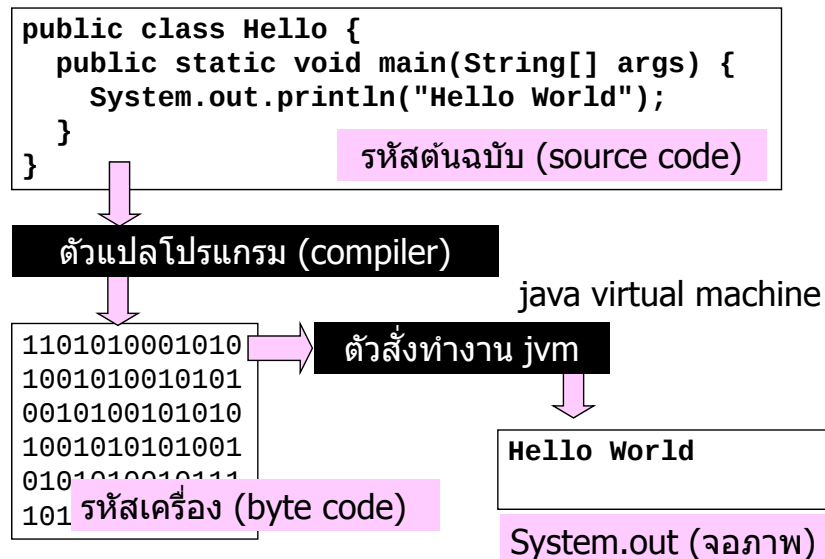
```
public class Hello {
    public static void main(String[] args) {
        System.out.println("Hello World");
        System.out.print("Hello ");
        System.out.println("Java");
        System.out.print( "4+5 = " );
        System.out.println( 4+5 );
    }
}
```

เรียกข้อความที่เขียนครอบด้วย " " ว่า สตริง (String)

System.out

```
Hello World
Hello Java
4+5 = 9
—
```

รหัสต้นฉบับ → รหัสเครื่อง



โปรแกรมต้อง

- เขียนคำสั่งถูกต้องตามหลักภาษา
 - System.out.~~display~~[~~สวัสดี~~]
- ทำงานถูกต้องตามข้อกำหนด
 - circleArea = ~~2 * 3.14159 * r;~~
- เขียนให้อ่านง่าย

```
public class Hello { public static void main(String[]
args) { System.out.println("Hello World");
System.out.print("Hello "); System.out.println
("World"); System.out.print( "4+5 = " ); System.
out.println( 4
+5 )
; }}

```

ตัวแปลโปรแกรม

- ตรวจสอบความถูกต้องของรหัสต้นฉบับ
 - ถ้าผิด : แจ้งความผิดพลาดให้ทราบ
 - ถ้าถูก : สร้างรหัสเครื่องจากรหัสต้นฉบับ

```
public CLASS Hello {  
    puBlic static void main(String[] args) {  
        System.out.println("Hello World")  
    }  
}
```

รหัสต้นฉบับต้องถูกหลักไวยากรณ์ของจาวา 100 %

ผิดที่ไหน ?

```
public class Hello {  
    public static void main(String a) {  
        System.out.println("Hello World");  
        system.out.println("Hello ")  
        System.out.println('Hello');  
    }  
}
```

แก้ไขให้ถูกต้อง

ผิดที่ไหน ?

```
public class LongString {  
    public static void main(String[] args) {  
        System.out.println("พระเสด็จถามพระเสด็จว่า  
        พระเสด็จจะเสด็จไปก่อนหรือไม่เสด็จ  
        ถ้าพระเสด็จไม่เสด็จ พระเสด็จจะเสด็จไปก่อน");  
    }  
}
```

แก้ไขให้ถูกต้อง

โปรแกรมสวย อ่านง่าย

```
public class Hello { public static void  
main(String[] args) { System.out  
.println("Hello World"); System.  
out.print("Hello "); System. out.println  
("World"); System.out.print( "4+5 = " );  
System.  
out.println( 4  
+5 )  
; }}
```

```
public class Hello {  
    public static void main(String[] args) {  
        System.out.println("Hello World");  
        System.out.print("Hello ");  
        System.out.println("Java");  
        System.out.print( "4+5 = " );  
        System.out.println( 4+5 );  
    }  
}
```

เติมหมายเหตุ (comment)

```

/*
รายละเอียด : โปรแกรมสาธิตการใช้งานคำสั่งแสดงผล
ผู้เขียน : สมชาย ประสิทธิ์จตุระกุล
วันที่ : 23 ตุลาคม 2551
*/
public class Hello {
    public static void main(String[] args) {
        System.out.println("Hello World");
        System.out.print("Hello ");
        System.out.println("Java");
        System.out.print( "4+5 = " );
        System.out.println( 4+5 ); // คำนวณแล้วค่อยแสดง
    }
}

```

หมายเหตุระหว่าง /* กับ */

เริ่มหมายเหตุจนสุดบรรทัด

สว่ยเกินไปอาจผิดกฎ

```

*****
* public class Hello {
*
*     PUBLIC STATIC VOID main(String[] args) {
*
*         System.out.print("Hello ");
*
*         System.out.println("Java");
*
*     }
*
* }
*****

```

เขียนได้หรือไม่

```

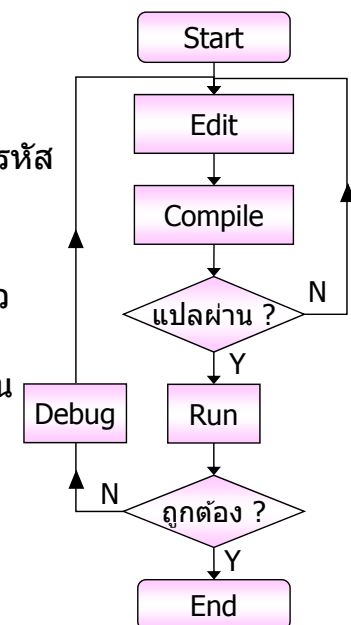
public class Comment1 {
    public static void main(String[] args) {
        System.out. /*comment*/ println("Hello");
        System.out. //      println("Hello");
    }
}

```



วงจรการพัฒนาการโปรแกรม

- Edit : เขียนโปรแกรมต้นฉบับ (source program, source code)
- Compile : แปล source code เป็นรหัสเครื่อง (byte code)
- Run : สั่งทำงานรหัสเครื่องที่ได้
- Debug : หาดำเนินการที่เราเขียนแล้วทำงานผิด (จะสาธิตให้ชมในอนาคต)
- ใช้ซอฟต์แวร์ช่วยพัฒนาโปรแกรมใน "Edit-Compile-Run-Debug"



ก่อนเริ่มเขียนโปรแกรม

Java SE Development Kit



SE : Standard Edition

ME : Micro Edition

EE : Enterprise Edition



Integrated Development Environment



www.netbeans.org



BlueJ

www.bluej.org



www.eclipse.org



CD-ROM : Java + JLab

- copy ลง CD-R เองได้ที่ศูนย์เรียนรู้ (ศูนย์คอมฯ ชั้น 2)
- หรือใช้ CD-ROM หลังหนังสือ



สรุป

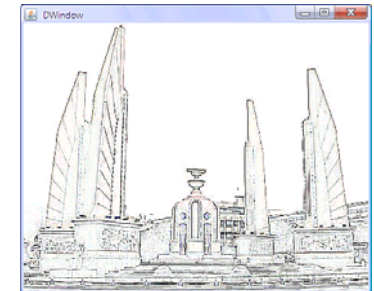
- หนึ่งโปรแกรมมีหลายคลาส
- หนึ่งคลาสมีหลายเมทอด
- หนึ่งเมทอดมีหลายคำสั่ง
- โปรแกรมจาวาเริ่มทำงานที่เมทอดชื่อ main
- การทำงานจะทำทีละคำสั่งตามลำดับจากบนลงล่าง
- โปรแกรมต้องถูกหลักไวยากรณ์
- โปรแกรมต้องทำงานถูกต้องตามข้อกำหนด
- โปรแกรมต้องอ่านเข้าใจได้ง่าย ๆ
- วงจรการพัฒนาโปรแกรม
edit – compile – run – debug

การประมวลผลจำนวน

คำนวณ ๆ ๆ



อ่านภาพเก็บเป็น matrix



ได้ภาพเส้นขอบ

$$q_{x,y} = b + \left(\sum_{i=0}^2 \sum_{j=0}^2 f_{i,j} \cdot p_{x+i-1,y+i-1} \right) \text{ if } \sum f = 0$$

(matrix convolution)

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (25/05/52)

2

หัวข้อ

- ตัวดำเนินการ : + - * /
- ประเภทข้อมูล : int double
- ที่เก็บข้อมูล : ตัวแปร
- การอ่านข้อมูลจากแป้นพิมพ์ : Scanner
- ฟังก์ชันทางคณิตศาสตร์ : Math

บวก ลบ คูณ หาร

```
public class Operators {
    public static void main(String[] args) {
        System.out.println( 4 + 2 );
        System.out.println( 4 - 2 );
        System.out.println( 4 * 2 );
        System.out.println( 4 / 2 );
        System.out.println( 1+2+3+4+5+6+7+8+9+10 );
    }
}
```

ลำดับการคำนวณ

- $2 + 3 * 4 = ?$
- ลองกด $2 + 3 \times 4$ ด้วยเครื่องคิดเลข
 - ใช้ Citizen CT-600 ได้ 20
 - ใช้ Sharp EL-515s ได้ 14
- ในจาวาบนเครื่องใด ๆ $2 + 3 * 4$ ได้ 14 เสมอ
- จาวากำหนดลำดับการคำนวณดังนี้

- ทำในวงเล็บ
- ติดลบ
- คูณ กับหาร
- บวก กับ ลบ

$2 * 3 + 8 / -(2 - 4) - 1$	
$2 * 3 + 8 / -(-2) - 1$	
$2 * 3 + 8 / 2$	- 1
$6 + 8 / 2$	- 1
$6 + 4$	- 1
10	- 1
	9

บวกกับสตริง คือการต่อสตริง

```
public class StringConcatenation {
    public static void main(String[] args) {
        System.out.println( "a" + "b" );    // "ab"
        System.out.println( "a" + 1 );      // "a1"
        System.out.println( 1 + "a" );      // "1a"
        System.out.println( "1" + "2" );    // "12"
        System.out.println( "1" + 2 );      // "12"
        System.out.println( 1 + "2" );      // "12"
        System.out.println( "a" + 1 + 2 );  // "a12"
        System.out.println( "a" + (1 + 2) ); // "a3"
        System.out.println( "a" + (1 - 2) ); // "a-1"
        System.out.println( "a" + 1 - 2 );  // ผิด
    }
}
```

"a" + 1 + 2
"a1" + 2
"a12"

1 + 2 + "a"
3 + "a"
"3a"

ตัวอย่าง : พื้นที่ของวงกลมรัศมี 4

```
public class CircleArea {
    public static void main(String[] args) {
        System.out.print( "พื้นที่วงกลมรัศมี 4 = " );
        System.out.println( 3.14159 * 4 * 4 );
    }
}
```

```
public class CircleArea {
    public static void main(String[] args) {
        System.out.println( "พื้นที่วงกลมรัศมี 4 = " +
            ( 3.14159 * 4 * 4 ) );
    }
}
```

ใส่วงเล็บบังคับให้คำนวณก่อน

จำนวนเต็ม กับ จำนวนจริง

0

0.0

5

10.

-10

1.5

201

-20000

~~20,000~~

-2e4

-2×10^4

~~030~~

5E-3

5×10^{-3}

0.5e-2

เลขยกกำลังต้อง
เป็นจำนวนเต็ม

50e-4

~~1.5e0.5~~

~~$1.5 \times 10^{-0.5}$~~

เริ่มด้วย 0 จาวาอ่านแบบ
ฐานแปด ∴ อย่าเขียน

จำนวนเต็ม กับ จำนวนจริง

```
public class IntegerAndReal {
    public static void main(String[] args) {
        System.out.println( 5 / 2 );           // 2
        System.out.println( 5.0 / 2.0 );       // 2.5
        System.out.println( 5.0 / 2 );         // 2.5
        System.out.println( 5 / 2.0 );         // 2.5
        System.out.println( 5.25 / 0.5 );      // 10.5
    }
}
```

จาวาจะเปลี่ยนจำนวนเต็มที่ดำเนินการกับ
จำนวนจริงให้เป็นจำนวนจริงก่อน

คำถาม

$1/2 + 1/2$

$1.0/2 + 1/2.0$

$0.0 + 1/2$

$(0.0 + 1)/2$

ตัวแปร (Variables)

data sum count
average

sum = data

ทำสำเนาข้อมูล
ใน data ไปเก็บ
ใน sum

ไม่ใช่ย้ายข้อมูล
ใน data ไปไว้ที่
sum

```
sum    = 0.0; count = 0;
data   = 10.0;
sum    = data;
count  = count + 1;
data   = 20;
sum    = sum + data;
count  = count + 1;
data   = 69;
sum    = sum + data;
count  = count + 1;
average = sum / count;
```

การประกาศตัวแปร

- ต้องตั้งชื่อตัวแปร
- ต้องกำหนดประเภทข้อมูลที่จะเก็บ
- ต้องประกาศก่อนใช้

```
public class Variables {
    public static void main(String[] args) {
        int count = 0;
        double sum = 0.0;

        ...
    }
}
```

ตั้งค่าเริ่มต้นตอนประกาศก็ได้

int double String

ตัวอย่าง : การประกาศตัวแปร

```
int n;  
double a, b, c;  
double x = 0, y, z = 8.5;  
String s = "hello", t;  
int p; double q;
```

ถูก

```
Int s;  
string t;  
y = 6.6;  
double y;  
int p, double q;
```

ผิด

ตัวอย่าง : พื้นที่ของวงกลมรัศมี 4

```
public class CircleArea {  
    public static void main(String[] args) {  
        double r = 4.0;  
        double pi = 3.14159;  
        String msg = "พื้นที่วงกลมรัศมี";  
        double area = pi * r * r;  
        System.out.println(msg + r + "=" + area);  
    }  
}
```

พื้นที่วงกลมรัศมี 4.0 = 50.26544

กฎการตั้งชื่อ

- ชื่อประกอบด้วยตัวอักษร ตัวเลข \$ หรือ _
- ชื่อห้ามขึ้นต้นด้วยตัวเลข
- ชื่อยาว ๆ ได้
- ตัวอักษรตัวใหญ่ไม่เหมือนตัวเล็ก
- ต้องไม่ซ้ำกับคำสงวนของภาษาจาวา
- ต้องไม่ซ้ำกับชื่ออื่นของโปรแกรมที่ใช้มาก่อน

ถูก

```
int17 power Public day_of_week
```

ผิด

```
7zean String ohOH! ed-edd-n-eddy
```

คำสงวนของจาวา

abstract	assert	boolean	break
byte	case	catch	char
class	const	continue	default
do	double	else	enum
extends	false	final	finally
float	for	goto	if
implements	import	instanceof	int
interface	long	native	new
null	package	private	protected
public	return	short	static
strictfp	super	switch	synchronized
this	throw	throws	transient
true	try	void	volatile
while			

ธรรมเนียมนิยมในการตั้งชื่อ

- ชื่อคลาสขึ้นด้วยตัวอักษร (อังกฤษ) ใหญ่
- ชื่อตัวแปรขึ้นด้วยตัวอักษร (อังกฤษ) เล็ก
- ตั้งให้สื่อความหมาย
- ถ้าเป็นวลี (มีหลายคำ) เขียนแบบ "หลังอุล्ल"
- หลีกเลี่ยงการใช้ \$ และ _ (\$\$\$\$__\$\$\$\$ ไม่ผิด)

```
public class Demo {
    public static void main(String[] args) {
        double weight = 76.0;
        double powerLevel = 5.0;
        int dayOfWeek = 4;
        ...
    }
}
```

dayOfWeek

การให้ค่ากับตัวแปร

- ตัวแปรเก็บข้อมูลได้เฉพาะแบบที่ประกาศไว้


```
int a = 20; // ถูก
double x = 1.25; // ถูก
int b = 1.5; // ผิด
int c = 1.0; // ผิด
```
- ยกเว้น : นำจำนวนเต็มเก็บใส่ double ได้


```
double y = 2; // จาวาเปลี่ยน 2 เป็น 2.0
```
- นำสตริงไปใส่ตัวแปรแบบ int กับ double ไม่ได้


```
int n = "1"; // ผิด
```
- นำ int กับ double ไปเก็บในตัวแปรสตริงไม่ได้


```
String s = 1; // ผิด
```

ตัวอย่าง

`x = y + 2;` ให้ x มีค่าเท่ากับ y + 2

`x = x + 1;` เพิ่มค่าของตัวแปร x อีก 1 $x \leftarrow x + 1$

`x = 2 * x;` เพิ่มค่าของตัวแปร x อีกเท่าตัว

`x = 0 - x;` `x = -1 * x;` `x = -x;` เหมือนกันทั้งสามคำสั่ง

`dx = x1 - x2;`
`dy = y1 - y2;`
`len2 = dx*dx + dy*dy;`

$$len2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$$

`double d = 1234;` นำจำนวนเต็มใส่ในตัวแปรจำนวนจริงได้

ตัวอย่างผิดๆ

`5 = x;` `x + 2 = x;` `x = 2a + 8;`

`String s = 5;` `int k = "4";` `int k = 1.5;`

การเปลี่ยนประเภทข้อมูล

```
int a = 1, b = 2, c;
double d = 2.6;
double x = 7;

c = (int) d; // 2 <- 2.6
d = (double) a; // 1.0 <- 1
d = ((double) a) / ((double) b); // 1.0/2.0
d = (double) a / (double) b; // 1.0/2.0
d = (double) a / b; // 1.0/2
d = a / (double) b; // 1/2.0
d = (double) (a / b); // 1/2
```

type casting

การอ่านข้อมูลจากแป้นพิมพ์

```
import java.util.Scanner;

public class CircleArea {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("รัศมี = ");
        double r = kb.nextDouble();
        double pi = 3.14159;
        String msg = "พื้นที่วงกลมรัศมี ";
        double area = pi * r * r;
        System.out.println(msg + r + " = " + area);
    }
}
```

บอกตัวแปรโปรแกรมให้ใช้
ชื่อย่อ Scanner ได้

สร้างตัวอ่าน
แป้นพิมพ์

รัศมี = 4
พื้นที่วงกลมรัศมี 4.0 = 50.26544

new Scanner(System.in)

- **System.in** : แป้นพิมพ์
- **Scanner** : ตัวอ่าน
- **new Scanner(System.in)**
สร้าง ตัวอ่าน จาก แป้นพิมพ์
- **Scanner kb** : ตัวแปรแบบ Scanner
- **Scanner kb = new Scanner(System.in);**
- **kb.nextDouble()** : อ่านจำนวนจริงจากผู้ใช้
- **kb.nextInt()** : อ่านจำนวนเต็มจากผู้ใช้
- **kb.nextLine()** : อ่านสตริงหนึ่งบรรทัด

ตัวอย่าง : ดัชนีมวลกาย

- เขียนโปรแกรมรับข้อมูลทางแป้นพิมพ์
 - น้ำหนัก (หน่วยเป็นกิโลกรัม)
 - ความสูง (หน่วยเป็นเซนติเมตร)
- แสดงดัชนีมวลกาย ซึ่งคำนวณได้จากสูตร

$$BMI = \frac{weight(kg.)}{height^2(m^2)}$$

- หุ่นดีต้องมี BMI ระหว่าง 18.5 – 25
น้อยไปผอม มากไปอ้วน

ตัวอย่าง : ดัชนีมวลกาย

```
import java.util.Scanner;
// โปรแกรมคำนวณดัชนีมวลกาย
public class BodyMassIndex {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("น้ำหนัก (kg.) = ");
        double weight = kb.nextDouble();
        System.out.print("ความสูง (cm.) = ");
        double height = kb.nextDouble();
        double hm = height / 100.0;
        double bmi = weight / (hm * hm);
        System.out.println("ดัชนีมวลกาย = " + bmi);
    }
}
```

การใช้ตัวแปร

```
import java.util.Scanner;
// โปรแกรมคำนวณดัชนีมวลกาย
public class BodyMassIndex {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("น้ำหนัก (kg) = ");
        double weight = kb.nextDouble();
        System.out.print("ความสูง (cm.) = ");
        double height = kb.nextDouble();
        double hm = height / 100;
        double bmi = weight / (hm * hm);
        System.out.println("ดัชนีมวลกาย = " + bmi);
    }
}
```

ตัวแปรเสริม

ตัวแปรเก็บข้อมูลขาเข้า

ตัวแปรเก็บผลลัพธ์

ตัวอย่าง : การหารากที่สอง

- หารากที่สองของ a
- ใช้ Babylonian method

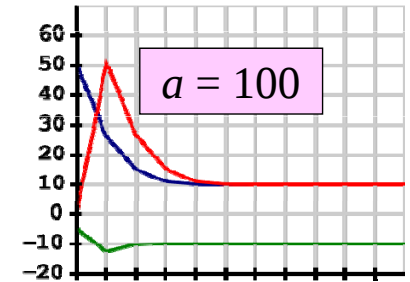
$$x_0 \approx \sqrt{a}$$

$$x_1 = (x_0 + a/x_0) / 2$$

$$x_2 = (x_1 + a/x_1) / 2$$

$$x_3 = (x_2 + a/x_2) / 2$$

...



$x_0 = -5.0$
 $x_1 = -12.5$
 $x_2 = -10.25$
 $x_3 = -10.00$

$x_0 = 50.0$
 $x_1 = 26.0$
 $x_2 = 14.92$
 $x_3 = 10.81$
 $x_4 = 10.03$
 $x_5 = 10.00$

$x_0 = 1.0$
 $x_1 = 50.5$
 $x_2 = 26.24$
 $x_3 = 15.03$
 $x_4 = 10.84$
 $x_5 = 10.03$
 $x_6 = 10.00$

ลองใช้ Excel

$$x_1 = (x_0 + a/x_0) / 2$$

$$x_2 = (x_1 + a/x_1) / 2$$

$$x_3 = (x_2 + a/x_2) / 2$$

...

a	x			
100	1.00			
	50.50			
	26.24			
	15.03			
	10.84			
	10.03			
	10.00			
	10.00			
	10.00			
	10.00			
	10.00			

โปรแกรม : การหารากที่สอง

```
public class SquareRoot{
    public static void main(String[] args) {
        double a = 100;
        double x0 = 1;
        double x1 = (x0 + (a / x0)) / 2.0; // 50.5
        double x2 = (x1 + (a / x1)) / 2.0; // 26.24
        double x3 = (x2 + (a / x2)) / 2.0; // 15.03
        double x4 = (x3 + (a / x3)) / 2.0; // 10.84
        double x5 = (x4 + (a / x4)) / 2.0; // 10.03
        double x6 = (x5 + (a / x5)) / 2.0; // 10.00
        System.out.println("รากที่สองของ " + a + " = " + x6);
    }
}
```

java SquareRoot

รากที่สองของ 100.0 = 10.000052895642693

$$x_k = \frac{1}{2} \left(x_{k-1} + \frac{a}{x_{k-1}} \right)$$

โปรแกรม : การหารากที่สอง (ปรับปรุง)

```
double a = 100;
double x0 = 1;
double x1 = (x0 + (a / x0)) / 2.0;
double x2 = (x1 + (a / x1)) / 2.0;
double x3 = (x2 + (a / x2)) / 2.0;
double x4 = (x3 + (a / x3)) / 2.0;
double x5 = (x4 + (a / x4)) / 2.0;
double x6 = (x5 + (a / x5)) / 2.0;
```

ใช้ตัวแปร 8 ตัว

```
double a = 100;
double x0 = 1, x1;
x1 = (x0 + (a / x0)) / 2.0;
x0 = (x1 + (a / x1)) / 2.0;
x1 = (x0 + (a / x0)) / 2.0;
x0 = (x1 + (a / x1)) / 2.0;
x1 = (x0 + (a / x0)) / 2.0;
x0 = (x1 + (a / x1)) / 2.0;
```

ใช้ตัวแปร 3 ตัว

```
double a = 100;
double x = 1;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
x = (x + (a / x)) / 2.0;
```

ใช้ตัวแปร 2 ตัว

x ได้รับค่าใหม่

ค่าของ x ปัจจุบัน

ฟังก์ชันทางคณิตศาสตร์

• ใช้บริการจากคลาส Math

```
double s = 1, t = 2;
double radian, degree;
radian = Math.toRadians(90); // องศาเป็นเรเดียน
degree = Math.toDegrees(radian); // เรเดียนเป็นองศา
t = Math.sin(radian); // ฟังก์ชันไซน์ (sine)
s = Math.cos(radian); // ฟังก์ชันโคไซน์ (cosine)
s = Math.sqrt(100.0); // ฟังก์ชันหารากที่สอง
s = Math.abs(s - t); // ฟังก์ชันหาค่า absolute
s = Math.max(s, t); // คืนค่ามากกว่าระหว่าง s กับ t
s = Math.min(s, t); // คืนค่าน้อยกว่าระหว่าง s กับ t
s = Math.random(); // คืนค่าสุ่มระหว่าง [0.0, 1.0)
```

ตัวอย่าง : รากของ $ax^2+bx+c=0$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$x_1 = \left(-b + \sqrt{b^2 - 4ac} \right) / (2a)$$

$$x1 = ((-b) + \text{Math.sqrt}(b*b - 4*a*c)) / (2*a)$$

$$x1 = (-b + \text{Math.sqrt}(b*b - 4*a*c)) / 2*a$$

ผิด

ตัวอย่าง : รากของ $ax^2+bx+c=0$

```
import java.util.Scanner;
public class QuadraticEqn {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("a = ");
        double a = kb.nextDouble();
        System.out.print("b = ");
        double b = kb.nextDouble();
        System.out.print("c = ");
        double c = kb.nextDouble();
        double t = Math.sqrt(b*b - 4*a*c);
        double x1 = (-b + t) / (2*a);
        double x2 = (-b - t) / (2*a);
        System.out.println("x1 = " + x1 + ", x2 = " + x2);
    }
}
```

$$2x^2 + 5x + 2 = 0$$

$$(2x+1)(x+2) = 0$$

```
JLab>java QuadraticEqn
a = 2
b = 5
c = 2
x1 = -0.5, x2 = -2.0
```

ตัวอย่าง : เซลเซียส → ฟาเรนไฮต์

$$f = \frac{9}{5}c + 32$$

```
import java.util.Scanner;
public class Celsius2Fahrenheit {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("c = ");
        double c = kb.nextDouble();
        double f = (9/5)*c + 32;
        System.out.println("อุณหภูมิ " + c +
            " เซลเซียส = " + f + " ฟาเรนไฮต์");
    }
}
```

โปรแกรมนี้ถูกหรือผิด ?

ต้องระวังเรื่องประเภท + ลำดับการทำงาน

$$f = \frac{9}{5}c + 32$$

c = 100.0;

f = 9 * c / 5 + 32;

f =

f = c / 5 * 9 + 32;

f =

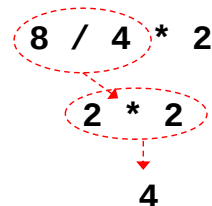
f = 9 / 5 * c + 32;

f =

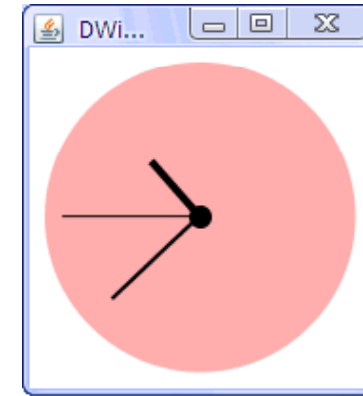
สรุปลำดับการคำนวณ

- ทำในวงเล็บทำก่อน : $-7 + 2 * \text{Math.sqrt}(5+4)$
- การเรียกใช้ method : $-7 + 2 * \text{Math.sqrt}(9)$
- ดัดลบ : $-7 + 2 * 3$
- คูณหาร * / : $-7 + 2 * 3$
- บวก ลบ + - : $-7 + 6$
- = : -1

ในกรณีมี operators
หลายตัวในระดับเดียวกัน
ให้ทำจากซ้ายไปขวา



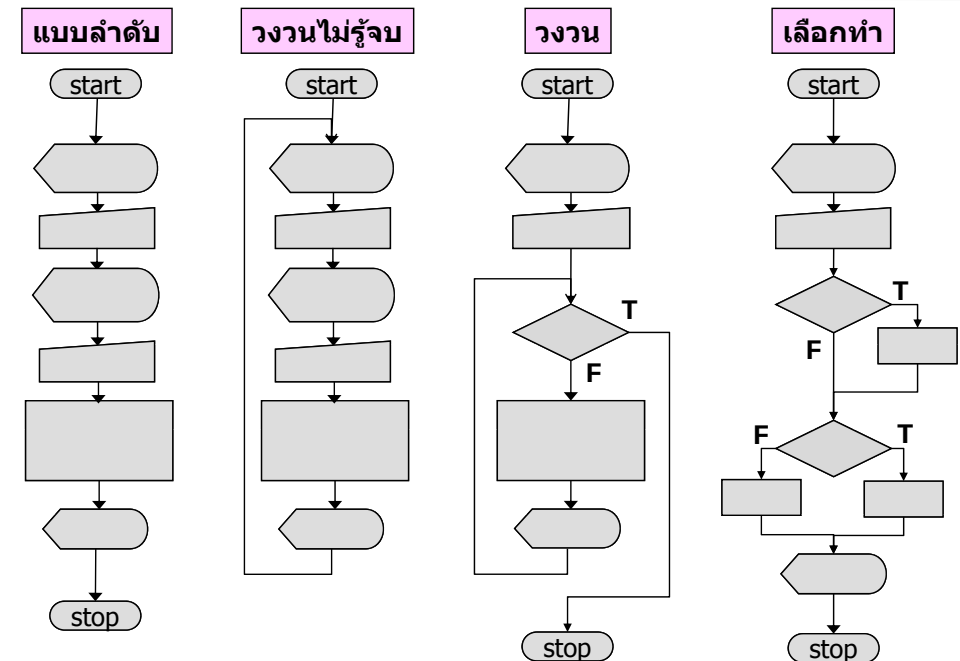
การทำงานแบบวงวนและเลือกทำ



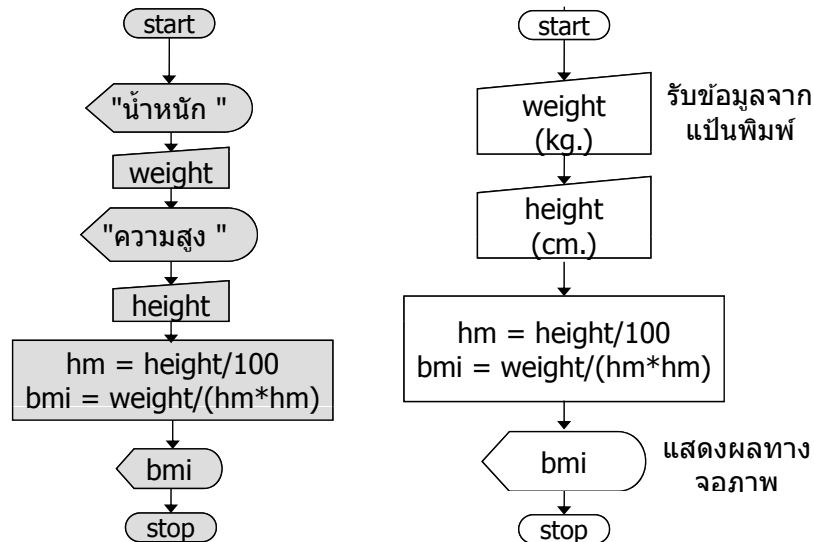
หัวข้อ

- ผังงาน
- วงวนไม่รู้จบ : `while(true)`
- ตัวดำเนินการ : `++ -- % < == > !=`
- การกระโดดออกจากวงวน : `if (...) break;`
- วงวนมาตรฐาน : `while`
- การเลือกทำ : `if, if-else`
- ตัวอย่าง ๆ ๆ

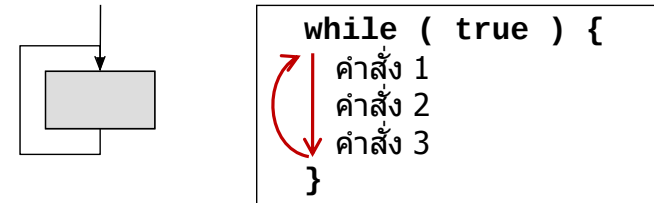
ผังงาน (Flowchart)



ผังงาน : บรรยายการทำงานหลัก



วงวนไม่รู้จบ : while(true) { ... }



```

public static void main(String[] args) {
    int k = 0;
    while (true) {
        System.out.println("k = " + k);
        k = k + 1;
    }
}

```

ดัชนีมวลกาย

```

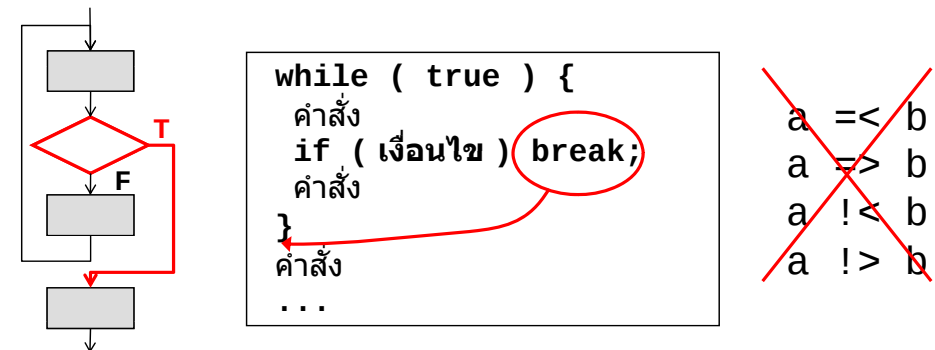
import java.util.Scanner;
public class BodyMassIndex {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        while ( true ) {
            System.out.print("น้ำหนัก (kg.) = ");
            double weight = kb.nextDouble();
            System.out.print("ความสูง (cm.) = ");
            double height = kb.nextDouble();
            double hm = height / 100.0;
            double bmi = weight / (hm * hm);
            System.out.println("ดัชนีมวลกาย = " + bmi);
        }
    }
}

```

$$BMI = \frac{\text{weight (kg.)}}{\text{height}^2 (m^2)}$$

ทำงานในวงวนไม่รู้จบ
ผู้ใช้ต้องหยุดโปรแกรมเอง

การกระโดดออกจากวงวน



บรรยายเงื่อนไขด้วย นิพจน์ตรรกะ (Boolean expression)
ใช้การเปรียบเทียบในการกำหนดเงื่อนไข
ด้วยตัวเปรียบเทียบข้อมูล < > <= >= != ==

if (a == b) break;

ผลการเปรียบเทียบจะได้ค่า true หรือ false

ดัชนีมวลกาย : เลิกโปรแกรมเมื่อน้ำหนักเป็นลบ

```
import java.util.Scanner;
public class BodyMassIndex {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        while ( true ) {
            System.out.print("น้ำหนัก (kg.) = ");
            double weight = kb.nextDouble();
            if (weight < 0) break;
            System.out.print("ความสูง (cm.) = ");
            double height = kb.nextDouble();
            double hm = height / 100.0;
            double bmi = weight / (hm * hm);
            System.out.println("ดัชนีมวลกาย = " + bmi);
        }
    }
}
```

ตัวอย่าง : แสดงเลขนับขึ้น

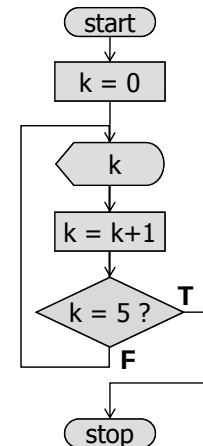
```
public static void main(String[] args) {
    int k = 0;
    while (true) {
        System.out.println("k = " + k);
        k = k + 1;
        if (k == 5) break;
    }
}
```

โปรแกรมนี้แสดงอะไร

k = 1
k = 2
k = 3
k = 4
k = 5

k = 0
k = 1
k = 2
k = 3
k = 4
k = 5

k = 0
k = 1
k = 2
k = 3
k = 4



เขียนได้หลายแบบ

```
int k = 0;
while (true) {
    System.out.println("k = " + k);
    k = k + 1;
    if (k == 5) break;
}
```

```
int k = 0;
while (true) {
    if (k == 5) break;
    System.out.println("k = " + k);
    k = k + 1;
}
```

```
int k = 0;
while (true) {
    if (k >= 5) break;
    System.out.println("k = " + k);
    k = k + 1;
}
```

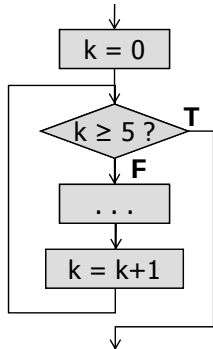
ใช้บ่อย : เพิ่มอีกหนึ่ง (++) ลดลงหนึ่ง (--)

- `c = c + 1;` `c += 1;` `c++;` `++c;`
- `c = c - 1;` `c -= 1;` `c--;` `--c;`
- ++ กับ -- ใช้กับตัวแปรเท่านั้น
(a + b)++ ไม่ได้

```
int k = 0;
while (true) {
    if (k >= n) break;
    // ...
    k++;
}
```

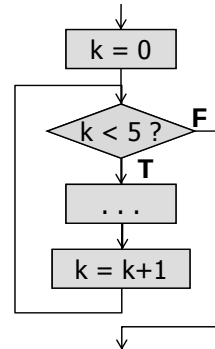
วงวนแบบ while

```
int k = 0;
while (true) {
    if (k >= 5) break;
    ...
    k++;
}
```



```
int k = 0;
while (k < 5) {
    ...
    k++;
}
```

ส่วนใหญ่เขียนแบบนี้



ตัวอย่าง : ค่าเฉลี่ยของข้อมูล 10 ตัว

```
import java.util.Scanner;
public class Average {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        double sum = 0;
        int k = 0;
        while (k < 10) {
            System.out.print("ข้อมูล = ");
            double data = kb.nextDouble();
            sum = sum + data;
            k = k + 1;
        }
        double average = sum / 10.0;
        System.out.println("ค่าเฉลี่ย = " + average);
    }
}
```

ตัวอย่าง : ค่าเฉลี่ยของข้อมูล n ตัว

```
import java.util.Scanner;
public class Average {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("จำนวนข้อมูล = ");
        int n = kb.nextInt();
        double sum = 0;
        int k = 0;
        while (k < n) {
            System.out.print("ข้อมูล = ");
            double data = kb.nextDouble();
            sum = sum + data;
            k = k + 1;
        }
        double average = sum / n;
        System.out.println("ค่าเฉลี่ย = " + average);
    }
}
```

ตัวอย่าง : ค่าเฉลี่ยของชุดข้อมูล (เลิกเมื่อพบลบ)

```
import java.util.Scanner;
public class Average {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        double sum = 0;
        int k = 0;
        while (true) {
            System.out.print("ข้อมูล = ");
            double data = kb.nextDouble();
            if (data < 0) break;
            sum = sum + data;
            k = k + 1;
        }
        double average = sum / k;
        System.out.println("ค่าเฉลี่ย = " + average);
    }
}
```

ตัวอย่าง : หาคากที่สอง (บาบิโลน) แบบใช้วงวน

```
public static void main(String[] args) {
    Scanner kb = new Scanner(System.in);
    System.out.print("a = ");
    double a = kb.nextDouble();
    double x = 1.0;
    while (true) {
        x = (x + a/x) / 2.0;
        if ( Math.abs(x*x - a) <= 1e-5 ) break;
    }
    System.out.println("รากที่สอง = " + x );
}
```

$$x_k = \frac{1}{2} \left(x_{k-1} + \frac{a}{x_{k-1}} \right)$$

$$|x^2 - a| < 10^{-5}$$

ต้องระวังเรื่องความเท่ากันของ double
เช่น $0.1+0.1+0.1$ ไม่เท่ากับ 0.3
 $(0.1+0.1+0.1 - 0.3) \approx 5.6 \times 10^{-17}$

double : ละเอียดยแต่ไม่ 100%

```
public static void main(String[] args) {
    double a = 0.1;
    double b = a + a + a;
    System.out.println(b - 0.3);
}
```

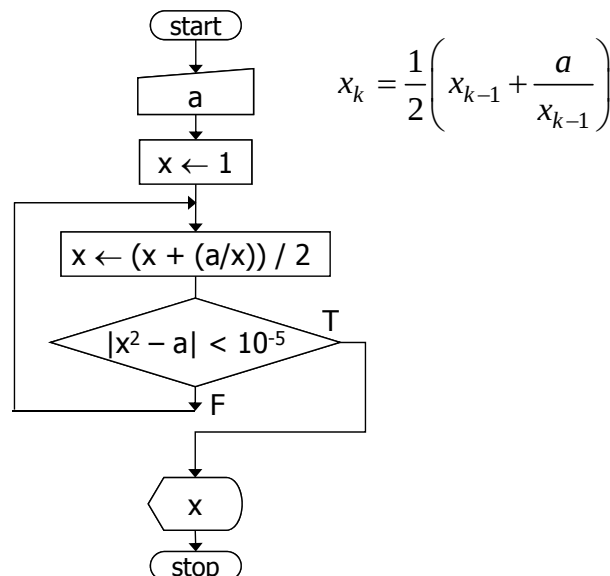
5.551115123125783E-17

- คอมพิวเตอร์เก็บ double หนึ่งตัวด้วยเนื้อที่ 64 บิต
- มีขนาดจำกัด จึงมีบางค่าเก็บได้ไม่แม่นยำ 100%
- จึงต้องระวังการเปรียบเทียบความเท่ากันของ double
- ควรใช้การเปรียบเทียบการมีค่าใกล้เคียงกันมาก ๆ แทน

$$|a - b| < 10^{-10}$$

$$|a - b| / \max(|a|, |b|) < 10^{-10}$$

ผังงาน : การหาคากที่สองของ a



$$x_k = \frac{1}{2} \left(x_{k-1} + \frac{a}{x_{k-1}} \right)$$

ปรับวงวนของการหาคากที่สอง

```
while (true) {
    x = (x + a/x) / 2.0;
    if (Math.abs(x*x - a) <= 1e-5) break;
}
```

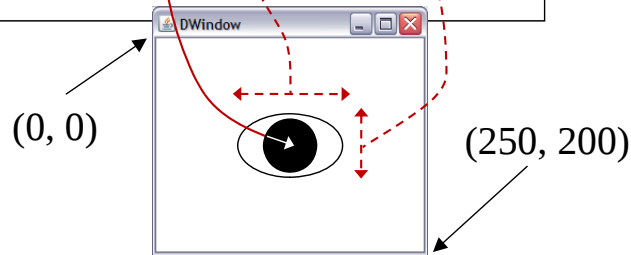
```
while (true) {
    if (Math.abs(x*x - a) <= 1e-5) break;
    x = (x + a/x) / 2.0;
}
```

```
while (Math.abs(x*x - a) > 1e-5) {
    x = (x + a/x) / 2.0;
}
```

DWindow

คลาสเสริมของ JLab ช่วยวาดเส้น วงรี สีเหลี่ยม

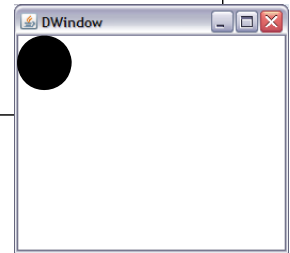
```
import jlab.graphics.DWindow;
public class Eye {
    public static void main(String[] args) {
        DWindow w = new DWindow(250, 200);
        w.drawEllipse(125, 100, 100, 60);
        w.fillEllipse(125, 100, 50, 50);
    }
}
```



ลูกบอลเคลื่อนที่

```
import jlab.graphics.DWindow;
public class MovingBall {
    public static void main(String[] args) {
        DWindow w = new DWindow(250, 200);
        double x = 0, y = 0;
        while (true) {
            w.fade(1); // ภาพในวินโดว์จางลง 100% (ลบหมด)
            w.fillEllipse(x, y, 50, 50);
            x = x + 2; // ปรับตำแหน่ง x เพื่อใช้ในรอบถัดไป
            y = y + 1; // ปรับตำแหน่ง y เพื่อใช้ในรอบถัดไป
            w.sleep(100); // หยุดการทำงาน 100msec.
        }
    }
}
```

ลอง w.fade(0.1) ดู



ตัวดำเนินการเสริม

- เศษจากการหาร : %

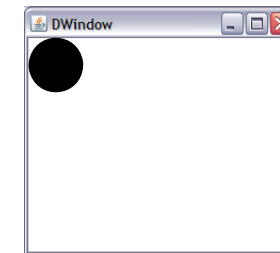
5	%	2	ได้	1
5.2	%	2	ได้	1.2

```
import jlab.graphics.DWindow;
public class MovingBall {
    public static void main(String[] args) {
        DWindow w = new DWindow(250, 200);
        double x = 0, y = 0;
        while (true) {
            w.fade(1);
            w.fillEllipse(x, y, 50, 50);
            x = (x + 2) % 250;
            y = (y + 1) % 200;
            w.sleep(100);
        }
    }
}
```

ต้องการให้ลูกบอลออกขวา
วนกลับมาทางซ้าย

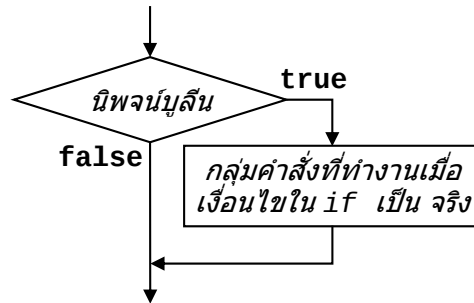
```
x = 246
x = (246 + 2) % 250 = 248 % 250 = 248
x = (248 + 2) % 250 = 250 % 250 = 0
```

ทำอย่างไรให้ลูกบอลแต่งขอบวินโดว์



ต้องใช้คำสั่งทดสอบ
ให้กลับทิศการเคลื่อนที่เมื่อชนขอบ

if : จริงถึงทำ



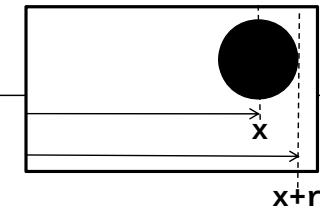
```

if ( นิพจน์ที่ได้ค่า จริง หรือ เท็จ ) {
    กลุ่มคำสั่งที่ทำงานเมื่อเงื่อนไขใน if เป็นจริง
}
  
```

ลูกบอลแตงผนัง

```

public static void main(String[] args) {
    DWindow w = new DWindow(250, 200);
    double x = 0, y = 0, dx = 2, dy = 1, r = 25;
    while (true) {
        w.fade(1);
        w.fillEllipse(x, y, 2*r, 2*r);
        x = x + dx;
        y = y + dy;
        if (x + r > 250) { dx = -dx; }
        if (x - r < 0) { dx = -dx; }
        if (y + r > 200) { dy = -dy; }
        if (y - r < 0) { dy = -dy; }
        w.sleep(100);
    }
}
  
```



ตรวจสอบความถูกต้องก่อนทำงาน

```

public static void main(String[] args) {
    Scanner kb = new Scanner(System.in);
    System.out.print("a = ");
    double a = kb.nextDouble();
    if (a < 0) {
        System.out.println("a ต้องไม่เป็นลบ ");
        System.exit(-1);
    }
    double x = 1.0;
    while (Math.abs(x*x - a) > 1e-5) {
        x = (x + a/x) / 2.0;
    }
    System.out.println("รากที่สอง = " + x );
}
  
```

ใช้ return; ก็ได้

ถ้า a เป็นลบ $|x*x - a|$ มีค่า $(x*x + |a|)$, วนไม่รู้จบ

ตัวอย่าง : วันใดของสัปดาห์

$$w = \left(d + \left\lfloor \frac{26(m+1)}{10} \right\rfloor + k + \left\lfloor \frac{k}{4} \right\rfloor + \left\lfloor \frac{c}{4} \right\rfloor + 5c \right) \bmod 7$$

$$k = y \bmod 100, c = \left\lfloor \frac{y}{100} \right\rfloor$$

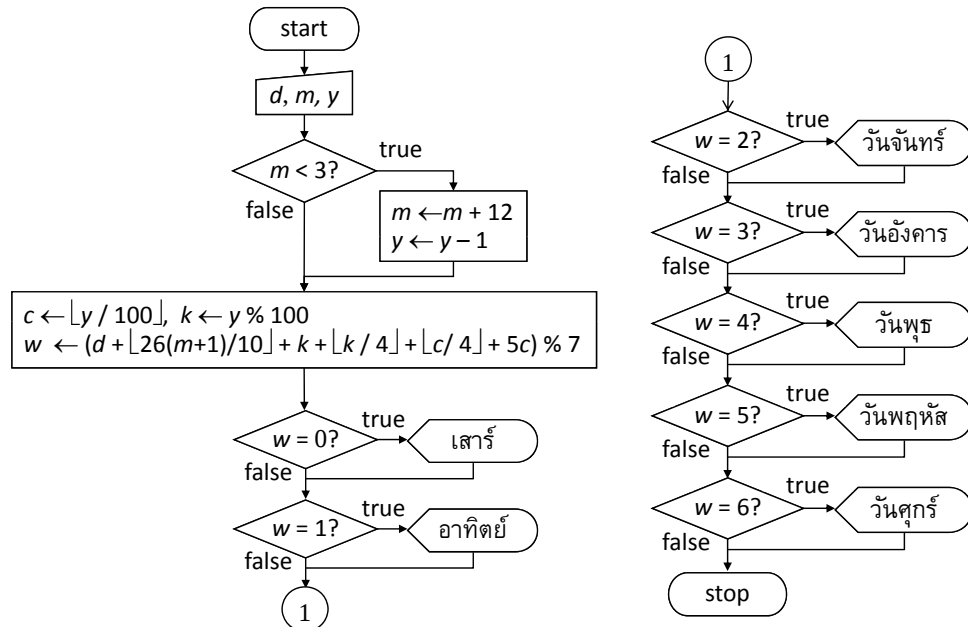
y คือเลขปี ค.ศ ส่วน m คือเลขเดือน = 3, 4, ..., 13, 14
 แปลงเล็กน้อย m = 13 แทนมกราคม และ 14 แทนกุมภาพันธ์
 ถ้าเป็นสองเดือนพิเศษนี้ให้ลดค่าปีลง 1 ด้วย

ตัวอย่าง : วันที่ 2 มิถุนายน ปี ค.ศ. 2009 เป็นวันอะไร ?

d = 2, m = 6, y = 2009
 k = 9, c = 20
 $w = (2 + 18 + 9 + 2 + 5 + 100) \% 7$
 $= 136 \% 7$
 $= 3$

0 - เสาร์, 1 - อาทิตย์, 2 - จันทร์, ..., 6 - ศุกร์
 วันที่ 2 มิถุนายน ปี ค.ศ. 2009 คือ วันอังคาร

ผังงาน : วันใดของสัปดาห์

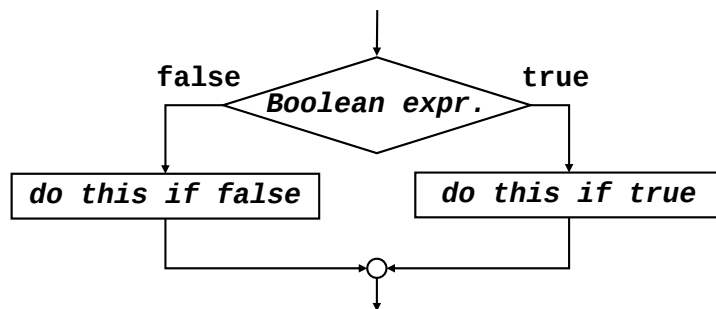


โปรแกรม : วันใดของสัปดาห์

```

Scanner kb = new Scanner(System.in);
System.out.print("วัน = ");
int d = kb.nextInt();
System.out.print("เลขเดือน = ");
int m = kb.nextInt();
System.out.print("ปี(ค.ศ.) = ");
int y = kb.nextInt();
if (m < 3) {
    m = m + 12;
    y = y - 1;
}
int c = y / 100, k = y % 100;
int w = (d + 26*(m+1)/10 + k + k/4 + c/4 + 5*c) % 7;
if (w == 0) { System.out.println("เสาร์"); }
if (w == 1) { System.out.println("อาทิตย์"); }
if (w == 2) { System.out.println("จันทร์"); }
if (w == 3) { System.out.println("อังคาร"); }
if (w == 4) { System.out.println("พุธ"); }
if (w == 5) { System.out.println("พฤหัสบดี"); }
if (w == 6) { System.out.println("ศุกร์"); }
  
```

if-else : จริงทำอย่าง เท็จทำอย่าง



```

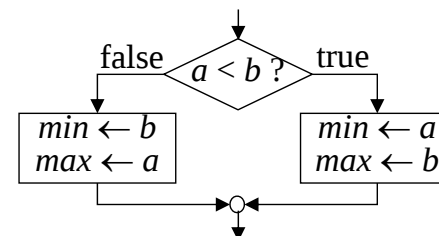
if ( นิพจน์ที่ได้ค่า จริง หรือ เท็จ ) {
    กลุ่มคำสั่งที่ทำงานเมื่อเงื่อนไขใน if เป็น จริง
} else {
    กลุ่มคำสั่งที่ทำงานเมื่อเงื่อนไขใน if เป็น เท็จ
}
  
```

ตัวอย่าง : ตัวน้อย ตัวมาก

- มีข้อมูลในตัวแปร a กับ b
- ต้องการให้ตัวแปร
 - min เก็บตัวน้อยของ a กับ b
 - max เก็บตัวมากของ a กับ b

```

min = a;
max = b;
if (a > b) {
    min = b;
    max = a;
}
  
```



```

if (a < b) {
    min = a;
    max = b;
} else {
    min = b;
    max = a;
}
  
```

ตัวอย่าง : หาค่าประมาณของ π ที่ดีกว่า $22/7$

- $|\pi - 22/7| \approx 0.001264...$
- หาค่า n และ d ที่ทำให้ $|\pi - n/d| < 10^{-6}$

n/d $\pi - (n+1)/d$ $\pi - n/(d+1)$

22/7 0.144 0.392

23/7 0.287

23/8 0.142

24/8 0.017

25/8 0.108

26/8 0.233

27/8 0.358

27/9 0.030

28/9 0.081

...

353/112 0.019

353/113 0.009

354/113 0.000

0.392

เลือกตัวน้อย

0.475

0.364

0.253

0.142

0.442

0.342

...

0.018

0.045

0.036

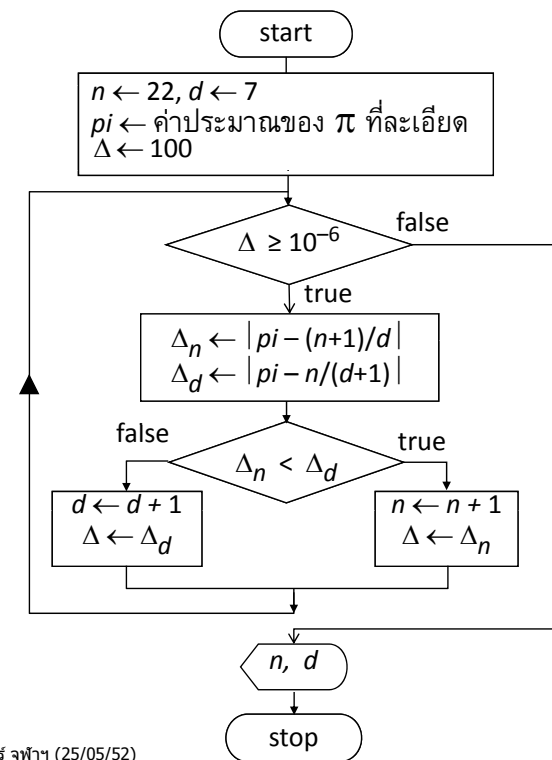


祖冲之

พบ 355/113

เมื่อปี ค.ศ. 480

355/113



โปรแกรม : หาค่าประมาณของ π ที่ดีกว่า $22/7$

```

public static void main(String[] args) {
    double n = 22, d = 7;
    double pi = Math.PI; // 3.141592653589793;
    double diff = 100, diffN, diffD;
    while (diff >= 1e-6) {
        diffN = Math.abs(pi - (n + 1) / d);
        diffD = Math.abs(pi - n / (d + 1));
        if (diffN < diffD) {
            n = n + 1;
            diff = diffN;
        } else {
            d = d + 1;
            diff = diffD;
        }
    }
    System.out.println(n + "/" + d + " = " + (n/d));
}
    
```

กลุ่มคำสั่ง (Block)

- Block คือกลุ่มคำสั่ง ครอบด้วย { }
- ถ้ามีคำสั่งเดียวหลัง if, else, while ไม่ต้องใส่ใน block ก็ได้

```

if (diffN < diffD) {
    n = n + 1;
    diff = diffN;
} else {
    d = d + 1;
    diff = diffD;
}
    
```

```

if (x < 0) {
    x = -x;
}
    
```

```

if (x < 0)
    x = -x;
    
```

```

if (x < 0) x = -x;
    
```

```

if (a < b)
    min = a;
else
    min = b;
    
```

```

while (Math.abs(x*x - a) > 1e-5)
    x = (x + a/x) / 2.0;
    
```

ตัด { } หลัง if, else, while ต้องระวัง

```
Scanner kb = new Scanner(System.in);
System.out.print("วัน = ");
int d = kb.nextInt();
System.out.print("เลขเดือน = ");
int m = kb.nextInt();
System.out.print("ปี(ค.ศ.) = ");
int y = kb.nextInt();
if (m < 3) {
    m = m + 12; if (m < 3) m = m + 12; y = y - 1;
    y = y - 1;
}
int c = y / 100, k = y % 100;
int w = (d + 26*(m+1)/10 + k + k/4 + c/4 - 2*c) % 7;
if (w == 0) System.out.println("เสาร์");
if (w == 1) System.out.println("อาทิตย์");
if (w == 2) System.out.println("จันทร์");
if (w == 3) System.out.println("อังคาร");
if (w == 4) System.out.println("พุธ");
if (w == 5) System.out.println("พฤหัสบดี");
if (w == 6) System.out.println("ศุกร์");
```

ตัวอย่าง : นิพจน์บูลีน

- ถ้า a เป็นจำนวนคี่

```
if (a % 2 == 1)
```

- ถ้าหลักสิบของ a เป็นเลข 7

```
if (((int)a/10)%10 == 7)
```

- ถ้า a มีค่าตั้งแต่ 1 ถึง 100

```
if (1 <= a && a <= 100)
```

$$1 \leq a \leq 100$$

$$1 \leq a \text{ และ } a \leq 100$$

จะนำเสนอเรื่อง "และ"
"หรือ" ในโอกาสต่อไป

ตัวอย่าง : นิพจน์บูลีน

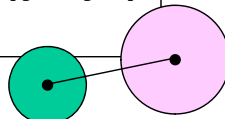
- ถ้าจุดสองจุดบนระนาบสองมิติห่างกันเกิน 10

```
if ((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2) > 100)
```

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} > 10$$

- ถ้าวงกลมสองวงแตะหรือทับกัน

```
if ((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2)
    <=(r1+r2)*(r1+r2))
```



- ถ้าโยนเหรียญแล้วได้ "ก้อย"

```
if (Math.random() < 0.5)
```

การประมวลผลข้อความ

การเข้ารหัสลับ Rot13

I see trees of green red roses too I see em bloom for me and for you And I think to myself what a wonderful world	 	V frr gerrf bs terra erq ebfrf gbb V frr rz oybbz sbe zr naq sbe lbh Naq V guvax gb zlfrys jung n jbaqreshy jbeyq
--	--	--

A	B	C	D	E	F	G	H	I	J	K	L	M
↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

หัวข้อ

- การอ่านสตริงจากแป้นพิมพ์
- บริการของสตริง
- วงวนแบบ for
- การอ่านเขียนแฟ้มข้อความ
- ตัวอย่าง

nextLine() : อ่าน String จากแป้นพิมพ์

```
import java.util.Scanner;

public class ReadString1 {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("ชื่อ = ");
        String name = kb.nextLine();
        System.out.println("สวัสดีค่ะ " + name);
    }
}
```

JLab>java ReadString1
ชื่อ = จาวา แซ่ซัน
สวัสดีค่ะ จาวา แซ่ซัน

สตริง + ? ก็ได้สตริง

- สตริง + สตริง คือการต่อสตริง "AB" + "C" ได้ "ABC"
- สตริง + จำนวน หรือ จำนวน + สตริง จะแปลงจำนวนเป็นสตริงก่อน แล้วต่อ
 - "A" + 2 ได้ "A2"
 - 2.5 + "A" ได้ "2.5A"
 - "A" + (4+2) ได้ "A6"
 - "A" + 4 + 2 ได้ "A42"
 - 2 + "A" + 4 ได้ "2A4"
 - 2 + 4 + A ได้ "6A"
- * / - % ใช้กับสตริงไม่ได้
 - "2" * 1 ผิด

สตริง = ลำดับของอักขระ



บริการของ String

สตริง.ชื่อเมทอด() method

สตริง.length()

คืนความยาวของสตริง

```
String s = "12345";  
int a = s.length();  
a = "Hello".length();
```

บริการของ String

สตริง.trim()

คืนสตริงใหม่เหมือนเก่า
แต่ตัดช่องว่างซ้ายขวาทิ้ง

```
String s = " ab c ";  
String t = s.trim();  
// t เก็บ "ab c",  
// s ไม่เปลี่ยน
```

บริการของ String

สตริง.toUpperCase()

คืนสตริงใหม่เหมือนเก่า
แต่เปลี่ยนเป็นตัวอังกฤษใหญ่หมด

```
String s = "HeLLo";  
String t = s.toUpperCase();  
// t เก็บ "HELLO",  
// s ไม่เปลี่ยน
```

บริการของ String

สตริง.toLowerCase()

คืนสตริงใหม่เหมือนเก่า
แต่เปลี่ยนเป็นตัวอังกฤษเล็กหมด

```
String s = "HeLLo";  
String t = s.toLowerCase();  
// t เก็บ "hello",  
// s ไม่เปลี่ยน
```

บริการของ String

สตริง.substring(i,j)

คืนสตริงใหม่ซึ่งเป็นสตริงย่อย
ตั้งแต่ตัวที่ i ถึงตัวที่ j-1

```
String s = "hello";  
           01234  
String t = s.substring(1,4);  
// t เก็บ "ell",  
// s ไม่เปลี่ยน
```

บริการของ String

สตริง1.indexOf(สตริง2)

คืนดัชนีของสตริง1 ที่มีข้อความเหมือนกับสตริง2

```
String s = "hello hello";  
           01234  
int k = s.indexOf("el");
```

บริการของ String

สตริง1.equals(สตริง2)

คืน true ถ้าสตริง1 เหมือนสตริง2

```
String ans = kb.nextLine();  
if (ans.equals("yes")) {
```

บริการของ String

สตริง1.equalsIgnoreCase(สตริง2)

คืน true ถ้าสตริง1 เหมือนสตริง2
แบบไม่สนใจตัวอักษรใหญ่เล็ก

```
String ans = kb.nextLine();  
if (ans.equalsIgnoreCase("yes")) {  
  
    yes, Yes, yEs, YEs,  
    yeS, YeS, yES, YES
```

สรุปบริการของ String

"12345".length()	→	5
" a bc ".trim()	→	"a bc"
"0123".substring(1,3)	→	"12"
"abcbc".indexOf("bc")	→	1
"AbC2".toUpperCase()	→	"ABC2"
"AbC2".toLowerCase()	→	"abc2"
"AB".equals("Ab")	→	false
"AB".equalsIgnoreCase("Ab")	→	true

ตัวอย่าง : นับช่องว่างในสตริง

```
import java.util.Scanner;  
  
public class CountBlank {  
    public static void main(String[] args) {  
        Scanner kb = new Scanner(System.in);  
        System.out.print("สตริง = ");  
        String t = kb.nextLine();  
        int count = 0;  
        int k = 0;  
        while (k < t.length()) {  
            if (t.substring(k, k+1).equals(" ")) count++;  
            k++;  
        }  
        System.out.println("มีช่องว่าง " + count + " ตัว");  
    }  
}
```

```
JLab>java CountBlank  
สตริง = VA JA JA VA  
มีช่องว่าง 3 ตัว
```

วงวนแบบ for

- เหมาะกับวงวนที่ทำเป็นจำนวนครั้งตามที่กำหนด
- มีตัวแปรเพิ่ม (หรือลด) ตามวงวน

```
int i = 0;
while ( i < n ) {
    // ...
    i++;
}
```

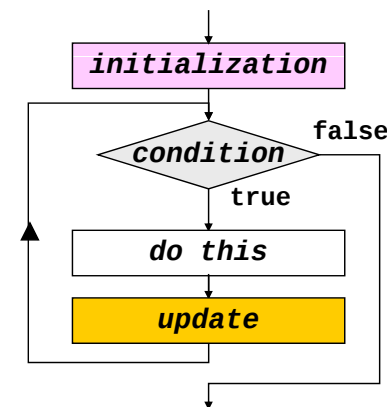
ค่าของ i ในวงวนคือ
0, 1, ..., n-1

```
int i;
for ( i = 0 ; i < n ; i++ ) {
    // ...
}
```

ใช้ตัวแปร i ได้เฉพาะในวงวน

```
for ( initialization ; condition ; update ) {
    do this
}
```

วงวนแบบ for



```
for ( initialization ; condition ; update ) {
    do this
}
```

ตัวอย่าง : วงวน for

```
for (int i = 0; i < 10; i++) {
    System.out.println(i);
}
```

```
for (int i = 10; i >= 0; i--) {
    System.out.println(i);
}
```

```
for (int i = 0; i < 11; i = i + 2) {
    System.out.println(i);
}
```

```
for (int i = 1; i < 11; i = i + 2) {
    System.out.println(i);
}
```

ตัวอย่าง : สตริงที่รับมามี "A" กี่ตัว

```
import java.util.Scanner;

public class CountBlank {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("สตริง = ");
        String t = kb.nextLine();
        int count = 0;
        for (int k = 0; k < t.length(); k++) {
            if (t.substring(k, k+1).equals(" ")) count++;
        }
        System.out.println("มี A " + count + " ตัว");
    }
}
```

JLab>java CountA
สตริง = JAVA Programming
มี A 2 ตัว

ทำอย่างไรจึงจะนับ
ทั้ง "A" และ "a" ?

Bar code มาตรฐาน EAN-13

- มาตรฐาน bar code ประกอบด้วยเลข 13 ตัว
- ตัวขวาสุดคือเลขตรวจสอบ (check digit)



หลักตำแหน่งคือ
ต้องคูณ 3

$$\begin{array}{r}
 9 \ 7 \ 8 \ 1 \ 9 \ 3 \ 2 \ 6 \ 9 \ 8 \ 1 \ 8 \\
 9+21+ 8+ 3+ 9+ 9+ 2+18+ 9+24+ 1+24 = 137 \\
 137 \% 10 = 7 \\
 10 - 7 = 3 \\
 3 \% 10 = 3
 \end{array}$$

Integer.parseInt(...), Double.parseDouble(...)

- "12" + "23" ได้ "1223" ไม่ได้ "35"
- ถ้ามีสตริงที่ภายในเป็นตัวเลข
- ต้องแปลงเป็น int หรือ double ก่อน ด้วย
 - Integer.parseInt(s)
 - Double.parseDouble(s)
- ตัวอย่าง
 - int a = Integer.parseInt("123");
 - double b = Double.parseDouble("123.45");

โปรแกรม : EAN-13

```

import java.util.Scanner;
public class EAN13 {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("ตัวเลข 12 หลัก = ");
        String d = kb.nextLine();
        int s = 0;
        for (int k = 0; k < 12; k++) {
            int dk = Integer.parseInt(d.substring(k, k+1));
            if (k % 2 == 0) {
                s = s + dk;
            } else {
                s = s + 3*dk;
            }
        }
        System.out.println("Check Digit คือ " + (10-s%10)%10);
    }
}
    
```

รับเลข 12 หลักเก็บใส่ int ไม่ได้
(เพราะมีค่าเกิน สองพันล้าน int รับไม่ไหว)

การอ่านเขียนแฟ้มข้อความ (text file)

- อ่านแฟ้มข้อความ

```

Scanner in =
    new Scanner(new File("c:/data-in.txt"));
String line = in.nextLine();
...
in.close();
    
```

มีการเปิดแฟ้มที่นี่

เลิกใช้ก็ปิดแฟ้ม

- เขียนแฟ้มข้อความ

```

PrintStream out =
    new PrintStream(new File("c:/data-out.txt"));
out.print( ... );
out.println( ... );
...
out.close();
    
```

การอ่านแฟ้มข้อความ (text file)

```
import java.util.Scanner;
import java.io.*;

public class ReadingTextFile {
    public static void main(String[] args)
        throws IOException {
        Scanner in = new Scanner(new File("data.txt"));
        for (int i=0; i<4; i++) {
            String s = in.nextLine();
            System.out.println(s);
        }
        in.close();
    }
}
```

การอ่านจนหมดแฟ้ม

```
import java.util.Scanner;
import java.io.*;

public class ReadingTextFile {
    public static void main(String[] args)
        throws IOException {
        Scanner in = new Scanner(new File("data.txt"));
        while ( in.hasNext() ) {
            String s = in.nextLine();
            System.out.println(s);
        }
        in.close();
    }
}
```

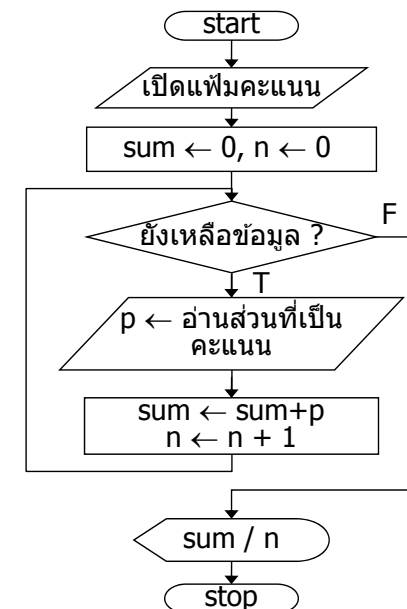
hasNext คืน true
ถ้ายังมีข้อมูลให้อ่าน

ตัวอย่าง : โปรแกรมทำสำเนาแฟ้มข้อความ

```
import java.util.Scanner;
import java.io.*;

public class WritingTextFile {
    public static void main(String[] args)
        throws IOException {
        Scanner in = new Scanner(new File("file1.java"));
        PrintStream out = new PrintStream(
            new File("file2.txt") );
        while (in.hasNext()) {
            String t = in.nextLine();
            out.println(t);
        }
        in.close();
        out.close();
    }
}
```

ผังงานการหาค่าเฉลี่ยคะแนนที่เก็บในแฟ้ม



5130120321	74.0
5130293921	85.5
5130294121	58.0
5130338421	90.2
...	

โปรแกรมหาค่าเฉลี่ยคะแนน

```
import java.util.Scanner;
import java.io.*;

public class AverageScore {
    public static void main(String[] args)
        throws IOException {
        Scanner in = new Scanner(new File("data.txt"));
        double sum = 0;
        int n = 0;
        while (in.hasNext()) {
            String id = in.next(); // อ่านสตริงมาหนึ่งก้อน (คั่นด้วยช่องว่าง)
            double p = in.nextDouble(); // อ่านเป็น double เลขก็ได้
            sum = sum + p;
            n++;
        }
        in.close();
        System.out.println("คะแนนเฉลี่ย = " + (sum / n));
    }
}
```

5130120321	74.0
5130293921	85.5
5130294121	58.0
5130338421	90.2
...	

21

29

โปรแกรมหาสถิติต่าง ๆ ของคะแนน

```
import java.util.Scanner;
import java.io.*;

public class ScoreStat {
    public static void main(String[] args) throws IOException {
        Scanner in = new Scanner(new File("data.txt"));
        double sum = 0.0, min = 1000.0, max = 0.0;
        int n = 0;
        while (in.hasNext()) {
            String id = in.next();
            double p = in.nextDouble();
            if (p < min) min = p;
            if (p > max) max = p;
            sum = sum + p;
            n++;
        }
        in.close();
        System.out.println("คะแนนเฉลี่ย = " + (sum / n));
        System.out.println("คะแนนน้อยสุด = " + min);
        System.out.println("คะแนนมากที่สุด = " + max);
    }
}
```

เริ่มต้นให้ min
มีค่ามาก

เริ่มต้นให้ max
มีค่าน้อย

5130120321	74.0
5130293921	85.5
5130294121	58.0
5130338421	90.2
...	

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (25/05/52)

30

โปรแกรมอ่านคะแนนแล้วให้เกรด

คะแนน	เกรด
$p \geq 80$	A
$70 \leq p < 80$	B
$60 \leq p < 70$	C
$50 \leq p < 60$	D
$p < 50$	F

input

5130120321	74.0
5130293921	85.5
5130294121	58.0
5130338421	90.2
...	

output

5130120321	74.0	B
5130293921	85.5	A
5130294121	58.0	D
5130338421	90.2	A

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (25/05/52)

31

โปรแกรมตัดเกรดเก็บลงแฟ้ม

```
import java.util.Scanner;
import java.io.*;

public class Grading {
    public static void main(String[] args) throws IOException {
        Scanner in = new Scanner(new File("data.txt"));
        PrintStream out = new PrintStream(new File("grade.txt"));
        while (in.hasNext()) {
            String id = in.next();
            double p = in.nextDouble();
            String grade = "";
            if (p >= 80) grade = "A";
            if (70 <= p && p < 80) grade = "B";
            if (60 <= p && p < 70) grade = "C";
            if (50 <= p && p < 60) grade = "D";
            if (p < 50) grade = "F";
            out.println(id + " " + p + " " + grade);
        }
        in.close();
        out.close();
    }
}
```

คะแนน	เกรด
$p \geq 80$	A
$70 \leq p < 80$	B
$60 \leq p < 70$	C
$50 \leq p < 60$	D
$p < 50$	F

สามารถเขียน if หลาย ๆ
บรรทัดนี้ได้หลายแบบ จะ
นำเสนอหลัง mid-term

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (25/05/52)

32

การเข้ารหัสลับ Rot13

I see trees of green
red roses too

V frr gerrf bs terra
erq ebfrf gbb

I see em bla
for me and

Julius Caesar ใช้วิธีนี้
มานานแล้ว แต่เป็น rot-3

And I think to myself
what a wonderful world

naq v gvaax gb zlfrys
jung n jbaqreshy jbeyq

A	B	C	D	E	F	G	H	I	J	K	L	M
↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Rotate-13

upperCase 01234567890123456789012345
ABCDEFGHIJKLMNOPQRSTUVWXYZ

```
8 int j = upperCase.indexOf("I");
V String c = upperCase.substring(j+13, j+14);

21 int j = upperCase.indexOf("V");
String c = upperCase.substring(j+13, j+14);

8 j = (j+13) % 26;
I c = upperCase.substring(j+13, j+14);
```

ดกขอบ

โปรแกรมเข้า/ถอดรหัสแบบ Rot-13

```
public static void main(String[] args) throws IOException {
    Scanner in = new Scanner(new File("data.txt"));
    PrintStream out = new PrintStream(new File("dataX.txt"));
    String upperCase = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    while (in.hasNext()) {
        String text = in.nextLine().toUpperCase();
        String rot13 = "";
        for (int k = 0; k < text.length(); k++) {
            String c = text.substring(k, k+1);
            int j = upperCase.indexOf(c);
            if (j >= 0) {
                j = (j + 13) % 26;
                rot13 = rot13 + upperCase.substring(j+13, j+14);
            } else {
                rot13 = rot13 + c;
            }
        }
        out.println( rot13 );
    }
    in.close(); out.close();
}
```

เพิ่มเติม : อักขระหลีก (escape char.)

ต้องการเก็บ พูดยว่า "สวัสดี"

• s = "พูดยว่า "สวัสดี"" ผิด

• s = "พูดยว่า \"สวัสดี\"" ถูก

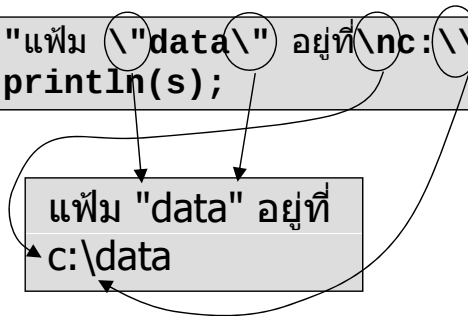
พ ู ด ว ่า " ส ว ั ส ด ี "

เราเรียก \" ที่เขียนในสตริงว่า escape character

อักขระหลีก (escape char.)

\"	แทน	"
\\	แทน	\
\n	แทน	การขึ้นบรรทัดใหม่
\t	แทน	รหัส tab
. . .		

```
String s = "แฟ้ม \"data\" อยู่ที่\nc:\\data";  
System.out.println(s);
```



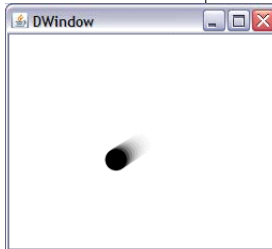
แถวลำดับ

หัวข้อ

- การประกาศ สร้าง และใช้แถวลำดับ
- การประมวลผลข้อมูลในอาเรย์
 - การค้นข้อมูล การเรียงลำดับข้อมูล ...
- ตัวอย่าง

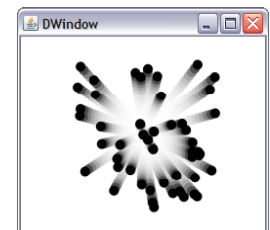
ลูกบอลหนึ่งลูก

```
import jlab.graphics.DWindow;
public class Ball1 {
    public static void main(String[] args) {
        DWindow w = new DWindow(250, 200);
        double x = 125, y = 100;
        double r = 5;
        double dx = -4 + 8*Math.random();
        double dy = -4 + 8*Math.random();
        while (true) {
            w.fade(0.3); // ทำวินโดว์ให้จางลง 30%
            w.fillEllipse(x, y, 2 * r, 2 * r); // วาดวงกลม
            x = x + dx; y = y + dy; // ปรับตำแหน่งในรอบต่อไป
            if (x + r > 250) dx = -dx; // ขนขอบขวา ?
            if (x - r < 0) dx = -dx; // ขนขอบซ้าย ?
            if (y + r > 200) dy = -dy; // ขนขอบล่าง ?
            if (y - r < 0) dy = -dy; // ขนขอบบน ?
            w.sleep(30);
        }
    }
}
```



ลูกบอล 50 ลูก

- หนึ่งลูกใช้ 5 ตัวแปร : x, y, dx, dy, r
- ต้องการ 50 ลูก : 250 ตัวแปร !
 - x1, y1, dx1, dy1, r1
 - x2, y2, dx2, dy2, r2
 - ...
 - x50, y50, dx50, dy50, r50
- ทางออกที่ง่ายกว่า : ใช้แถวลำดับเก็บข้อมูล (อาเรย์)



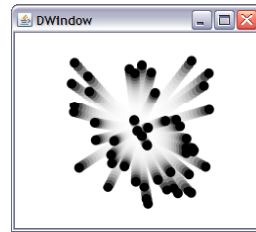
	0	1	2	3		46	47	48	49
x	100	10	15	30	. . .	173	40	45	180
y	30	110	115	83	. . .	30	40	20	80
dx	1	2	3	-2	. . .	-1	-2	3	-2
dy	-2	-3	3	0	. . .	-1	2	1	3
r	5	5	5	5	. . .	5	5	5	5

ลูกบอล 50 ลูก

```
DWindow w = new DWindow(250, 200);
int n = 50;
double[] x = new double[n], y = new double[n];
double[] r = new double[n];
double[] dx = new double[n], dy = new double[n];
for (int i = 0; i < n; i++) {
    x[i] = 125; y[i] = 100; r[i] = 5;
    dx[i] = -4 + 8 * Math.random();
    dy[i] = -4 + 8 * Math.random();
}
while (true) {
    w.fade(0.3);
    for (int i = 0; i < n; i++) {
        w.fillEllipse(x[i], y[i], 2 * r[i], 2 * r[i]);
        x[i] = x[i] + dx[i]; y[i] = y[i] + dy[i];
        if (x[i] + r[i] > 250) dx[i] = -dx[i];
        if (x[i] - r[i] < 0) dx[i] = -dx[i];
        if (y[i] + r[i] > 200) dy[i] = -dy[i];
        if (y[i] - r[i] < 0) dy[i] = -dy[i];
    }
    w.sleep(30);
}
```

สร้างอาร์เรย์

ให้ค่าเริ่มต้น



วาดและปรับค่า

21101

5

แถวลำดับ (อาร์เรย์ – array)

- ที่เก็บข้อมูลประเภทเดียวกัน เก็บเรียงเป็นแถว
- อาร์เรย์มีชื่อกำกับ
- แต่ละช่องของอาร์เรย์มีหมายเลขกำกับ เริ่มที่ 0
- วิธีใช้ : ชื่อ[หมายเลขช่อง]

	0	1	2	3	4	5	6	7	8	9
x	100	10	15	30	45	96	173	40	45	180
y	30	110	115	83	145	52	30	40	20	80

y[2]

i = 4;
y[i+1] = 52;

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (02/06/52)

6

การประกาศตัวแปรอาร์เรย์และการสร้างตัวอาร์เรย์

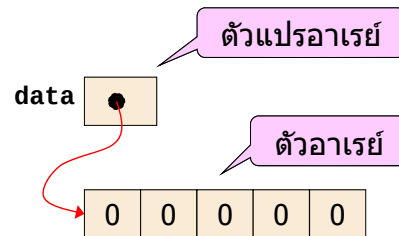
- ก่อนใช้อาร์เรย์ ต้อง

– ประกาศตัวแปรอาร์เรย์

```
int[] data;
```

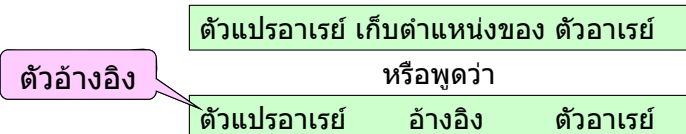
– สร้างตัวอาร์เรย์

```
data = new int[5];
```



- ประกาศตัวแปรและสร้างอาร์เรย์พร้อมกันก็ได้

```
int [] data = new int[5];
```



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (02/06/52)

7

การประกาศตัวแปรอาร์เรย์และการสร้างตัวอาร์เรย์

- การประกาศตัวแปรอาร์เรย์ ไม่ได้เป็นการสร้างตัวอาร์เรย์
`int[] data;`
- ขนาดของอาร์เรย์ถูกกำหนดตอนสร้างตัวอาร์เรย์
`data = new int[1000];`
- ขนาดของอาร์เรย์เป็นค่าของนิพจน์ก็ได้
`data = new int[ 3*n + 1];`
- อาร์เรย์ที่สร้างขึ้น ไม่สามารถเพิ่มหรือลดขนาดได้
- ระบบจะตั้งค่าเริ่มต้นของข้อมูลในอาร์เรย์โดยอัตโนมัติ (ให้ค่า 0 กับอาร์เรย์ของจำนวน)
- `d.length` เก็บจำนวนช่องของอาร์เรย์ d
`data = new int[4];`
`System.out.println( data.length ); // 4`

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (02/06/52)

8

ตัวอย่าง : โปรแกรมเก็บสถิติคะแนนสอบ

- แฟ้มคะแนนเก็บรหัสนักเรียน + คะแนน
- คะแนนมีตั้งแต่ 0 ถึง 4 (จำนวนเต็ม)
- อยากทราบว่านักเรียนได้ 0 ก็คน 1 ก็คน ... 4 ก็คน

4531001321	3
4531003621	3
4531004221	4
4531005921	3
4531006521	3
4531101421	4
4531102021	4
4531102221	0



คะแนน	จำนวนนักเรียน
0	1
1	0
2	0
3	4
4	3

โปรแกรมเก็บสถิติคะแนนสอบ : ใช้ตัวแปร 5 ตัว

```
public static void main(String[] args)
    throws IOException {
    Scanner in = new Scanner(new File("scores.txt"));
    int c0 = 0, c1 = 0, c2 = 0, c3 = 0, c4 = 0;
    while (in.hasNext()) {
        String id = in.next();           // อ่านรหัส
        int p = in.nextInt();           // อ่านคะแนน
        if (p == 0) c0++;
        if (p == 1) c1++;
        if (p == 2) c2++;
        if (p == 3) c3++;
        if (p == 4) c4++;
    }
    in.close();
    System.out.println(c0 + " คนได้ 0 คะแนน");
    System.out.println(c1 + " คนได้ 1 คะแนน");
    System.out.println(c2 + " คนได้ 2 คะแนน");
    System.out.println(c3 + " คนได้ 3 คะแนน");
    System.out.println(c4 + " คนได้ 4 คะแนน");
}
```

4531001321	3
4531003621	3
4531004221	4
4531005921	3
4531006521	3
...	

โปรแกรมเก็บสถิติคะแนนสอบ : ใช้อาเรย์ 5 ช่อง

```
public static void main(String[] args)
    throws IOException {
    Scanner in = new Scanner(new File("scores.txt"));
    int[] c = new int[5];
    while (in.hasNext()) {
        String id = in.next();           // อ่านรหัส
        int p = in.nextInt();           // อ่านคะแนน
        c[p]++;
    }
    in.close();
    for (int p = 0; p < c.length; p++) {
        System.out.println(c[p] + " คนได้ " + p + " คะแนน");
    }
}
```

จำนวนช่องของอาเรย์ c

c[p] แทนจำนวนนักเรียนที่ได้ p คะแนน
 p = 0, 1, 2, 3, 4
 c[0], c[1], c[2], c[3], c[4]

ประกาศ + สร้าง + ตั้งค่าเริ่มต้น

```
int[] data = new int[4];
data[0] = 23;
data[1] = 3;
data[2] = 21;
data[3] = 47;
```

23	3	21	47
----	---	----	----

ไม่ต้องบอกจำนวนช่อง

```
int[] data = new int[]{23, 3, 21, 47};
data = new int[]{23, 3, 21, 47};
```

เขียนแบบสั้น

```
int[] data = {23, 3, 21, 47};
```

~~int[] data;
data = {23, 3, 21, 47};~~ เขียนแยกบรรทัดไม่ได้

โปรแกรมรับเลขเดือน → แสดงชื่อเดือน

```
import java.util.Scanner;

public class Month {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("เลขเดือน = ");
        int m = kb.nextInt();
        if (1 <= m && m <= 12) {
            String[] month = {"JAN", "FEB", "MAR", "APR", "MAY",
                               "JUN", "JUL", "AUG", "SEP", "OCT", "NOV", "DEC" };
            System.out.println( month[m - 1] );
        } else {
            System.out.println("เลขเดือนไม่ถูกต้อง");
        }
    }
}
```

เดือน m อยู่ที่
ช่อง m - 1

โปรแกรมค้นชื่อจากรหัส

- เพิ่มข้อมูลเก็บรหัสและชื่อนักเรียน
- เขียนโปรแกรม
 - รับรหัสนักเรียนจากผู้ใช้งาน (ทางแป้นพิมพ์)
 - แสดงชื่อของนักเรียนที่มีรหัสที่ได้รับ (ทางจอภาพ)

3413	นาย จาวา แซ่ซัน
1002	นาย หมด ฤทธิเดช
9821	น.ส. ดวงดี มากล้น
2123	นาย ห้าม ไม่ฟัง
4225	นาย ลิม ไปแล้ว
5421	น.ส. สมหญิง ตัวจริง
...	

โปรแกรมค้นชื่อจากรหัส : ค้นในแฟ้ม

```
Scanner kb = new Scanner(System.in);
while (true) {
    System.out.print("ID = ");
    int sid = kb.nextInt();
    if (sid < 0) break;
    Scanner in = new Scanner(new File("id-names.txt"));
    String sname = "";
    while (in.hasNext()) {
        int id = in.nextInt();
        String name = in.nextLine();
        if (id == sid) {
            sname = name.trim();
            break; // ออกจากวงวนทันที เมื่อพบ
        }
    }
    in.close();
    if (sname.equals("")) {
        System.out.println("--- ไม่พบชื่อของรหัส " + sid);
    } else {
        System.out.println(sname);
    }
}
```

3413	นาย จาวา แซ่ซัน
1002	นาย หมด ฤทธิเดช
9821	น.ส. ดวงดี มากล้น
...	

ค้นจากแฟ้มทุกครั้ง
ซ้ำ !!!

โปรแกรมค้นชื่อจากรหัส : ค้นในอาเรย์

```
Scanner in = new Scanner(new File("id-names.txt"));
int[] id = new int[1000];
String[] name = new String[1000];
int n = 0; // ตัวแปรนับจำนวนนักเรียน
while (in.hasNext()) {
    id[n] = in.nextInt(); name[n] = in.nextLine().trim();
    n++;
}
in.close();
Scanner kb = new Scanner(System.in);
while (true) {
    System.out.print("ID = ");
    int sid = kb.nextInt();
    if (sid < 0) break;
    int i = 0;
    for (i = 0; i < n; i++) {
        if (sid == id[i]) break;
    }
    if (i == n) {
        System.out.println("--- ไม่พบชื่อของรหัส " + sid);
    } else {
        System.out.println(name[i]);
    }
}
```

ทำอย่างไรจึงจะสร้างอาเรย์
ให้พอดีกับจำนวนนักเรียน

อ่านรหัสและชื่อเก็บ
ใส่อาเรย์ก่อน

ใช้ id.length ไม่ได้ เพราะ
ข้อมูลเก็บใน n ช่องทางซ้าย

ไล่เปรียบเทียบทีละช่องใน
อาเรย์ พบก็ออกนอกวงวน

การค้นข้อมูลแบบลำดับ

ไล่เปรียบเทียบไปทีละตัวจากซ้ายไปขวา

	0	1	2	3	4	5	6
d	23	2	3	14	5	99	9

ค้น 14

ค้น 11

ถ้าข้อมูลในอาเรย์เรียงลำดับ

ใช้การค้นแบบทวิภาค (binary search)
เร็วกว่า

	0	1	2	3	4	5	6	7	8	9	10	11
d	2	3	5	9	11	20	25	39	44	49	52	79



ค้น 25

ถ้าข้อมูลในอาเรย์เรียงลำดับ

ใช้การค้นแบบทวิภาค (binary search)
เร็วกว่า

	0	1	2	3	4	5	6	7	8	9	10	11
d	2	3	5	9	11	20	25	39	44	49	52	79



ค้น 27

โปรแกรมค้นชื่อจากรหัส : ค้นในอาเรย์แบบทวิภาค

```
int[] id = new int[1000];
String[] name = new String[1000];
int n = 0; // ตัวแปรนับจำนวนนักเรียน
// ขอละค่าสิ่งตรงนี้อ่านข้อมูลในแฟ้มลงอาเรย์ และเรียงลำดับอาเรย์ตามรหัสแล้ว

Scanner kb = new Scanner(System.in);
while (true) {
    System.out.print("ID = ");
    int sid = kb.nextInt();
    if (sid < 0) break;
    int left = 0, right = n - 1, mid = 0;
    while (left <= right) {
        mid = (left + right) / 2;
        if (sid == id[mid]) break;
        if (sid < id[mid])
            right = mid - 1;
        else
            left = mid + 1;
    }
    if (sid == id[mid]) {
        System.out.println(name[mid]);
    } else {
        System.out.println("--- ไม่พบชื่อของรหัส " + sid);
    }
}
```

การเรียงลำดับข้อมูลในอาเรย์



การเรียงลำดับข้อมูลแบบเลือก



การหาค่ามากสุดในอาเรย์

```
int[] d = ... ; // มีอาเรย์ของจำนวนเต็ม
// ...

int max = d[0]; // max เก็บค่ามากสุด
for (int i = 1; i < d.length; i++) {
    if (d[i] > max) max = d[i];
}
```

```
...
int maxIndex = 0; // maxIndex เก็บหมายเลขช่อง
                  // ที่เก็บค่ามากสุด
for (int i = 1; i < d.length; i++) {
    if (d[i] > d[maxIndex]) maxIndex = i;
}
```

โปรแกรมการเรียงลำดับข้อมูลแบบเลือก

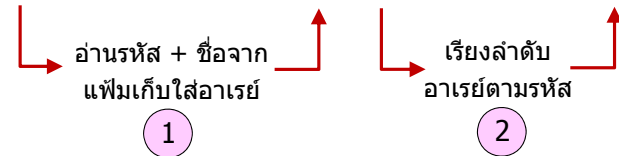
```
int[] d = { 1, 4, 2, 6, 3, 7, 8, 9, 0 };
//-----
for (int lastIndex = d.length - 1; lastIndex >= 1;
    lastIndex--) {

    int maxIndex = 0;
    for (int i = 1; i <= lastIndex; i++) {
        if (d[i] > d[maxIndex]) maxIndex = i;
    }
    int t = d[maxIndex]; // สลับข้อมูลตัวที่ maxIndex
    d[maxIndex] = d[lastIndex]; // กับ lastIndex
    d[lastIndex] = t;
}
//-----
for (int i = 0; i < d.length; i++) {
    System.out.print(d[i] + " "); // 0 1 2 3 4 6 7 8 9
}
System.out.println();
```

โปรแกรมค้นชื่อจากรหัส (อีกครั้ง)

1. อ่านรหัส + ชื่อจากแฟ้มเก็บใส่อาเรย์ (สร้างอาเรย์ตามจำนวนนักเรียน)
2. เรียงลำดับอาเรย์ตามรหัส
3. ค้นรหัสด้วย binary search

3413	นาย จาวา แซ่ซัน	3413	นาย จาวา แซ่ซัน	1002	นาย หมด ฤทธิ์เดช
1002	นาย หมด ฤทธิ์เดช	1002	นาย หมด ฤทธิ์เดช	2123	นาย ห้าม ไม่ฟัง
9821	น.ส. ดวงดี มากล้น	9821	น.ส. ดวงดี มากล้น	3413	นาย จาวา แซ่ซัน
2123	นาย ห้าม ไม่ฟัง	2123	นาย ห้าม ไม่ฟัง	4225	นาย สิม ไปแล้ว
4225	นาย สิม ไปแล้ว	4225	นาย สิม ไปแล้ว	5421	น.ส. สมหญิง ตัวจริง
5421	น.ส. สมหญิง ตัวจริง	5421	น.ส. สมหญิง ตัวจริง	9821	น.ส. ดวงดี มากล้น



1. อ่านรหัส + ชื่อเก็บใส่อาเรย์

```
Scanner in = new Scanner(new File("id-names.txt"));
int n = 0;
while (in.hasNext()) {
    in.nextLine();
    n++;
}
in.close();

int[] id = new int[n];
String[] name = new String[n];

in = new Scanner(new File("id-names.txt"));
for (int i = 0; i < id.length; i++) {
    id[i] = in.nextInt();
    name[i] = in.nextLine().trim();
}
in.close();
```

อ่านแฟ้มรอบแรกเพื่อ
นับจำนวนนักเรียน

สร้างอาเรย์ตามจำนวนนักเรียน

อ่านแฟ้มอีกรอบ
เก็บข้อมูลใส่อาเรย์

2. เรียงลำดับอาเรย์ตามรหัส

```
for (int lastIndex = id.length - 1; lastIndex >= 1;
     lastIndex--) {

    int maxIndex = 0;
    for (int i = 1; i <= lastIndex; i++) {
        if (id[i] > id[maxIndex]) maxIndex = i;
    }

    int tid = id[maxIndex];
    id[maxIndex] = id[lastIndex];
    id[lastIndex] = tid;

    String tname = name[maxIndex];
    name[maxIndex] = name[lastIndex];
    name[lastIndex] = tname;
}
```

3. ค้นรหัสแบบ binary search

```
Scanner kb = new Scanner(System.in);
while (true) {
    System.out.print("ID = ");
    int sid = kb.nextInt();
    if (sid < 0) break;
    int left = 0, right = n - 1, mid = 0;
    while (left <= right) {
        mid = (left + right) / 2;
        if (sid == id[mid]) break;
        if (sid < id[mid])
            right = mid - 1;
        else
            left = mid + 1;
    }
    if (sid == id[mid]) {
        System.out.println(name[mid]);
    } else {
        System.out.println("--- ไม่พบชื่อของรหัส " + sid);
    }
}
```

ข้อควรระวัง

- index ของอาร์เรย์ต้องเป็นจำนวนเต็ม int
- ถ้าอาร์เรย์มี n ช่อง, $0 \leq \text{index} \leq (n - 1)$
- เกิดข้อผิดพลาด เมื่อใช้ช่องที่นอกช่วงของอาร์เรย์

```
public class Array3 {
    public static void main(String [] args) {
        int[] data = new int[100];
        System.out.println( data[100] );
    }
}
```

```
JLab>java Array3
java.lang.ArrayIndexOutOfBoundsException
    at Array3.main(Array3.java:4)
Exception in thread "main"
JLab>
```

ฝึกเติมค่าในอาร์เรย์

	0	1	2	3	4	...	94	95	96	97	98	99
d	0	1	2	3	4	...	94	95	96	97	98	99

	0	1	2	3	4	...	94	95	96	97	98	99
d	1	2	3	4	5	...	95	96	97	98	99	100

	0	1	2	3	4	...	94	95	96	97	98	99
d	100	99	98	97	96	...	6	5	4	3	2	1

	0	1	2	3	4	...	94	95	96	97	98	99
d	1	-1	2	-2	3	...	48	-48	49	-49	50	-50

	0	1	2	3	4	5	6	7	8	9	...	99
d	0	1	3	6	10	15	21	28	36	45	...	4950

ทบทวนก่อนสอบกลางภาค

ตัวอย่างที่จะนำเสนอ

- คำตอบของสมการเชิงเส้นตัวแปรเดียว
- หาร่วมมาก
- มากสุดอันดับสอง
- เกมทายตัวเลข
- วรรณยุกต์ใดใช้มากที่สุด
- แปลคำอังกฤษเป็นไทย

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (02/06/52)

2

คำตอบของสมการเชิงเส้นสองตัวแปร

- เขียนโปรแกรม TwoUnknowns เพื่อคำนวณหาค่าของ x_1 และ x_2 จากสมการ

$$a_{11}x_1 + a_{12}x_2 = b_1$$

$$a_{21}x_1 + a_{22}x_2 = b_2$$

- รับค่าคงตัว a และ b จากแป้นพิมพ์

input {
JLab>java TwoUnknowns
a11 = 3
a12 = -1
a21 = 2
a22 = 3
b1 = 5
b2 = 7
output {
x1 = 2.0
x2 = 1.0
JLab>

$$\begin{aligned} 3x_1 - x_2 &= 5 \\ 2x_1 + 3x_2 &= 7 \end{aligned}$$

$$\text{ได้ } x_1 = 2 \text{ และ } x_2 = 1$$

TwoUnknowns

$$a_{11}x_1 + a_{12}x_2 = b_1$$

$$a_{21}x_1 + a_{22}x_2 = b_2$$

$$a_{11}a_{22}x_1 + a_{12}a_{22}x_2 = b_1a_{22}$$

$$a_{12}a_{21}x_1 + a_{12}a_{22}x_2 = b_2a_{12}$$

$$a_{11}a_{22}x_1 - a_{12}a_{21}x_1 = b_1a_{22} - b_2a_{12}$$

$$x_1 = \frac{b_1a_{22} - b_2a_{12}}{a_{11}a_{22} - a_{12}a_{21}}$$

$$x_2 = \frac{b_1a_{21} - b_2a_{11}}{a_{12}a_{21} - a_{11}a_{22}}$$

TwoUnknowns

```
Scanner kb = new Scanner(System.in);
System.out.print("a11 = ");
double a11 = kb.nextDouble();
System.out.print("a12 = ");
double a12 = kb.nextDouble();
System.out.print("a21 = ");
double a21 = kb.nextDouble();
System.out.print("a22 = ");
double a22 = kb.nextDouble();
System.out.print("b1 = ");
double b1 = kb.nextDouble();
System.out.print("b2 = ");
double b2 = kb.nextDouble();
```

$$x_1 = \frac{b_1 a_{22} - b_2 a_{12}}{a_{11} a_{22} - a_{12} a_{21}}$$

$$x_2 = \frac{b_1 a_{21} - b_2 a_{11}}{a_{12} a_{21} - a_{11} a_{22}}$$

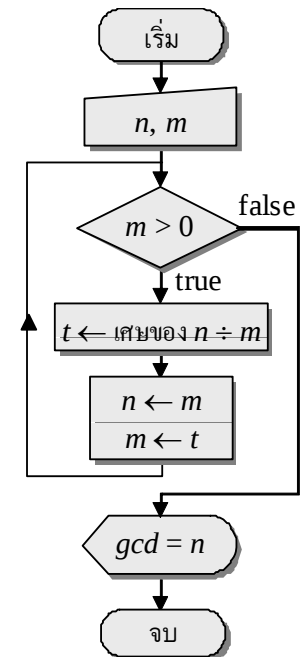
```
double x1 = (b1*a22 - b2*a12)/(a11*a22 - a21*a12);
double x2 = (b1*a21 - b2*a11)/(a12*a21 - a11*a22);
```

```
System.out.println("x1 = " + x1);
System.out.println("x2 = " + x2);
```

GCD : หาร่วมมาก

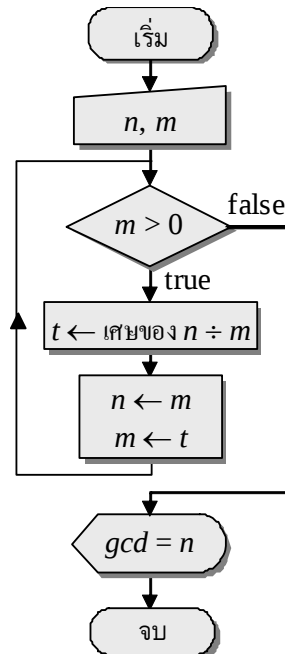
- เขียนโปรแกรมรับค่า n กับ m ทางแป้นพิมพ์
- หาค่าหารร่วมมากของ n กับ m ตามผังงานด้านขวา

n	m	t
2211	1122	1089
1122	1089	33
1089	33	0



GCD : หาร่วมมาก

```
Scanner kb = new Scanner(System.in);
System.out.print("n = ");
int n = kb.nextInt();
System.out.print("m = ");
int m = kb.nextInt();
while (m > 0) {
    int t = n % m;
    n = m;
    m = t;
}
System.out.println("gcd = " + n);
```



SecondMax

- เพิ่มชื่อ data.txt เก็บจำนวนเต็มมากมาย
- จงเขียนโปรแกรม SecondMax ที่หาและแสดงจำนวนเต็มบวกที่มีค่ามากที่สุดเป็นอันดับสองในแฟ้ม data.txt
- ถ้ามีจำนวนมากที่สุดซ้ำกัน ให้ถือว่าจำนวนมากที่สุดเหล่านั้นเป็นอันดับหนึ่ง
- ถ้าทุกจำนวนในแฟ้มมีค่าเท่ากันหมด หรือแฟ้มนี้มีข้อมูลน้อยกว่าสองจำนวน ให้แสดงว่า ไม่มีตัวมากที่สุดอันดับสอง

2	2	4	4
4	4	4	4
0	3	4	4
3	5	4	4
5	5	4	4
1	5	4	4

SecondMax

```
Scanner in = new Scanner(new File("data.txt"));
int max = -1;
int max2 = -1;
while (in.hasNext()) {
    int v = kb.nextInt();
    if (v > max) {
        max2 = max;
        max = v;
    }
    if (max2 < v && v < max) max2 = v;
}
in.close();
if (max2 == -1) {
    System.out.println("ไม่มีตัวมากที่สุดอันดับสอง");
} else {
    System.out.println("ตัวมากที่สุดอันดับสอง = " + max2);
}
```

เกมทายตัวเลข

- โปรแกรมสุ่มจำนวนเต็ม 1 ถึง 100
- ให้โอกาสทาย 7 ครั้ง
- ถ้าทายไม่ถูก โปรแกรมจะบอก "น้อยกว่า" "มากกว่า"

เกมทายตัวเลข : คุณมีโอกาส 7 ครั้ง
การทายครั้งที่ 1 : 50
มากกว่า
การทายครั้งที่ 2 : 87
น้อยกว่า
การทายครั้งที่ 3 : 62
น้อยกว่า
การทายครั้งที่ 4 : 57
น้อยกว่า
การทายครั้งที่ 5 : 55
ถูกต้องแล้วครับ

GuessMyNumber

```
System.out.println("เกมทายตัวเลข : คุณมีโอกาส 7 ครั้ง");
Scanner kb = new Scanner(System.in);
int myNumber = 1 + (int)(100 * Math.random());
int yourGuess = 0;
for (int k = 1; k <= 7; k++) {
    System.out.print("การทายครั้งที่ " + k + " : ");
    yourGuess = kb.nextInt();
    if (myNumber == yourGuess) break;
    if (myNumber > yourGuess)
        System.out.println("มากกว่า");
    else
        System.out.println("น้อยกว่า");
}
if (myNumber == yourGuess) {
    System.out.println("ถูกต้องแล้วครับ");
} else {
    System.out.println("เสียใจครับ");
    System.out.println("ผมกำลังนึกถึงเลข " + myNumber);
}
```

วรรณยุกต์ตัวใดใช้มากที่สุด

- ไม่เอก ไม่โท ไม่ตรี หรือไม่จัตวา ?
- ขอใช้วิธีนับจากคำศัพท์ในพจนานุกรมราชบัณฑิตย
- แฟ้ม riwords0.txt เก็บศัพท์ บรรทัดละคำหรือวลี

ก
ก็
กก
ก็ก

...

ไฮโดรมิเตอร์
ไฮไฟ
ไฮโล
ไฮฮิ

มีสามหมื่นกว่าคำ

ToneCount

```
Scanner in = new Scanner(new File("c:/riwords0.txt"));
int c1 = 0, c2 = 0, c3 = 0, c4 = 0;
String t1 = "ิ", t2 = "ึ", t3 = "ึ", t4 = "ึ";
while (in.hasNext()) {
    String line = in.nextLine();
    line = line.trim();
    for (int k = 0; k < line.length(); k++) {
        String ch = line.substring(k, k + 1);
        if (ch.equals(t1)) c1++;
        if (ch.equals(t2)) c2++;
        if (ch.equals(t3)) c3++;
        if (ch.equals(t4)) c4++;
    }
}
in.close();
System.out.println("ไม้เอกมี\t " + c1 + " ตัว");
System.out.println("ไม้โทมี\t " + c2 + " ตัว");
System.out.println("ไม้ตรีมี\t " + c3 + " ตัว");
System.out.println("ไม้จัตวามี\t " + c4 + " ตัว");
```

21

13

โปรแกรมแปลคำอังกฤษเป็นไทย

- มีแฟ้ม verb-dict.txt
 - เก็บคำกริยาภาษาอังกฤษ 1 บรรทัด
 - ตามด้วยคำแปลภาษาไทย 1 บรรทัด
- nextInt ครั้งแรกจากแฟ้มจะได้จำนวนค่า

2026 abate
(อะเบท') vt.,vi. ลดน้อยลง, บรรเทา, เล็กล้น, เล็ก, ระวัง
abbreviate
(อะบริวียเอท) vt. ย่อ, เขียนย่อ, คำย่อ, การย่อ
abide
(อะไบด์') vi. รักษาหรือ ปฏิบัติตาม,อาศัยอยู่,ทน, ยึดถือ, รอคอย
...
yowl
(เยาล) vi. ร้องเสียงยาวแสดงความไม่พอใจหรือผิดหวัง,ว้าย!. n. เสียงร้องดังกล่าว

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (02/06/52)

14

Eng2Thai

```
import java.util.Scanner;
import java.io.*;

public class Eng2Thai {
    public static void main(String[] args)
        throws IOException {
        Scanner in = new Scanner(new File("dict.txt"));
        int n = in.nextInt();
        String[] eng = new String[n];
        String[] thai = new String[n];
        for (int i = 0; i < eng.length; i++) {
            eng[i] = in.nextLine().trim();
            thai[i] = in.nextLine().trim();
        }
        in.close();
```

2026 abate
(อะเบท') vt.,vi. ลดน้อยลง, บรรเทา, เล็กล้น, เล็ก, ระวัง
abbreviate
(อะบริวียเอท) vt. ย่อ, เขียนย่อ, คำย่อ, การย่อ
...

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (02/06/52)

15

Eng2Thai

```
Scanner kb = new Scanner(System.in);
while (true) {
    System.out.print("eng = ");
    String word = kb.nextLine();
    if (word.equals(".")) break;
    int i = 0;
    for (i = 0; i < n; i++) {
        if (word.equals(eng[i])) break;
    }
    if (i == n) {
        System.out.println("--- ไม่คำแปลของ " + word);
    } else {
        System.out.println(thai[i]);
    }
}
}
```

ต้องประกาศ i นอก for
เพราะจะใช้ i นอก for

อย่าใช้ == ในการเปรียบเทียบสตริง
if (word == eng[i]) ...

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (02/06/52)

16

การเขียนโปรแกรมย่อย

- องค์ประกอบของเมทอด
- การเรียกเมทอดและการจัดเก็บตัวแปร
- การส่งและรับค่าของเมทอด
- เมทอดกับอาเรย์
- ตัวอย่าง

การเขียนโปรแกรม

- แบ่งโปรแกรมเป็นส่วนย่อยๆ
- เรียกส่วนย่อยเหล่านี้ว่า subroutine, subprogram, function หรือ method
- แต่ละ method มีหน้าที่ในตัวเองอย่างเด่นชัด

```
class Game {  
    main(...) {  
        ...  
    }  
  
    begin(...) {  
    }  
  
    option(...) {  
    }  
  
    ending(...) {  
    }  
  
    demo(...) {  
    }  
}
```

ตัวอย่าง : ดัชนีมวลกาย

```
import java.util.Scanner;  
public class BodyMassIndex {  
    public static void main(String[] args) {  
        Scanner kb = new Scanner(System.in);  
  
        double weight = readDouble(kb, "น้ำหนัก (kg.) = ");  
  
        double height = readDouble(kb, "ความสูง (cm.) = ");  
        double hm = height / 100.0;  
        double bmi = weight / (hm * hm);  
        System.out.println("ดัชนีมวลกาย = " + bmi);  
    }  
  
    public static double readDouble(Scanner kb,  
                                    String msg){  
        System.out.print(msg);  
        double v = kb.nextDouble();  
        return v;  
    }  
}
```

องค์ประกอบของเมทอด

ประเภทของผลลัพธ์ ชื่อเมทอด รายการพารามิเตอร์

```
public static int readInt(Scanner kb, String msg){
    System.out.print(msg);
    int v = kb.nextInt();
    return v;
}
```

คืนผลจากเมทอด

static – เมทอดประจำคลาส
public – เมทอดสาธารณะ
private – เมทอดที่ใช้ได้เฉพาะในคลาสที่เขียน

argument (อาร์กิวเมนต์)

ส่วนหัว : ชื่อเมทอด

- ใช้กฎการตั้งชื่อเหมือนกับตัวแปร
- มักตั้งชื่อเมทอดขึ้นต้นด้วยตัวอักษรเล็ก
- มักตั้งชื่อเมทอดให้เป็นกริยา

fillEllipse	drawLine	fade	readDouble
print	println	hasNext	nextInt
random	max	min	cos

ส่วนหัว : ประเภทของผลลัพธ์

- **int, double, String, ...**
- ถ้าไม่คืนผล, ใช้คำว่า **void**

```
public static int max(int a, int b)
```

```
public static double sqrt(double x)
```

```
public static int getCurrentMonth()
```

```
public static void printError(String msg)
```

ส่วนหัว : รายการของพารามิเตอร์

- parameter คือตัวแปรสำหรับรับข้อมูลจากผู้เรียก

```
public static void printError(String msg)
```

- parameter แต่ละตัว เขียนเหมือนการประกาศตัวแปร (คั่นด้วย , ไม่ต้องปิดท้ายด้วย ;)

```
public static int max(int a, int b)
```

- รายการของ parameters อยู่ภายในวงเล็บ (ถ้าไม่รับ parameter ใดๆ ก็ไม่ต้องใส่อะไรในวงเล็บ)

```
public static int getCurrentMonth()
```

ส่วนหัว : ตัวอย่างที่ไม่ผิด

```
public static int a( int a )
```

```
public static void Nothing( )
```

```
public static void $$$ ( double $1, int $2 )
```

ชื่อเมทอดซ้ำกันได้ถ้ารายการของประเภทพารามิเตอร์ไม่เหมือน

```
public static int f(int a, int b) {  
    ...  
}  
public static int f(int a, int b, double c) {  
    ...  
}
```

ส่วนหัว : ตัวอย่างผิด ๆ

```
public static getFirstCharacter(String txt)
```

```
public static int min( int a, b )
```

```
public static void clear( void )
```

```
public static int f(int a, int b) {  
    ...  
}  
public static double f(int a, int b) {  
    ...  
}  
public static double f(int x, int y) {  
    ...  
}
```

ส่วนตัว : การคืนการทำงานสู่ผู้เรียก

- เมทอดแบบ **void** คืนการทำงาน เมื่อทำคำสั่ง **return** หรือ ทำจนถึง } ที่ปิดตัวเมทอด

```
public static void main(String[] args) {  
    Scanner kb = new Scanner(System.in);  
    double a = readDouble("a = ");  
    if (a < 0) {  
        System.out.println("ค่า a ต้องมีมากกว่า 0");  
        return;  
    }  
    double x = 1.0;  
    while ((x*x - a) >= 1e-5) {  
        x = (x + (a / x)) / 2.0;  
    }  
    System.out.println("รากที่สอง = " + x);  
}
```

ส่วนตัว : การคืนการทำงานสู่ผู้เรียก

- เมทอดที่มีการคืนผล คืนการทำงาน เมื่อ
 - ทำคำสั่ง **return** ตามด้วยนิพจน์ที่แทนค่าที่คืน
 - คืนผลได้เพียงค่าเดียวเท่านั้น
 - ค่าที่คืนต้องเป็น
 - ประเภทเดียวกับที่ประกาศไว้ที่หัว
 - ประเภทที่เปลี่ยนเป็นแบบที่ประกาศไว้อย่างอัตโนมัติได้

```
public static double clip(double x) {  
    if (x < 0) return 0;  
    return x;  
}
```

ส่วนตัว : พารามิเตอร์และตัวแปร

- ตัวแปรเฉพาะที่ (local variables)
 - ประกาศในเมทอดใด ใช้ได้เฉพาะในเมทอดนั้น
 - ระบบสร้างตัวแปรเฉพาะที่ เมื่อเมทอดถูกเรียก
 - ระบบลบตัวแปรเฉพาะที่ เมื่อเมทอดทำงานเสร็จ
 - ซ้ำกันไม่ได้ ถ้าอยู่คนละเมทอด

```
public static int b(int x) {  
    int a = x + 8;  
    return a;  
}  
public static int a(int x) {  
    int a = 2 * x;  
    return a;  
}
```

ตัวอย่างผิด ๆ

```
public static int abs(int a) {  
    if ( a < 0 ) a = -a;  
}
```

```
public static int getBlackColor() {  
    return 0.0;  
}
```

```
public static double length(double dx, double dy){  
    double dx = Math.abs(dx);  
    return Math.sqrt( dx*dx + dy*dy );  
}
```

GCD : หาร่วมมาก

```
public static void main(String[] args) {  
    Scanner kb = new Scanner(System.in);  
    System.out.print("n = ");  
    int n = kb.nextInt();  
    System.out.print("m = ");  
    int m = kb.nextInt();  
    System.out.println("gcd = " + gcd(n,m) );  
}  
public static int gcd(int n, int m) {  
    while (m > 0) {  
        int t = n % m;  
        n = m;  
        m = t;  
    }  
    return n;  
}
```

โปรแกรม : วันใดของสัปดาห์

```
public static String day(int d, int m, int y) {  
    if (m < 3) {  
        m = m + 12; y = y - 1;  
    }  
    int c = y / 100, k = y % 100;  
    int w = (d + 26*(m+1)/10 + k + k/4 + c/4 + 5*c) % 7;  
    String dow = "";  
    if (w == 0) dow = "เสาร์";  
    if (w == 1) dow = "อาทิตย์";  
    if (w == 2) dow = "จันทร์";  
    if (w == 3) dow = "อังคาร";  
    if (w == 4) dow = "พุธ";  
    if (w == 5) dow = "พฤหัสบดี";  
    if (w == 6) dow = "ศุกร์";  
    return dow;  
}
```

การเรียกเมทอด

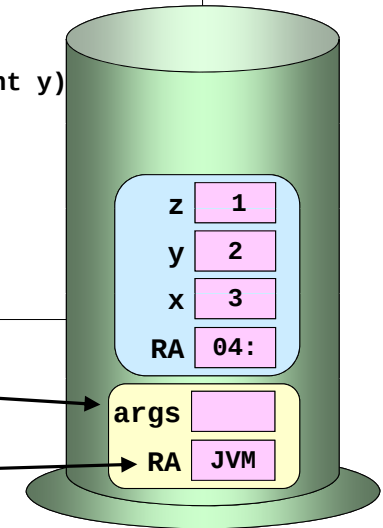
```
01:public class StackFrame {
02:  public static void main(String[] args) {
03:    a(3, 2);
04:    b(5);
05:  }
06:  public static void a(int x, int y) {
07:    int z = x/y;
08:    b(z);
09:  }
10:  public static void b(int x) {
11:    ++x;
12:  }
13:}
```

ตัวแปรและพารามิเตอร์ในเมทอดหนึ่ง "เกิด" เมื่อเมทอดนั้นถูกเรียกให้ทำงาน
ตัวแปรเหล่านี้ "หาย" ไป เมื่อคืนการทำงานกลับสู่ผู้เรียก

การจัดเก็บตัวแปร

```
01:public class StackFrame {
02:  public static void main(String[] args) {
03:    a(3, 2);
04:    b(5);
05:  }
06:  public static void a(int x, int y) {
07:    int z = x/y;
08:    b(z);
09:  }
10:  public static void b(int x) {
11:    ++x;
12:  }
13:}
```

parameters &
local variables
return address



การส่งค่าและการรับผล

```
int a = -9;
int b = clip(a + 5);
```



```
public static int clip(int a) {
  if (a < 0 ) return 0;
  if (a > 255) return 255;
  return a;
}
```



การส่งค่าและการรับผล

```
int a = 300;
int b = clip(a);
```



```
public static int clip(int a) {
  if (a < 0 ) return 0;
  if (a > 255) return 255;
  return a;
}
```



การส่งค่าและการรับผล

```
int a = 300;
int b = clip(a);
```



```
public static int clip(int a) {
    a = Math.max(a, 0);
    a = Math.min(a, 255);
    return a;
}
```



การส่งค่าและการรับผล

```
int q = distance(10, 10, 20, 30);
int q = (int) distance(10, 10, 20, 30);
double x = 20.5;
double p = distance(10, 10, 20, x);
double p = distance(10, 10, 20, (int) x);
```

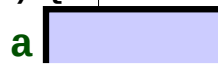
```
public static double distance(int x1, int y1, int x2, int y2){
    double dx = (x1 - x2);
    double dy = (y1 - y2);
    double d = Math.sqrt(dx*dx + dy*dy);
    return d;
}
```

ไม่มีประโยชน์

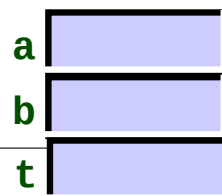
```
int a = 99;
int b = 20;
clear( a );
swap( a, b );
```



```
public static void clear(int a) {
    a = 0;
}
```

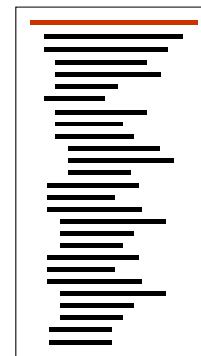


```
public static void swap(int a, int b) {
    int t = a;
    a = b;
    b = t;
}
```



เมื่อใดควรเขียนเมทอดใหม่

เมื่อเมทอดที่เขียนอยู่ยาวเกินไป



เมื่อใดควรเขียนเมท็อดใหม่

เมื่อมีกลุ่มคำสั่งที่เขียนซ้ำ ๆ



ข้อแนะนำในการเขียนเมท็อด

- ควรตั้งชื่อที่สื่อความหมาย
- ควรมีภาระที่ต้องทำหนึ่งอย่างตามชื่อ
- ควรสั้นกะทัดรัด อ่านเข้าใจง่าย
- ควรมีพารามิเตอร์จำนวนไม่มาก
- ควรเขียนหมายเหตุกำกับ

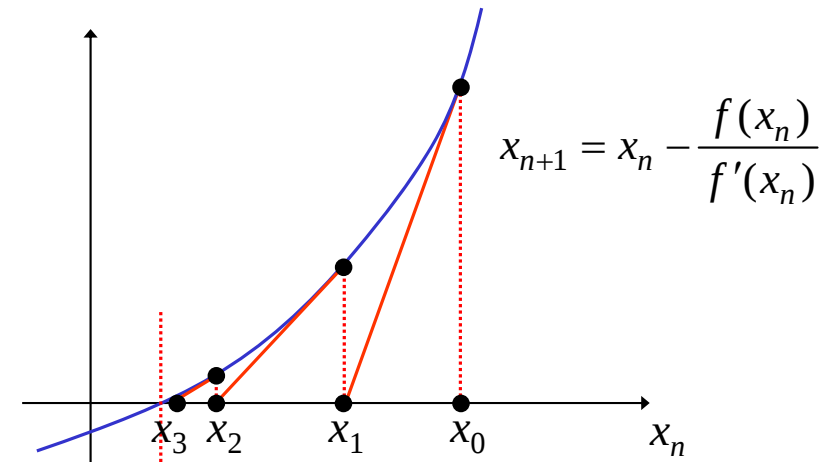
Newton's Method

- ต้องการหารากของสมการ $f(x) = 2x^2 - 11x + 5$
- หาอนุพันธ์เตรียมไว้ก่อน $f'(x) = 4x - 11$
- เริ่มด้วยการให้ค่าเริ่มต้นกับ x_0
- คำนวณ $x_1, x_2, x_3, \dots, x_n, x_{n+1}, \dots$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

- วนคำนวณจนค่า x_n และ x_{n+1} เปลี่ยนแปลงน้อยมาก ๆ

Newton's Method



ตัวอย่าง : โปรแกรมการหารากโดยวิธีของนิวตัน

```
import java.util.Scanner;
public class Newton {
    public static void main(String[] args) {
        Scanner kb = new Scanner(System.in);
        System.out.print("initial x = ");
        double x0 = kb.nextDouble();
        double err = 1.0;
        while (err > 1e-10) {
            System.out.println(x0);
            double x1 = x0 - f(x0) / df(x0);
            err = Math.abs(x1 - x0);
            x0 = x1;
        };
        System.out.println(x0);
    }
    private static double f(double x) {
        return 2*x*x - 11*x + 5;
    }
    private static double df(double x) {
        return 4*x - 11;
    }
}
```

```
JLab>java Newton
initial x = 2
2.0
-1.0
0.19999999
0.48235294
0.49993133
0.49999999
0.5
0.5
```

```
JLab>java Newton
initial x = 10
10.0
6.724137931
5.373999551
5.026653141
5.000156016
5.000000005
5.0
5.0
```

หมายเหตุ โปรแกรมนี้หา
ได้เฉพาะรากจำนวนจริง

$$f(x) = 2x^2 - 11x + 5 = (2x - 1)(x - 5)$$

211 29

เมท็อดที่รับอาร์เรย์

```
public static double sum(double[] d) {
    ...
}

public static int search(int[] d, int x) {
    ...
}

public static void sort(int[] d) {
    ...
}
```

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์ (10/06/52)

30

การส่งอาร์เรย์ไปยังเมท็อด

- ต้องการส่งอาร์เรย์ใดให้เมท็อด
- ส่งเฉพาะชื่อ ไม่ต้องมี []

```
int[] a = { 1, 3, 4, 0, 2, 5 };
int k = search(a, 0);    // ส่ง a ให้ search
sort(a);                 // ส่ง a ให้ sort
int m = max(a[0], a[1]); // ไม่ได้ส่งอาร์เรย์
```

```
public static int search(int[] d, int x) {
    ...
}
public static void sort(int[] d) {
    ...
}
public static int max(int x, int y) {
    if (x > y) return x;
    return y;
}
```

2110101 วิชา

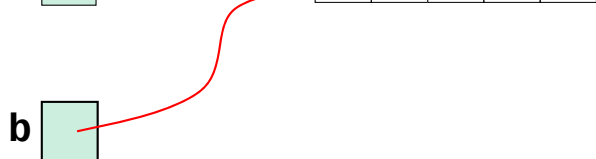
31

ตัวแปรหลายตัวอ้างอิงอาร์เรย์เดียวกันได้

```
int[] a = new int[5];
```



b



```
int[] b = a;
```

a กับ b มีค่าเท่ากัน หมายความว่า อ้างอิงอาร์เรย์เดียวกัน
ดังนั้นผลการเปรียบเทียบ `a == b` ได้ค่า true และ
`a[i]` กับ `b[i]` อ้างอิงที่เก็บข้อมูลที่เดียวกัน

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์ (10/06/52)

32

ส่งอาร์เรย์ = ส่งตำแหน่งของอาร์เรย์

- การส่งอาร์เรย์ไม่ได้ส่งข้อมูลของทั้งอาร์เรย์ไป เป็นเพียงการส่งค่าของตัวแปรอาร์เรย์
- ตัวแปรอาร์เรย์เก็บตำแหน่งอาร์เรย์ (อ้างอิงอาร์เรย์)
- ผู้เรียกเมทอด กับภายในเมทอด ใช้อาร์เรย์เดียวกัน

```
public static void main(String[] args) {
    double[] data = {5, -2, 7, 4, 18};
    ...
    double a = sort( data );
}
```

data

```
public static void sort(double[] x) {
    ...
}
```

x

5 -2 7 4 18

ตัวอย่างเมทอดที่รับอาร์เรย์

```
public static double sum(double[] d) {
    double sum = 0;
    for (int i = 0; i < d.length; i++) {
        sum = sum + d[i];
    }
    return sum;
}
```

```
public static int search(double[] d, int x) {
    for (int i = 0; i < d.length; i++) {
        if (d[i] == x) return i;
    }
    return -1; // หาไม่พบ คืน -1
}
```

ตัวอย่างเมทอดที่รับอาร์เรย์

```
public static double max(double[] d) {
    double max = d[0];
    for (int i = 1; i < d.length; i++) {
        if (d[i] > max) max = d[i];
    }
    return max;
}
```

```
public static int maxIndex(double[] d) {
    int maxIndex = 0;
    for (int i = 1; i < d.length; i++) {
        if (d[i] > d[maxIndex]) maxIndex = i;
    }
    return maxIndex;
}
```

การเรียงลำดับข้อมูลแบบเลือก

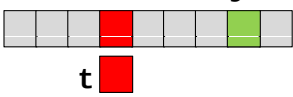
0	1	2	3	4	5	6
23	2	3	14	5	99	9
23	2	3	14	5	9	99
9	2	3	14	5	23	99
9	2	3	5	14	23	99
5	2	3	9	14	23	99
3	2	5	9	14	23	99
2	3	5	9	14	23	99

เมท็อดการเรียงลำดับข้อมูลแบบเลือก

```
public static int maxIndex(double[] d, int k) {
    int maxIndex = 0;
    for (int i = 1; i <= k; i++) {
        if (d[i] > d[maxIndex]) maxIndex = i;
    }
    return maxIndex;
}

public static void swap(double[] d, int i, int j) {
    double t = d[i];
    d[i] = d[j];
    d[j] = t;
}

public static void sort(double[] d) {
    for (int lastIndex = d.length-1; lastIndex >= 1; lastIndex--) {
        int maxIndex = maxIndex(d, lastIndex);
        swap(d, maxIndex, lastIndex);
    }
}
```

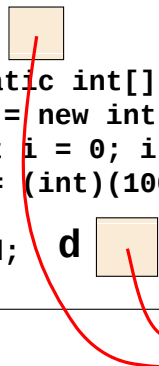


เมท็อดที่คืนอาร์เรย์

สิ่งที่คืนคือตำแหน่งอ้างอิงอาร์เรย์

```
public static void main(String[] args) {
    int[] x = randomIntArray(4);
    ... X
}

public static int[] randomIntArray(int n) {
    int[] d = new int[n];
    for (int i = 0; i < d.length; i++) {
        d[i] = (int)(100 * Math.random());
    }
    return d;
}
```



10 5 35 26

ถ้าเมท็อดต้องคืนข้อมูลมากกว่าหนึ่งตัว

- ถ้าคืนผลที่เป็นประเภทเดียวกัน
 - ให้เมท็อดสร้าง + คืนอาร์เรย์ที่เก็บผลตามช่องต่าง ๆ

```
public static int[] minmax(int a, int b) {
    if (a < b)
        return new int[] { a, b };
    else
        return new int[] { b, a };
}
```

- ให้ผู้เรียกเตรียม + ส่งอาร์เรย์ที่เก็บผลมาให้เมท็อดเติมผล

```
public static void minmax(int a, int b, int[] out) {
    if (a < b) {
        out[0] = a; out[1] = b;
    } else {
        out[0] = b; out[1] = a;
    }
}
```

- ถ้าคืนผลที่มีหลายประเภท : ใช้อ็อบเจกต์ (บทหลัง)

ตัวอย่าง : รากของ $ax^2+bx+c=0$

```
public static void main(String[] args) {
    Scanner kb = new Scanner(System.in);
    System.out.print("a = ");
    double a = kb.nextDouble();
    System.out.print("b = ");
    double b = kb.nextDouble();
    System.out.print("c = ");
    double c = kb.nextDouble();
    double[] r = quadraticRoots(a, b, c);
    System.out.println(r[0] + ", " + r[1]);
}

public static double[] quadraticRoots(
    double a, double b, double c) {
    double t = Math.sqrt(b * b - 4 * a * c);
    double x1 = (-b + t) / (2 * a);
    double x2 = (-b - t) / (2 * a);
    return new double[] {x1, x2};
}
```

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

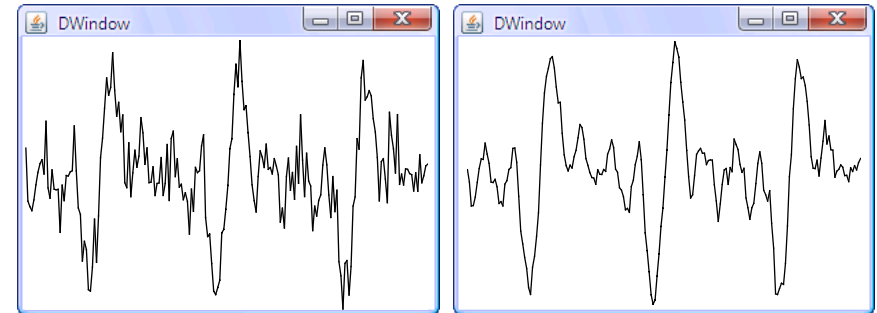
ตัวอย่าง : รากของ $ax^2+bx+c=0$

```
public static void main(String[] args) {
    Scanner kb = new Scanner(System.in);
    System.out.print("a = ");
    double a = kb.nextDouble();
    System.out.print("b = ");
    double b = kb.nextDouble();
    System.out.print("c = ");
    double c = kb.nextDouble();
    double[] r = new double[2];
    quadraticRoots(a, b, c, r);
    System.out.println(r[0] + " " + r[1]);
}

public static void quadraticRoots(
    double a, double b, double c, double[] r) {
    double t = Math.sqrt(b * b - 4 * a * c);
    r[0] = (-b + t) / (2 * a);
    r[1] = (-b - t) / (2 * a);
}
```

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

ค่าเฉลี่ยเคลื่อนที่ (moving average)



in	10	12	13	12	15	10
out	11	11.67	12.33	13.33	12.33	12.5

ตัวอย่าง : ค่าเฉลี่ยเคลื่อนที่

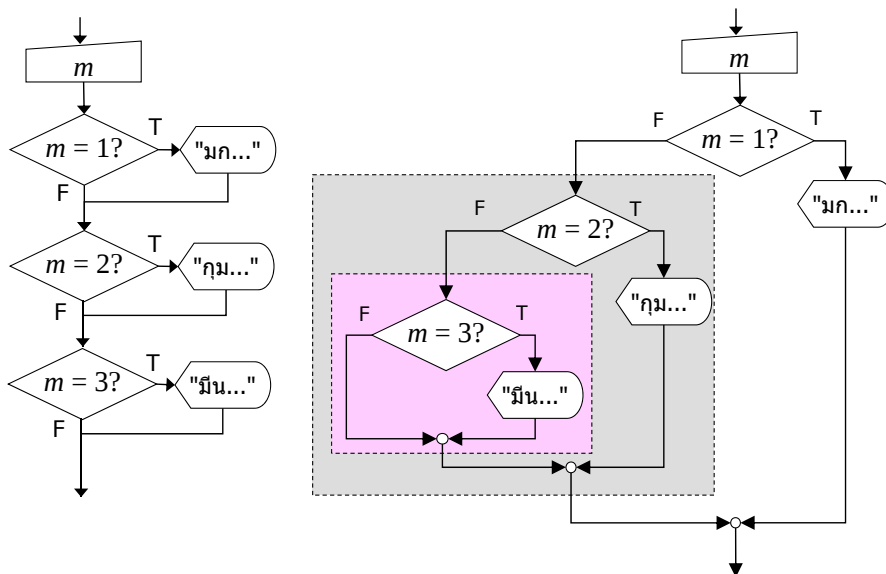
```
public static double[] movingAverage(double[] in) {
    int n = in.length;
    double[] out = new double[n];
    for (int i = 1; i < n - 1; i++) {
        out[i] = (in[i - 1] + in[i] + in[i + 1]) / 3.0;
    }
    out[0] = (in[0] + in[1]) / 2.0;
    out[n - 1] = (in[n - 2] + in[n - 1]) / 2.0;
    return out;
}
```

การเลือกและวงวน (ภาคสอง)

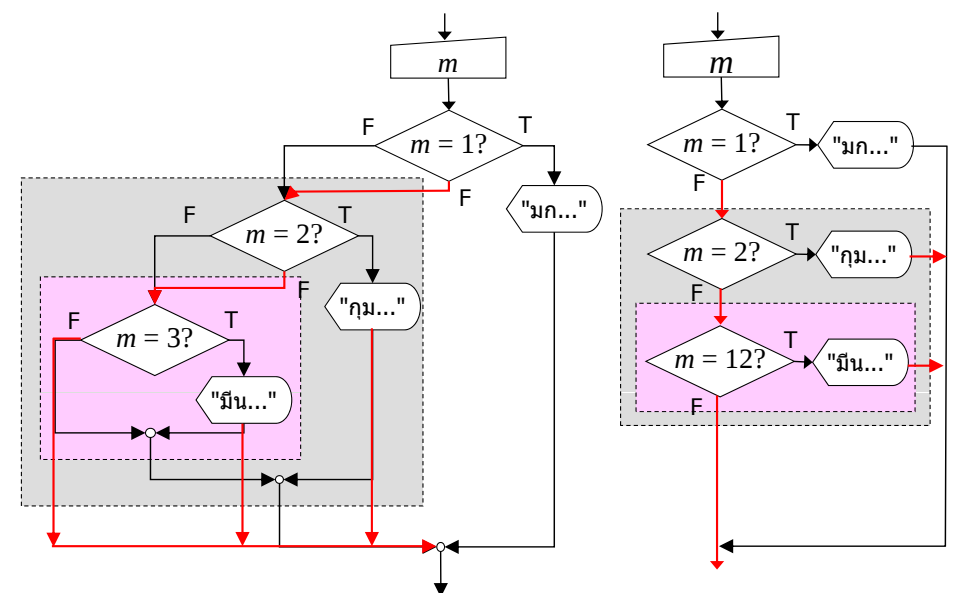
หัวข้อ

- if, if-else หลายชั้น
- การจัดการและการแปลงนิพจน์บูลีน :
- ข้อมูลและตัวแปรแบบ **boolean**
- วงวนหลายชั้น

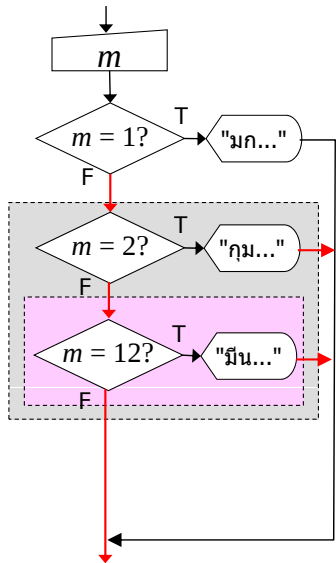
if-else หลายชั้น : แสดงชื่อเดือน



if-else หลายชั้น : แสดงชื่อเดือน



if-else หลายชั้น : แสดงชื่อเดือน



```

int m = kb.nextInt();

if (m == 1) {
    System.out.println("มก...");
} else {
    if (m == 2) {
        System.out.println("กุม...");
    } else {
        if (m == 3) {
            System.out.println("มีน...");
        }
    }
}
  
```

if-else หลายชั้น : แสดงชื่อเดือน

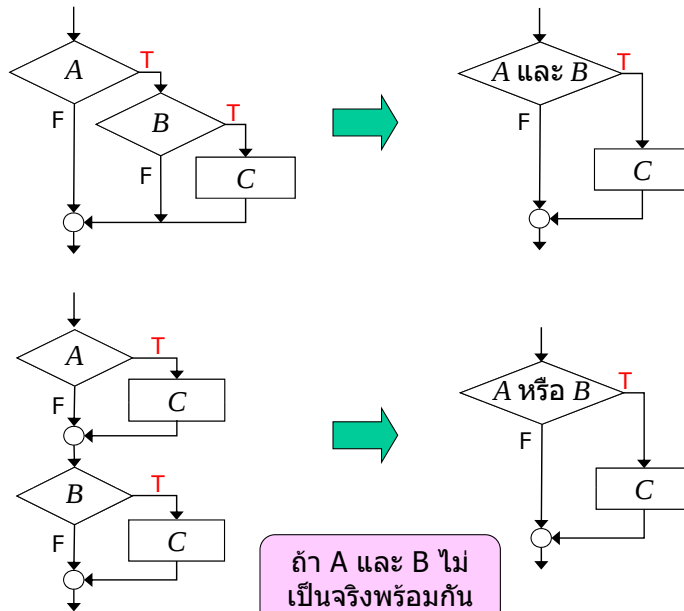
```

int m = kb.nextInt();
if (m == 1)
    System.out.println("มก...");
else
    if (m == 2)
        System.out.println("กุม...");
    else
        if (m == 3)
            System.out.println("มีน...");
  
```

```

int m = kb.nextInt();
if (m == 1) System.out.println("มก...");
else if (m == 2) System.out.println("กุม...");
else if (m == 3) System.out.println("มีน...");
  
```

การรวมเงื่อนไขการทดสอบ



และ หรือ ไม่

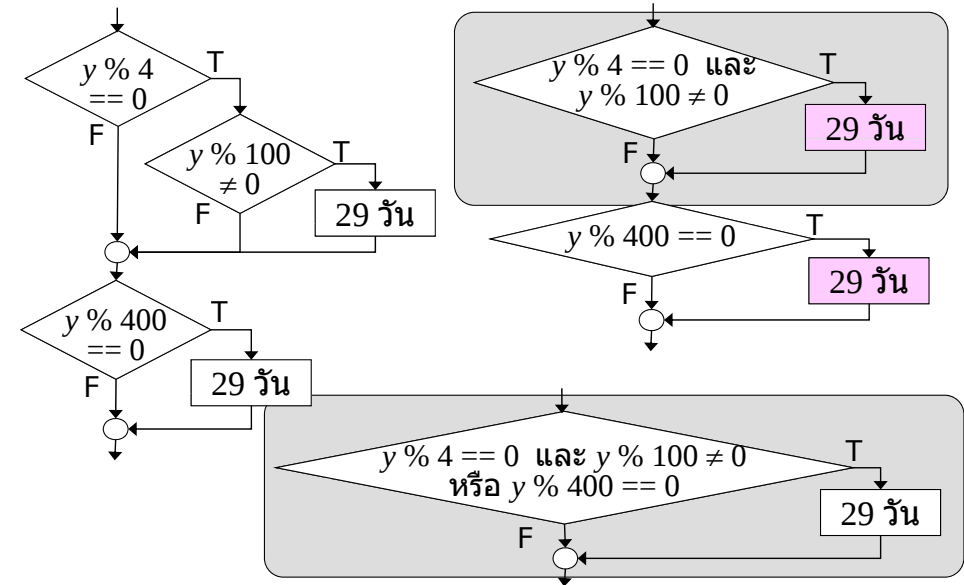
- ให้ P และ Q คือนิพจน์ตรรกะ

		และ	หรือ	ไม่
P	Q	P & Q	P Q	!P
false	false	false	false	true
false	true	false	true	true
true	false	false	true	false
true	true	true	true	false

ตัวอย่าง : เดือนกุมภาพันธ์มีกี่วัน

- เดือนกุมภาพันธ์ของปี y (ค.ศ.) มี 29 วันเมื่อ
 - เลขปีต้องหารด้วย 4 ลงตัว แต่หารด้วย 100 ไม่ลงตัว หรือไม่ก็กรณีที่เลขปีหารด้วย 400 ลงตัว เช่น
 - ปี 2002 ก.พ. มี 28 วัน (หารด้วย 4 ไม่ลงตัว)
 - ปี 2004 ก.พ. มี 29 วัน (หารด้วย 4 ลงตัว และหารด้วย 100 ไม่ลงตัว)
 - ปี 2100 ก.พ. มี 28 วัน (หารด้วย 100 ลงตัว)
 - ปี 2000 ก.พ. มี 29 วัน (หารด้วย 400 ลงตัว)
- ให้ y คือเลขปี (ค.ศ.)
 - $(y \% 4) == 0$ หารด้วย 4 ลงตัว ?
 - $(y \% 100) != 0$ หารด้วย 100 ไม่ลงตัว ?
 - $(y \% 400) == 0$ หารด้วย 400 ลงตัว ?

ตัวอย่าง : เดือนกุมภาพันธ์มีกี่วัน



ตัวอย่าง : เดือนกุมภาพันธ์มีกี่วัน

```

d = 28;
if (y%4 == 0) {
    if (y%100 != 0) {
        d = 29;
    }
}
if (y%400 == 0) {
    d = 29;
}
    
```

```

d = 28;
if ((y%4==0) && (y%100!=0)) {
    d = 29;
}
if (y%400 == 0) {
    d = 29;
}
    
```

และ

```

d = 28;
if (((y%4==0) && (y%100!=0)) || (y%400 == 0)) {
    d = 29;
}
    
```

หรือ

ตัวอย่าง : เมื่อกู้คืนจำนวนวันของเดือนปี

```

public static int days(int m, int y) {
    int d = 31;
    if (m == 2) {
        d = 28;
        if (((y%4 == 0) && (y%100 != 0)) || (y%400 == 0)) {
            d = 29;
        }
    } else if (m == 4 || m == 6 || m == 9 || m == 11) {
        d = 30;
    }
    return d;
}
    
```

ลำดับการทำงานของตัวดำเนินการ

1. วงเล็บ
2. เรียกใช้เมทอด
3. การดำเนินการ ! ++ -- -
4. การคำนวณ * / %
5. การคำนวณ + -
6. การเปรียบเทียบ < > <= >=
7. การเปรียบเทียบ == !=
8. ตรรกะ &&
9. ตรรกะ ||
10. การให้ค่า (ด้วยตัวดำเนินการ =)

```
y%4 == 0 && y%100 != 0 || y%400 == 0
```

ตัวอย่าง : ตัวมากที่สุดของ a กับ b

```
if (a > b)
    max = a;
else
    max = b;
```

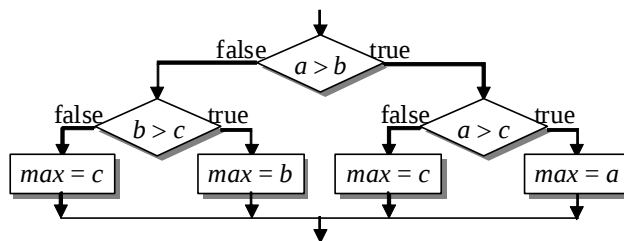
```
if (a >= b)
    max = a;
else
    max = b;
```

```
max = a;
if (b > a) max = b;
```

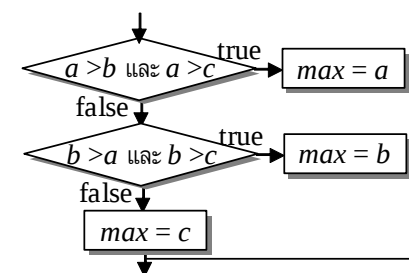
```
max = a;
if (b > max) max = b;
```

ตัวอย่าง : ตัวมากที่สุดของ a, b, c

```
if (a > b) {
    if (a > c) {
        max = a;
    } else {
        max = c;
    }
} else {
    if (b > c) {
        max = b;
    } else {
        max = c;
    }
}
```



ตัวอย่าง : ตัวมากที่สุดของ a, b, c



```
if (a > b && a > c)      max = a;
else if (b > a && b > c) max = b;
else                     max = c;
```

a=2, b=3, c=1 max = 3

a=1, b=1, c=3 max = 3

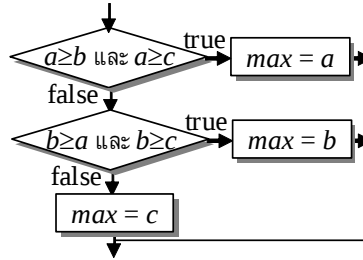
a=3, b=3, c=1 max = 1

ผิด

ผิด

ตัวอย่าง : ตัวมากที่สุดของ a, b, c

เปลี่ยน > เป็น ≥



```

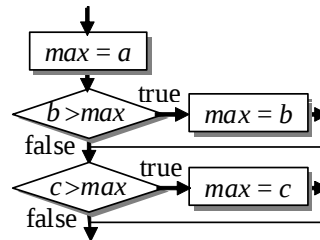
if (a >= b && a >= c)    max = a;
else if (b >= a && b >= c) max = b;
else                    max = c;
    
```

```

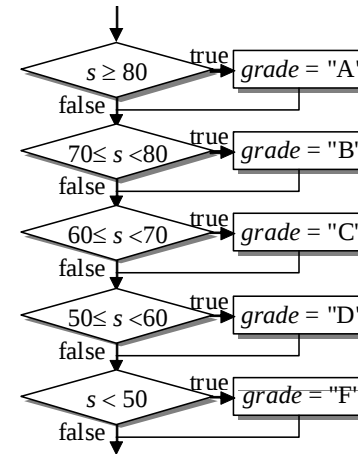
max = a;
if (b > max) max = b;
if (c > max) max = c;
    
```

ลองเขียนกรณีหาตัวมากที่สุดของ a, b, c, d, e,

f



ตัดเกรด

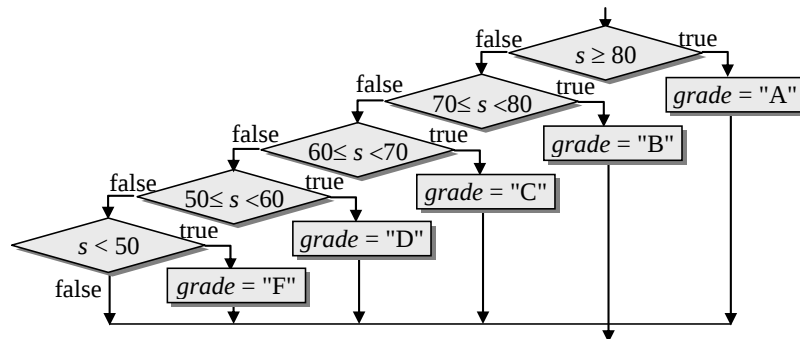


คะแนน	เกรด
$p \geq 80$	A
$70 \leq p < 80$	B
$60 \leq p < 70$	C
$50 \leq p < 60$	D
$p < 50$	F

```

if (s >= 80) grade = "A";
if (70 <= s && s < 80) grade = "B";
if (60 <= s && s < 70) grade = "C";
if (50 <= s && s < 60) grade = "D";
if (s < 50) grade = "F";
    
```

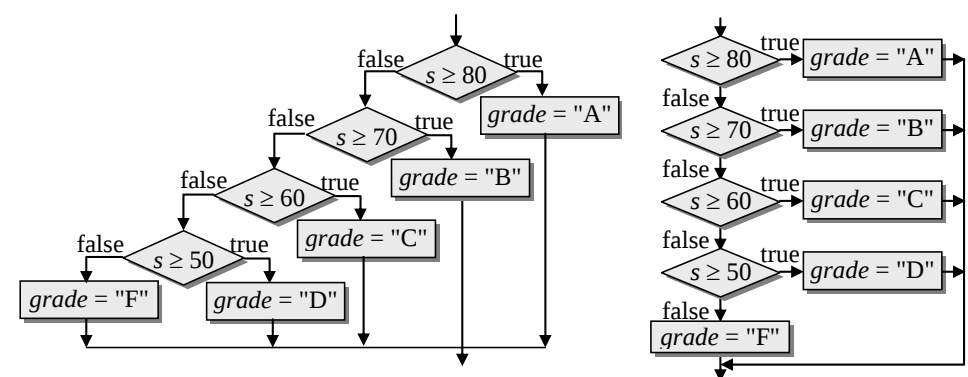
ตัดเกรด



```

if (s >= 80) grade = "A";
else if (70 <= s && s < 80) grade = "B";
else if (60 <= s && s < 70) grade = "C";
else if (50 <= s && s < 60) grade = "D";
else if (s < 50) grade = "F";
    
```

ตัดเกรด



```

if (s >= 80) grade = "A";
else if (70 <= s) grade = "B";
else if (60 <= s) grade = "C";
else if (50 <= s) grade = "D";
else grade = "F";
    
```

โจทย์ : ภาษีเงินได้

ขั้นเงินได้สุทธิ	อัตราภาษี
1 ถึง 100,000	5%
100,001 ถึง 500,000	10%
500,001 ถึง 1,000,000	20%
1,000,001 ถึง 4,000,000	30%
4,000,001 บาทขึ้นไป	37%

จงเขียนโปรแกรมรับเงินได้สุทธิทางแป้นพิมพ์
เพื่อคำนวณและแสดงภาษีเงินได้ส่วนบุคคลตาม
เกณฑ์ในตารางข้างบนนี้

ตัวอย่าง : ลูกบอลเต็งผนัง

```
while (true) {
    x = x + dx; y = y + dy;

    if ((x-r) <= 0 || (x+r) >= w.getWidth()) {
        dx = -dx;
    }

    if ((y-r) <= 0 || (y+r) >= w.getHeight()) {
        dy = -dy;
    }

    w.fade(0.3);
    w.fillEllipse(x, y, 2*r, 2*r);
    w.sleep(50);
}
```

การแปลงนิพจน์ตรรกะ

ให้ **a** และ **b** คือนิพจน์คำนวณ

!(a < b)	↔	(a >= b)
!(a > b)	↔	(a <= b)
!(a <= b)	↔	(a > b)
!(a >= b)	↔	(a < b)
!(a == b)	↔	(a != b)

การแปลงนิพจน์ตรรกะ

ให้ **p**, **q** และ **s** คือนิพจน์ตรรกะ

!!p	↔	p
(p&&q) (p&&s)	↔	p && (q s)
(p q) && (p s)	↔	p (q && s)
!(p && q)	↔	(!p) (!q)
!(p q)	↔	(!p) && (!q)

!(1<=m && m<=12)
!(1<=m) !(m<=12)
(1 > m) (m > 12)

ตัวอย่าง : นิพจน์บูลีน

- ถ้า a เป็นจำนวนเต็มบวกที่มีไม่เกิน 4 หลัก

```
if (a > 0 && a < 10000)
```

- ถ้า a มีค่าเป็นสี่เท่าหรือสามเท่าของ b

```
if (a == 3*b || a == 4*b)
```

- ถ้า a เป็นจำนวนคู่ หรือไม่ก็เป็นจำนวนเท่าของ 7

```
if (a % 2 == 0 || a % 7 == 0)
```

- ถ้า a เป็นจำนวนบวกไม่เกิน 100
ที่มีค่าไม่อยู่ในช่วง 10 ถึง 20

```
if ( ( 0 < a && a <= 100 ) &&  
    !(10 <= a && a <= 20 ) )
```

```
if ( ( 0 < a && a < 10 ) &&  
    (20 < a && a <= 100 ) )
```

ฝึกบ่อย

```
if (0 <= x <= 10)  
if x == 0  
if (x >= 0 && <= 10)  
if (x == 0 || == 1)  
if (x = 0)  
if (x => 0)
```



```
if (0 <= x && x <= 10)  
if (x == 0)  
if (x >= 0 && x <= 10)  
if (x == 0 || x == 1)  
if (x == 0)  
if (x >= 0)
```

```
if (x != 0 || x != 5)  
if (x < 3 && x > 9 )  
if (x == 0 && x != 0)  
if (x == 0 || x != 0)
```

นิพจน์ที่ได้ true
หรือ false เสมอ

ฝึกบ่อย

```
if (x == 0)  
    if (y == 0) z = 0;  
else if (a == 1)  
    z = 1;
```



```
if (x == 0)  
    if (y == 0)  
        z = 0;  
    else  
        if (a == 1) z = 1;
```



```
if (x == 0) {  
    if (y == 0) z = 0;  
} else if (a == 1)  
    z = 1;
```

ข้อมูลแบบ boolean

- นอกจาก : int, double, String
- ยังมี : boolean
 - เก็บได้สองค่าเท่านั้น : true กับ false
 - เก็บสถานะอะไรบางอย่างระหว่างการทำงาน
เพื่อใช้ทดสอบในภายหลัง

```
public static boolean isLeapYear(int y) {  
    return ((y%4 == 0) && (y%100 != 0)) || (y%400 == 0);  
}
```

```
public static boolean isLeapYear(int y) {  
    boolean b = ((y%4 == 0) && (y%100 != 0)) || (y%400==0);  
    return b;  
}
```

ตัวอย่าง : เมทีอดคืนจำนวนวันของเดือนปี

```
public static int days(int m, int y) {
    int d = 31;
    if (m == 2) {
        if ( isLeapYear(y) )
            d = 29;
        else
            d = 28;
    } else if (m == 4 || m == 6 || m == 9 || m == 11) {
        d = 30;
    }
    return d;
}
public static boolean isLeapYear(int y) {
    return (y%4 == 0) && (y%100 != 0) || (y%400 == 0);
}
```

โปรแกรมสรุปหัวข้อข่าวจาก CNN



อ่านข้อมูลจากอินเทอร์เน็ต

```
import jlab.URLStream;
import java.util.Scanner;

public class RSS {
    public static void main(String[] args) {
        String url =
            "http://rss.cnn.com/rss/cnn_topstories.rss";
        Scanner web = new Scanner(new URLStream(url));
        while (web.hasNext()) {
            String line = web.nextLine();
            System.out.println(line);
        }
        web.close();
    }
}

Scanner kb = new Scanner(System.in);
Scanner in = new Scanner(new File("c:/data.txt"));
Scanner web = new Scanner(new URLStream("http://cnn.com"));
```

URLStream เป็น
คลาสพิเศษของ JLab

ข้อมูลที่อ่านได้จาก cnn_topstories.rss

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet type="text/xsl" media="screen" href="/~d/...
<title>CNN.com</title>
<link>http://www.cnn.com/?eref=rss_topstories</link>
...
<image>
<title>CNN.com</title>
<link>http://www.cnn.com/?eref=rss_topstories</link>
...
<item>
<title>Emotional fight wages over Flight 93 memorial</title>
...
</item>
<item> <title>Thailand tourist's lung tissue 'gone'</title>
...
</item>
<item>
<title>Obama: Nominee 'would have restated' remark</title>
...
</item>
```

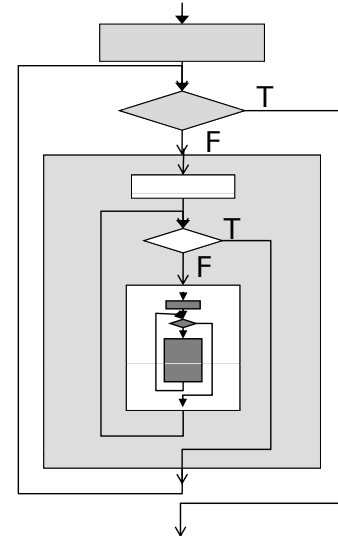
โปรแกรมสรุปหัวข้อข่าวจาก CNN

```
String url = "http://rss.cnn.com/rss/cnn_topstories.rss";
Scanner web = new Scanner(new URLStream(url));
boolean itemFound = false;
while (web.hasNext()) {
    String line = web.nextLine();
    if (line.indexOf("<item>") >= 0) itemFound = true;
    if (itemFound) {
        int start = line.indexOf("<title>");
        if (start >= 0) {
            int end = line.indexOf("</title>", start + 7);
            if (end >= 0) {
                String title = line.substring(start + 7, end);
                System.out.println( title );
            }
        }
        itemFound = false;
    }
}
web.close();
```

```
<item><title> ... </title>
...
```

```
<item>
<title> ... </title>
...
```

วงวนซ้อนวงวน



```
for (int i = 0; i < 10; i++) {
    for (int j = 20; j > 0; j--) {
        for (int k = 1; k <= 100; k++) {
            ...
        }
    }
}
```

```
int i = 0;
while (i < 10) {
    for (int j = 20; j > 0; j--) {
        for (int k = 1; k <= 100; k++) {
            ...
        }
    }
    i++;
}
```

วงวนซ้อนวงวน

i j
1,1
1,2
1,3
1,4
2,1
2,2
2,3
2,4
3,1
3,2
3,3
3,4

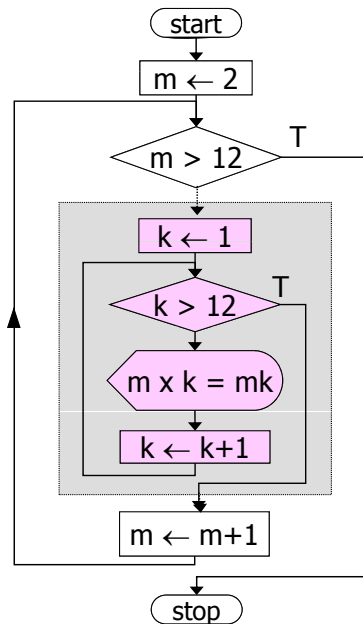
```
for (int i = 1; i <= 3; i++) {
    for (int j = 1; j <= 4; j++) {
        System.out.println(i + "," + j);
    }
}
```

วงวนซ้อนวงวน

i j
1,1
1,2
1,3
1,4
2,2
2,3
2,4
3,3
3,4

```
for (int i = 1; i <= 3; i++) {
    for (int j = i; j <= 4; j++) {
        System.out.println(i + "," + j);
    }
}
```

สูตรคูณแม่ 2 ถึง 12



```

int m = 2;
while (true) {
    if (m > 12) break;
    int k = 1;
    while (true) {
        if (k > 12) break;
        System.out.println(
            m + "x" + k + "=" + (m * k));
        k++;
    }
    m++;
}
  
```

สูตรคูณแม่ 2 ถึง 12

```

for (int m = 2; m <= 12; m++) {
    for (int k = 1; k <= 12; k++) {
        System.out.println(m + "x" + k + "\t = " + (m * k));
    }
    System.out.println("-----");
}
  
```

2x1	= 2
2x2	= 4
...	
2x8	= 16
2x9	= 18
2x10	= 20
2x11	= 22
2x12	= 24

3x1	= 3
3x2	= 6
3x6	= 18
...	

การเรียงลำดับข้อมูลแบบฟอง (bubble sort)

32	23	11	55	18
23	32	11	55	18
23	11	32	55	18
23	11	32	55	18
23	11	32	18	55
11	23	32	18	55
11	23	32	18	55
11	23	18	32	55
11	23	18	32	55
11	18	23	32	55
11	18	23	32	55

เมท็อดการเรียงลำดับข้อมูลแบบฟอง

```

public static void swap(double[] d, int i, int j) {
    double t = d[i];
    d[i] = d[j];
    d[j] = t;
}

public static void sort(double[] d) {
    for (int lastIndex = d.length-1; lastIndex >= 1;
        lastIndex--) {
        for (int k = 0; k < lastIndex; k++) {
            if (d[k] > d[k + 1]) swap(d, k, k + 1);
        }
    }
}
  
```

32	23	11	55	18
23	32	11	55	18
23	11	32	55	18
23	11	32	55	18
23	11	32	18	55
23	11	32	18	55
23	11	32	18	55
23	11	32	18	55
23	11	32	18	55
23	11	32	18	55
23	11	32	18	55

เมท็อดการเรียงลำดับข้อมูลแบบแบบฟอง

```
public static void sort(double[] d) {
    for (int lastIndex = d.length-1; lastIndex >= 1;
        lastIndex--) {
        boolean sorted = true;
        for (int k = 0; k < lastIndex; k++) {
            if (d[k] > d[k + 1]) {
                swap(d, k, k + 1);
                sorted = false;
            }
        }
        if (sorted) break;
    }
}
```

รอบในไม่มีการ swap
แสดงว่าเรียงแล้ว

32	3	11	25
3	32	11	25
3	11	32	25
3	11	25	
3	11	25	32

ทดสอบว่าอาเรย์ที่ได้รับมีค่าแตกต่างกันหมดทุกช่อง

- ลุยเปรียบเทียบข้อมูลทุกคู่ในอาเรย์ว่าแตกต่างกัน
- สมมติว่ามี 4 ช่อง ทุกคู่คือ
 - ~~(0,0)~~, (0,1), (0,2), (0,3), (1,0), ~~(1,1)~~, (1,2), (1,3)
 - (2,0), (2,1), ~~(2,2)~~, (2,3), (3,0), (3,1), (3,2), ~~(3,3)~~

```
public static boolean isUnique(int[] d) {
    for (int i = 0; i < d.length; i++) {
        for (int j = 0; j < d.length; j++) {
            if (i != j && d[i] == d[j]) return false;
        }
    }
    return true;
}
```

พบตัวซ้ำ เลิกเลย

ตรวจทุกคู่
แล้วไม่ซ้ำเลย

ทดสอบว่าอาเรย์ที่ได้รับมีค่าแตกต่างกันหมดทุกช่อง

- ลุยเปรียบเทียบข้อมูลทุกคู่ในอาเรย์ว่าแตกต่างกัน
- สมมติว่ามี 4 ช่อง ทุกคู่คือ
 - ~~(0,0)~~, (0,1), (0,2), (0,3), (1,0), ~~(1,1)~~, (1,2), (1,3)
 - (2,0), (2,1), ~~(2,2)~~, (2,3), (3,0), (3,1), (3,2), ~~(3,3)~~

```
public static boolean isUnique(int[] d) {
    boolean unique = true;
    for (int i = 0; i < d.length; i++) {
        for (int j = 0; j < d.length; j++) {
            if (i != j && d[i] == d[j]) {
                unique = false;
                break;
            }
        }
        if (!unique) break;
    }
    return unique;
}
```

บางคนชอบเขียนเมท็อดให้มี
จุด return ที่เดียว
ต้องใช้ตัวแปร boolean ช่วย

ทดสอบว่าอาเรย์ที่ได้รับมีค่าแตกต่างกันหมดทุกช่อง

- ลุยเปรียบเทียบข้อมูลทุกคู่ในอาเรย์ว่าแตกต่างกัน
- สมมติว่ามี 4 ช่อง ทุกคู่คือ
 - ~~(0,0)~~, (0,1), (0,2), (0,3), (1,0), ~~(1,1)~~, (1,2), (1,3)
 - (2,0), (2,1), ~~(2,2)~~, (2,3), (3,0), (3,1), (3,2), ~~(3,3)~~

```
public static boolean isUnique(int[] d) {
    boolean unique = true;
    for (int i = 0; unique && i < d.length; i++) {
        for (int j = 0; unique && j < d.length; j++) {
            if (i != j) unique = (d[i] != d[j]);
        }
    }
    return unique;
}
```

ย้ายเงื่อนไขการออกจากวงวน
ไปไว้ที่ for ให้โปรแกรมสั้น

unique เป็นเท็จ
จะกระโดดออกจากวงวนทันที

ทดสอบว่าอาเรย์ที่ได้รับมีค่าแตกต่างกันหมดทุกช่อง

- ลุยเปรียบเทียบข้อมูลทุกคู่ในอาเรย์ว่าแตกต่างกัน
- สมมติว่ามี 4 ช่อง ทุกคู่คือ
 - ~~(0,0)~~, (0,1), (0,2), (0,3), ~~(1,0)~~, ~~(1,1)~~, (1,2), (1,3)
 - ~~(2,0)~~, ~~(2,1)~~, ~~(2,2)~~, (2,3), ~~(3,0)~~, ~~(3,1)~~, ~~(3,2)~~, ~~(3,3)~~

```
public static boolean isUnique(int[] d) {
    boolean unique = true;
    for (int i = 0; unique && i < d.length; i++) {
        for (int j = i+1; unique && j < d.length; j++) {
            unique = (d[i] != d[j]);
        }
    }
    return unique;
}
```

หลีกเลี่ยงการตรวจสอบคู่ซ้ำ
ลดจำนวนรอบการทำงาน

ทดสอบว่าอาเรย์ที่ได้รับมีค่าแตกต่างกันหมดทุกช่อง

- เรียงลำดับข้อมูลในอาเรย์
- ถ้าไม่มีตัวติดกันเท่ากัน แสดงว่าต่างกันหมด
 - 2, 3, 4, 5, 6, 7, 8, 9
 - 2, 3, 4, 4, 5, 5, 5, 6

```
public static boolean isUnique(int[] d) {
    sort( d );
    for (int i = 0; i < d.length-1; i++) {
        if (d[i] == d[i+1]) return false;
    }
    return true;
}
```

อย่าเผลอเขียน d.length จะผิดตอน
if (d[i] == d[i+1]) ...

อาเรย์กับวงวน

จงเขียนเมทอดต่อไปนี้

- `public static int count(int[] d, int x)`
มี x ปรากฏในอาเรย์ d กี่ครั้ง
- `public static int mode(int[] d)`
คืนข้อมูลที่ปรากฏมากสุดใน d
- `public static boolean majority(int[] d)`
มีข้อมูลที่ปรากฏเป็นจำนวนเกินครึ่งใน d หรือไม่

อาร์เรย์ 2 มิติ

หัวข้อ

- การสร้างและการตั้งค่าเริ่มต้น
- อาร์เรย์กับวงวน for
- การประมวลผลเมทริกซ์
- การประมวลผลภาพ

อาร์เรย์หลายมิติ

```
int[] a = new int[4];  
a[2] = 3;          หนึ่งมิติ
```

	0	1	2	3
a	0	0	3	0

```
int[][] b = new int[3][4];  
b[1][3] = 4;        สองมิติ
```

	0	1	2	3
0	0	0	0	0
1	0	0	0	4
2	0	0	0	0

```
int[][][] c = new int[2][3][4];  
c[0][1][2] = 9;      สามมิติ
```

		0	0	0	0
	0	0	0	0	0
	0	0	9	0	0
	0	0	0	0	0

การตั้งค่าเริ่มต้น

```
int[] a = { 1, 2, 3, 4 };
```

	0	1	2	3
a	1	2	3	4

```
int[][] b = { {1, 2, 3, 4},  
              {5, 6, 7, 8},  
              {9, 10, 11, 12} };
```

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12

อาเรย์ กับ วงวน for

```
public static void print(int[] d) {
    for (int i = 0; i < d.length; i++) {
        System.out.print(d[i] + " ");
    }
    System.out.println();
}
```

จำนวนแถวแนวนอน

```
public static void print(int[][] d) {
    for (int i = 0; i < d.length; i++) {
        for (int j = 0; j < d[i].length; j++) {
            System.out.print(d[i][j] + " ");
        }
        System.out.println();
    }
}
```

จำนวนแถวแนวดิ่ง

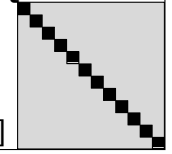
d เป็นอาเรย์ 2 มิติ
 d.length แทนจำนวนแถวแนวนอนของ d
 d[i].length แทนจำนวนช่องของอาเรย์แถวที่ i
 คิดแบบง่าย : d.length จำนวนแถว, d[0].length จำนวนคอลัมน์
 (ขนาดของอาเรย์ในแต่ละแถวไม่เท่ากันก็ได้ แต่เราจะไม่พูดถึง)

2110101

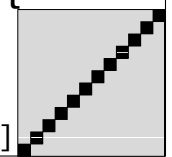
5

หาผลรวมของช่องในแนวทแยงมุมของอาเรย์ 2 มิติ

```
public static int sumDiag1(int[][] d) {
    int sum = 0;
    for (int i = 0; i < d.length; i++) {
        sum = sum + d[i][i];
    }
    return sum;
}
```



```
public static int sumDiag1(int[][] d) {
    int sum = 0;
    for (int i = 0; i < d.length; i++) {
        sum = sum + d[i][d.length-i-1];
    }
    return sum;
}
```

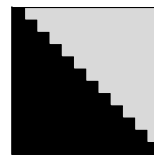


2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์ฯ (10/06/52)

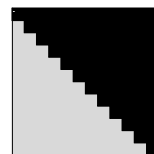
6

หาผลรวมของสามเหลี่ยมล่างและบนของอาเรย์ 2 มิติ

```
public static int sumLower(int[][] d) {
    int sum = 0;
    for (int i = 0; i < d.length; i++) {
        for (int j = 0; j <= i; j++) {
            sum = sum + d[i][j];
        }
    }
    return sum;
}
```



```
public static int sumUpper(int[][] d) {
    int sum = 0;
    for (int i = 0; i < d.length; i++) {
        for (int j = i; j < d.length; j++) {
            sum = sum + d[i][j];
        }
    }
    return sum;
}
```



2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์ฯ (10/06/52)

7

ตัวอย่าง : หา GPA ของนักเรียน

grade[j][i] เก็บเกรดของวิชา j ที่นักเรียนคนที่ i ได้

courses

0	→	"cs101"
1	→	"cs200"
2	→	"ee105"
3	→	"ee211"

students

grades

	↑	"สมบุญ"	↑	"สมชาย"	↑	"สมศรี"	↑	"สมหวัง"	↑	"สมปอง"	↑	"สมหมาย"	↑	"สมรักษ์"

	0	1	2	3	4	5	6
0	A	A	A		A	A	A
1	B	A		C	B+		C+
2	A		C+	A	A	A	B+
3	A	B	A	A	C	A	A

A, A, B → (4+4+3) / 3 = 3.67

gpa

3.75	3.67	3.50	3.33	3.38	4.00	3.50
------	------	------	------	------	------	------

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์ฯ (10/06/52)

8

เมท็อดคำนวณ GPA ของนักเรียนทุกคน

```
public static double[] studentGPA(String[][] grades) {
    double[] gpa = new double[grades[0].length];
    for (int i = 0; i < grades[0].length; i++) {
        gpa[i] = studentGPA(grades, i);
    }
    return gpa;
}
```

คำนวณ GPA ของ
นักเรียนคนที่ i

```
public static double studentGPA(
    String[][] grades, int i) {
    double sum = 0;
    int n = 0;
    for (int j = 0; j < grades.length; j++) {
        if (!grades[j][i].equals("")) {
            sum = sum + grade2point(grades[j][i]);
            n++;
        }
    }
    return sum / n;
}
```

หาคะแนนของวิชา j
นักเรียนคนที่ i

2110

9

เมท็อดเปลี่ยนเกรดเป็นคะแนน

```
public static double grade2point(String g) {
    if (g.equals("A")) return 4.0;
    else if (g.equals("B+")) return 3.5;
    else if (g.equals("B" )) return 3.0;
    else if (g.equals("C+")) return 2.5;
    else if (g.equals("C" )) return 2.0;
    else if (g.equals("D+")) return 1.5;
    else if (g.equals("D" )) return 1.0;
    else return 0.0;
}
```

```
public static double grade2point(String g) {
    String[] grades = { "A", "B+", "B", "C+",
                        "C", "D+", "D", "F" };
    double[] points = { 4, 3.5, 3, 2.5, 2, 1.5, 1, 0 };
    for (int k = 0; k < grades.length; k++) {
        if (grades[k].equals(g)) return points[k];
    }
    return 0.0;
}
```

21

10

ลองเขียนเอง : หา GPA ของวิชา

		students							course	
courses	grades	0	1	2	3	4	5	6	GPA	
	0	A	A	A		A	A	A	→	4.00
	1	B	A		C	B+		C+	→	3.00
	2	A		C+	A	A	A	B+	→	3.67
	3	A	B	A	A	C	A	A	→	3.57

ลองทำดู

จงเขียนเมท็อด

- void printEqualRows(int[][] d)
แสดงแถวใน d ที่มีค่าเหมือนกัน

1	2	4	3
9	6	5	7
1	2	4	3

- double saddlePoint(double[][] d)
คืนข้อมูลใน d ที่น้อยสุดในแถว
แต่มากที่สุดในคอลัมน์, ถ้าไม่มีคืน -1

1	2	4	3
9	6	5	7
1	2	4	3

การบวกเมทริกซ์

$$\begin{bmatrix} 2 & 3 & 4 \\ 5 & 6 & 7 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2+0 & 3+1 & 4+0 \\ 5+1 & 6+0 & 7+1 \end{bmatrix} = \begin{bmatrix} 2 & 4 & 4 \\ 6 & 6 & 8 \end{bmatrix}$$

a **b** **c**

$$c[i][j] = a[i][j] + b[i][j]$$

```
public static double[][] add(double[][] a, double[][] b) {
    int row = a.length, col = a[0].length;
    double[][] c = new double[row][col];
    for (int i = 0; i < row; i++) {
        for (int j = 0; j < col; j++) {
            c[i][j] = a[i][j] + b[i][j];
        }
    }
    return c;
}
```

การคูณเมทริกซ์กับจำนวนจริง

$$0.5 \begin{bmatrix} 2 & 4 & 6 \\ 8 & 2 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 1 & 2 \end{bmatrix}$$

s **a** **c**

$$c[i][j] = s * a[i][j]$$

```
public static double[][] add(double[][] a, double s) {
    int row = a.length, col = a[0].length;
    double[][] c = new double[row][col];
    for (int i = 0; i < row; i++) {
        for (int j = 0; j < col; j++) {
            c[i][j] = s * a[i][j];
        }
    }
    return c;
}
```

การคูณเวกเตอร์กับเมทริกซ์

$$\begin{bmatrix} a_0 & a_1 \end{bmatrix} \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} \\ b_{1,0} & b_{1,1} & b_{1,2} \end{bmatrix}$$

$$\begin{bmatrix} a_0 b_{0,0} + a_1 b_{1,0} & a_0 b_{0,1} + a_1 b_{1,1} & a_0 b_{0,2} + a_1 b_{1,2} \end{bmatrix}$$

$$c_i = \sum_{k=0}^{n-1} a_k b_{k,i}$$

```
public static double[] multiply(double[] a, double[][] b) {
    int row = b.length, col = b[0].length;
    double[] c = new double[col];
    for (int i = 0; i < col; i++) {
        for (int k = 0; k < row; k++) {
            c[i] = c[i] + a[k] * b[k][i];
        }
    }
    return c;
}
```

การคูณเมทริกซ์กับเมทริกซ์

$$\begin{bmatrix} a_{0,0} & a_{0,1} \\ a_{1,0} & a_{1,1} \end{bmatrix} \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} \\ b_{1,0} & b_{1,1} & b_{1,2} \end{bmatrix}$$

$$\begin{bmatrix} a_{0,0}b_{0,0} + a_{0,1}b_{1,0} & a_{0,0}b_{0,1} + a_{0,1}b_{1,1} & a_{0,0}b_{0,2} + a_{0,1}b_{1,2} \\ a_{1,0}b_{0,0} + a_{1,1}b_{1,0} & a_{1,0}b_{0,1} + a_{1,1}b_{1,1} & a_{1,0}b_{0,2} + a_{1,1}b_{1,2} \end{bmatrix}$$

$$c_{ij} = \sum_{k=0}^{n-1} a_{i,k} b_{k,j}$$

```
for (int k = 0; k < n; k++) {
    c[i][j] = c[i][j] + a[i][k] * b[k][j];
}
```

การคูณเมทริกซ์กับเมทริกซ์

$$\begin{bmatrix} a_{0,0} & a_{0,1} \\ a_{1,0} & a_{1,1} \end{bmatrix} \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} \\ b_{1,0} & b_{1,1} & b_{1,2} \end{bmatrix} = \begin{bmatrix} a_{0,0}b_{0,0} + a_{0,1}b_{1,0} & a_{0,0}b_{0,1} + a_{0,1}b_{1,1} & a_{0,0}b_{0,2} + a_{0,1}b_{1,2} \\ a_{1,0}b_{0,0} + a_{1,1}b_{1,0} & a_{1,0}b_{0,1} + a_{1,1}b_{1,1} & a_{1,0}b_{0,2} + a_{1,1}b_{1,2} \end{bmatrix}$$

$$= \begin{bmatrix} c_{0,0} & c_{0,1} & c_{0,2} \\ c_{1,0} & c_{1,1} & c_{1,2} \end{bmatrix} \quad c_{i,j} = \sum_{k=0}^{n-1} a_{i,k} b_{k,j}$$

```
public static double[][] multiply(double[][] a, double[][] b) {
    int row = a.length, col = b[0].length;
    int[][] c = new int[row][col];
    int n = b.length;
    for (int i = 0; i < row; i++) {
        for (int j = 0; j < col; j++) {
            for (int k = 0; k < n; k++) {
                c[i][j] = c[i][j] + a[i][k] * b[k][j];
            }
        }
    }
    return c;
}
```

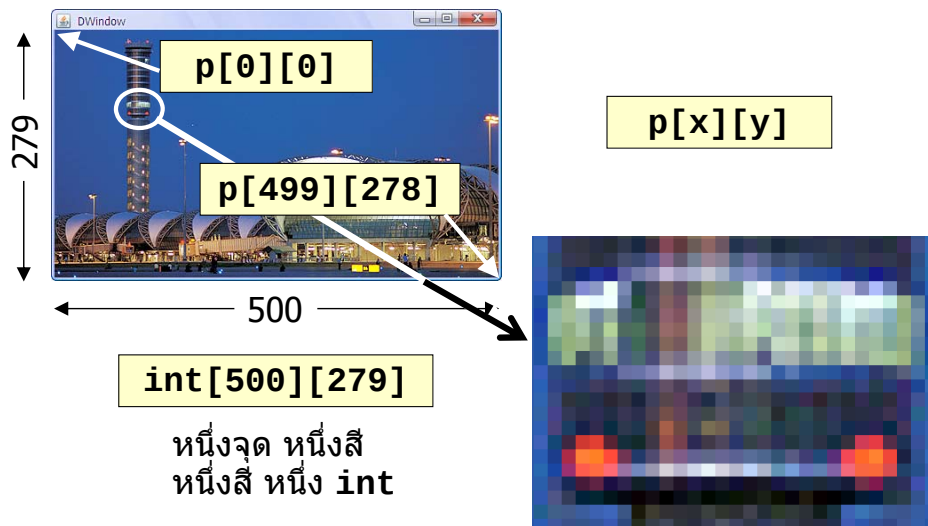
การอ่านแฟ้มภาพ

```
import javax.swing.*;

public class ImageDisplay {
    public static void main(String[] args) {
        JFrame w = new JFrame();
        w.loadImage("c:/java101/airport.jpg");
        w.fillEllipse(JFrame.YELLOW, 400, 70, 40, 40);
    }
}
```



แผนที่จุดภาพ (pixmap)



การอ่านและการตั้งแผนที่จุดภาพ

```
DWindow w = new DWindow();
w.loadImage("c:/java101/airport.jpg");
w.fillEllipse(JFrame.YELLOW, 400, 70, 40, 40);

int[][] p = w.getPixmap(); // คืนแผนที่จุดภาพ

...

w.setPixmap(p); // ตั้งแผนที่จุดภาพ
```

การพลิกภาพ (ซ้าย ↔ ขวา)



```
DWindow w = new DWindow();
w.loadImage("c:/java101/airport.jpg");
int[][] p = w.getPixmap();
int width = p.length, height = p[0].length;
for (int x = 0; x < width/2; x++) {
    for (int y = 0; y < height; y++) {
        int t = p[x][y];
        p[x][y] = p[width - x - 1][y];
        p[width - x - 1][y] = t;
    }
}
w.setPixmap(p);
```

สีมาตรฐานใน DWindow

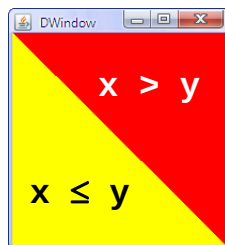


```
DWindow w = new DWindow(200, 200);
int[][] p = w.getPixmap();
for (int x = 0; x < p.length; x++) {
    for (int y = x; y < p[0].length; y++) {
        p[x][y] = DWindow.RED;
    }
}
w.setPixmap(p);
w.fillEllipse(DWindow.BLUE, 100, 100, 50, 50);
```

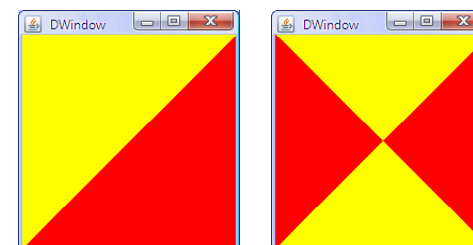
getPixmap, setPixmap

```
import jlab.graphics.DWindow;

public class FillColor {
    public static void main(String[] args) {
        DWindow w = new DWindow(200, 200);
        int[][] p = w.getPixmap();
        for (int x = 0; x < p.length; x++) {
            for (int y = 0; y < p[0].length; y++) {
                if (x > y)
                    p[x][y] = DWindow.RED;
                else
                    p[x][y] = DWindow.YELLOW;
            }
        }
        w.setPixmap(p);
    }
}
```



ลองเขียนดู



การผสมและการแยกสี

```
DWindow w = new DWindow(200, 200);
w.fillEllipse(DWindow.ORANGE, 100, 100, 20, 20);

int color = DWindow.mixRGB(255, 128, 0);
w.fillEllipse(color, 50, 50, 20, 20);
int r = DWindow.getR(color);
int g = DWindow.getG(color);
int b = DWindow.getB(color);
```

R G B

แม่สีมีค่าตั้งแต่ 0 -255

การทำภาพเนกาทีฟ



```
int[][] p = w.getPixmap();

for (int x = 0; x < p.length; x++)
    for (int y = 0; y < p[0].length; y++)
    {
        int r = DWindow.getR(p[x][y]);
        int g = DWindow.getG(p[x][y]);
        int b = DWindow.getB(p[x][y]);
        r = 255 - r;
        g = 255 - g;
        b = 255 - b;
        p[x][y] = DWindow.mixRGB(r, g, b);
    }
```

เมท็อดทำภาพเนกาทีฟ

```
public static void main(String[] args) {
    DWindow w = new DWindow(200, 200);
    w.loadImage("c:/java101/airport.jpg");
    int[][] p = w.getPixmap();
    w.setPixmap(negative(p));
}

public static int[][] negative(int[][] p) {
    for (int x = 0; x < p.length; x++) {
        for (int y = 0; y < p[0].length; y++) {
            int r = DWindow.getR(p[x][y]);
            int g = DWindow.getG(p[x][y]);
            int b = DWindow.getB(p[x][y]);
            r = 255 - r; g = 255 - g; b = 255 - b;
            p[x][y] = DWindow.mixRGB(r, g, b);
        }
    }
    return p;
}
```

การทำภาพขาวดำ



```
int[][] p = w.getPixmap();

for (int x = 0; x < p.length; x++)
    for (int y = 0; y < p[0].length; y++)
    {
        int r = DWindow.getR(p[x][y]);
        int g = DWindow.getG(p[x][y]);
        int b = DWindow.getB(p[x][y]);
        int gray = (r + g + b) / 3;
        p[x][y] = DWindow.mixRGB(gray, gray, gray);
    }
```

เมท็อดทำภาพขาวดำ

```
public static void main(String[] args) {
    DWindow w = new DWindow(200, 200);
    w.loadImage("c:/java101/airport.jpg");
    int[][] p = w.getPixmap();
    w.setPixmap(gray(p));
}

public static int[][] gray(int[][] p) {
    for (int x = 0; x < p.length; x++) {
        for (int y = 0; y < p[0].length; y++) {
            int r = DWindow.getR(p[x][y]);
            int g = DWindow.getG(p[x][y]);
            int b = DWindow.getB(p[x][y]);
            int gray = (r + g + b) / 3;
            p[x][y] = DWindow.mixRGB(gray, gray, gray);
        }
    }
    return p;
}
```

คลาสและอ็อบเจกต์

ที่ผ่านมา : คลาสคือโปรแกรม

```
public class ClassGPA {  
    public static void main(...) {  
        ...  
    }  
  
    เมื่้อดอื่น ๆ  
    ทุกเมื่้อดมีคำว่า static class methods  
    (เมื่้อดประจำคลาส) static methods  
  
}
```

ยังมีอีกแบบ : คลาสคือประเภทข้อมูล

```
public class Ball {  
    ...  
}  
  
public class Point {  
    ...  
}  
  
public class Rectangle {  
    ...  
}  
  
public class Employee {  
    ...  
}  
  
public class Classroom {  
    ...  
}  
  
public class Product {  
    ...  
}  
  
public class ComplexNumber {  
    ...  
}  
  
public class Date {  
    ...  
}  
  
public class Currency {  
    ...  
}  
  
public class BankAccount {  
    ...  
}
```

ตัวอย่างคลาสที่เป็นประเภทข้อมูล : Ball

```
public class Ball {  
    ตัวแปรประจำอ็อบเจกต์  
    object variables  
  
    ตัวสร้าง  
    constructors  
  
    เมื่้อดประจำอ็อบเจกต์  
    object methods  
  
    ตัวแปรประจำคลาส  
    class variables  
    เมื่้อดประจำคลาส  
    class method  
}
```

แบบง่ายสุด : มีเฉพาะข้อมูลย่อย

```
public class Ball {  
    ตัวแปรประจำอ็อบเจกต์  
    object variables  
}
```

data members
attributes
fields

บรรยายเฉพาะข้อมูลย่อยภายใน

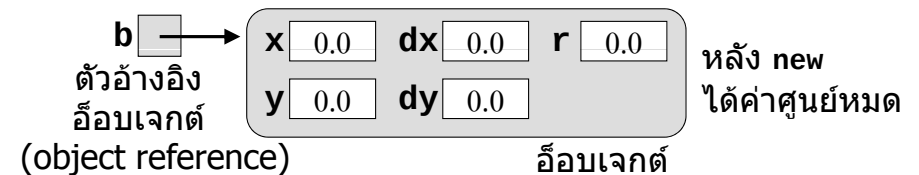
```
public class Ball {  
    public double x, y;  
    public double dx, dy;  
    public double r;  
}
```

สร้างอ็อบเจกต์ด้วย new

```
public class Ball {  
    public double x, y;  
    public double dx, dy;  
    public double r;  
}
```

ชื่อตัวแปร
ชื่อคลาส
สร้างอ็อบเจกต์ของ
คลาส **Ball**

```
Ball b = new Ball();
```



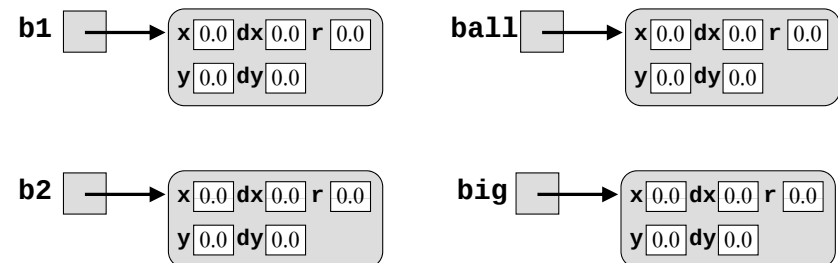
ค่าเริ่มต้นของตัวแปรประจำอ็อบเจกต์

- หลังจาก new, ค่าของตัวแปรประจำอ็อบเจกต์
 - จำนวน มีค่า 0
 - boolean มีค่า false
- ให้ค่าเริ่มต้นประกาศ object variable ได้

```
public class Ball {  
    public double x, y;  
    public double dx = 2, dy = 4;  
    public double r = 10;  
}
```

แต่ละอ็อบเจกต์มีตัวแปรประจำอ็อบเจกต์ของตัวเอง

```
Ball b1 = new Ball();  
Ball b2 = new Ball();  
Ball ball = new Ball();  
Ball big = new Ball();
```



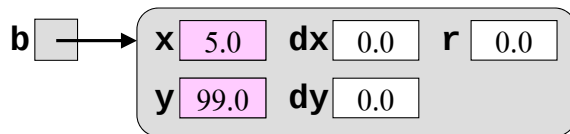
b1, b2, ball, big เป็นตัวอ้างอิงอ็อบเจกต์แบบ Ball
บางครั้งเราเรียก "อ็อบเจกต์ b1" แทนอ็อบเจกต์ที่ b1 อ้างอิงอยู่

การเปลี่ยนค่าของตัวแปรประจำอ็อบเจกต์

```
public class Ball {
    public double x, y;
    public double dx, dy;
    public double r;
}
```

```
Ball b = new Ball();
b.x = 5;
b.y = 99;
```

b.x อ่านว่า ตัวแปร x ของอ็อบเจกต์ b



เมทอดที่รับอ็อบเจกต์

- การส่งอ็อบเจกต์ให้เมทอด
คือการส่งตำแหน่งของอ็อบเจกต์ (ค่าของตัวอ้างอิง)
- ผู้ส่งกับผู้รับจึงอ้างอิงอ็อบเจกต์เดียวกัน

```
public static void main(String[] args) {
    Ball b = new Ball();
    moveRight(b, 5);
    ...
}
```

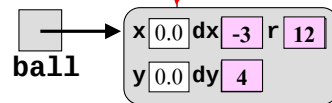
```
public static void moveRight(Ball ball, double d) {
    ball.x += d;
}
```

เมทอดที่คืนอ็อบเจกต์

- การคืนอ็อบเจกต์ให้ผู้เรียกเมทอด
คือการส่งตำแหน่งของอ็อบเจกต์คืนกลับไป

```
public static void main(String[] args) {
    Ball b = createRandomBall();
    ...
}
```

```
public static Ball createRandomBall() {
    Ball ball = new Ball();
    ball.r = random(10, 15);
    ball.dx = random(-5, 5);
    ball.dy = random(-5, 5);
    return ball;
}
```



ตัวแปรในเมทอดหาย เมื่อเมทอดทำงานเสร็จ แต่อ็อบเจกต์ที่สร้างในเมทอดไม่หาย ถ้ายังมีตัวแปรอ้างอิงมันอยู่

ตัวอย่าง : ลูกบอลแดงในวินโดว์

```
public static void main(String[] args) {
    DWindow w = new DWindow(250, 200);
    Ball b1 = createRandomBall();
    b1.x = 125; b1.y = 100;
    while (true) {
        w.fade(0.3);
        draw(w, b1);
        move(w, b1);
        w.sleep(30);
    }
}

public static void draw(DWindow w, Ball b) {
    w.fillEllipse(b.x, b.y, 2 * b.r, 2 * b.r);
}

public static void move(DWindow w, Ball b) {
    b.x = b.x + b.dx;
    b.y = b.y + b.dy;
    if (b.x + b.r > w.getWidth() || b.x < b.r) b.dx = -b.dx;
    if (b.y + b.r > w.getHeight() || b.y < b.r) b.dy = -b.dy;
}
```

```
public class Ball {
    public double x, y;
    public double dx, dy;
    public double r;
}
```

ให้คลาส Ball รับผิชอบการ draw และ move

```
DWindow w = new DWindow(250, 200);
Ball b1 = createRandomBall();
b1.x = 125; b1.y = 100;
while (true) {
    w.fade(0.3);
    Ball.draw(w, b1);
    Ball.move(w, b1);
    w.sleep(30);
}
```

เรียก draw ที่เขียน
ในคลาส Ball

```
public class Ball {
    public double x, y, dx, dy, r;

    public static void draw(DWindow w, Ball b) {
        w.fillEllipse(b.x, b.y, 2 * b.r, 2 * b.r);
    }
    public static void move(DWindow w, Ball b) {
        b.x = b.x + b.dx;
        b.y = b.y + b.dy;
        if (b.x + b.r > w.getWidth() || b.x < b.r) b.dx = -b.dx;
        if (b.y + b.r > w.getHeight() || b.y < b.r) b.dy = -b.dy;
    }
}
```

211 13

เขียน draw ในคลาส Ball ได้สองแบบ

class method

```
public class Ball {
    public double x, y, dx, dy, r;

    public static void draw(DWindow w, Ball b) {
        w.fillEllipse(b.x, b.y, 2 * b.r, 2 * b.r);
    }
}
```

Ball.draw(w, b1);

เรียก draw ที่เขียน
ในคลาส Ball

ต้องรับ Ball b มา เพื่อจะ
ใช้ค่าต่าง ๆ ของอ็อบเจกต์ b

object method

```
public class Ball {
    public double x, y, dx, dy, r;

    public void draw(DWindow w) {
        w.fillEllipse(x, y, 2 * r, 2 * r);
    }
}
```

b1.draw(w);

b1 เป็นอ็อบเจกต์ที่ถูก
เรียกใช้บริการ draw

ตัวแปรเหล่านี้คือตัวแปร
ของอ็อบเจกต์ที่ถูกเรียก

เรียก b1.draw(w), x ก็คือ b1.x

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (15/06/52)

14

class method กับ object method

- class method
 - มีคำว่า static ที่หัวเมทอด
 - รูปแบบการเรียก : ชื่อคลาส. ชื่อเมทอด (...)
- object method
 - ไม่มีคำว่า static ที่หัวเมทอด
 - รูปแบบการเรียก : ตัวอ้างอิงอ็อบเจกต์. ชื่อเมทอด (...)

บริการที่เกี่ยวข้องกับอ็อบเจกต์
มักเขียนเป็น object method

```
public class Ball {
    public double x, y, dx, dy, r;

    public static void draw(DWindow w, Ball b) {
        w.fillEllipse(b.x, b.y, 2 * b.r, 2 * b.r);
    }
}
```

class method

```
public class Ball {
    public double x, y, dx, dy, r;

    public void draw(DWindow w) {
        w.fillEllipse(x, y, 2 * r, 2 * r);
    }
}
```

object method

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ

15

องค์ประกอบของคลาส : ตัวสร้าง (constructor)

```
Ball b = new Ball();
b.x = 50; b.y = 50;
b.dx = 2; b.dy = 3;
b.r = 20;
```

```
Ball b = new Ball(50, 50, 2, 3, 20);
```

```
public class Ball {
    public double x, y;
    public double dx, dy;
    public double r;

    public Ball(double x1, double y1,
                double dx1, double dy1, double r1) {
        x = x1; y = y1; dx = dx1; dy = dy1; r = r1;
    }
    ...
}
```

ชื่อต้องเหมือนชื่อคลาส
ไม่มี static
ไม่มี ประสิทธิภาพ

2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (15/06/52)

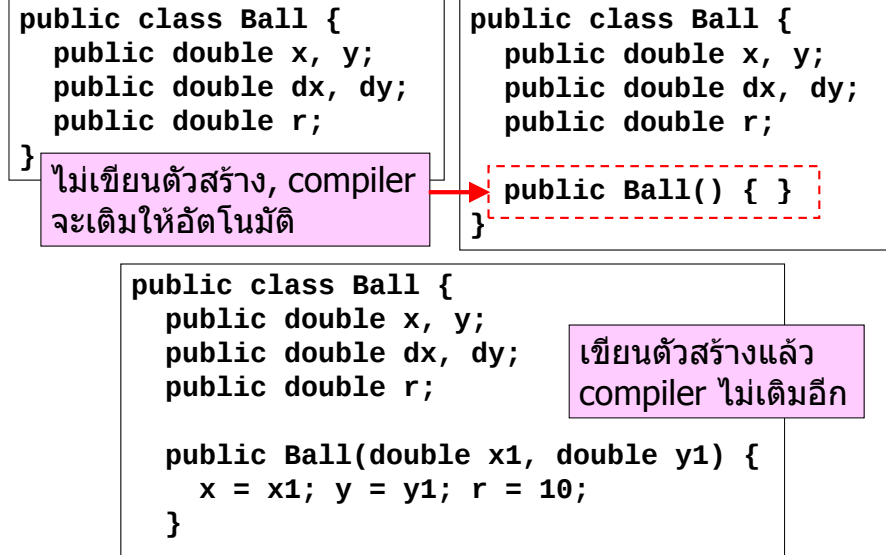
16

หนึ่งคลาสมีตัวสร้างหลายแบบก็ได้

```
public class Ball {
    public double x, y;
    public double dx, dy;
    public double r;

    public Ball() {
    }
    public Ball(double x1, double y1) {
        x = x1; y = y1; r = 10;
    }
    public Ball(double x1, double y1, double r1) {
        x = x1; y = y1; r = r1;
    }
    public Ball(double x1, double y1,
                double dx1, double dy1, double r1) {
        x = x1; y = y1; dx = dx1; dy = dy1; r = r1;
    }
    ...
}
```

Default Constructor (ตัวสร้างที่ระบบเติมให้ ถ้าไม่ได้เขียน)



ตัวอย่าง : BankAccount

```
public class BankAccount {
    public String id = "";
    public double balance = 0;

    public BankAccount(String newID, double initialBalance) {
        id = newID;
        balance = initialBalance;
    }
    public void withdraw(double amt) {
        if (amt > 0 && balance >= amt) balance = balance - amt;
    }
    public void deposit(double amt) {
        if (amt > 0) balance = balance + amt;
    }
}

BankAccount ba = new BankAccount("A11", 100);
ba.deposit(200);
ba.withdraw(50);
System.out.println(ba.balance);
ba.id = "99-99-9999-9"; // !!!
ba.balance = -1e6; // !!!
```

public ไม่ปลอดภัย, private ป้องกันได้

```
public class BankAccount {
    private String id = "";
    private double balance = 0;

    public BankAccount(String newID, double initialBalance) {
        id = newID;
        balance = initialBalance;
    }
    public void withdraw(double amt) {
        if (amt > 0 && balance >= amt) balance = balance - amt;
    }
    public void deposit(double amt) {
        if (amt > 0) balance = balance + amt;
    }
}

BankAccount ba = new BankAccount("A11", 100);
ba.id = "99-99-9999-9";
ba.balance = -1e6;
```

ใช้สมาชิกที่เป็น private ได้ เฉพาะในคลาสที่เขียนเท่านั้น

แปลไม่ผ่าน

อาจต้องให้บริการขอข้อมูลที่เป็น private

```
public class BankAccount {
    private String id;
    private double balance;
    ...
    public String getID() {
        return id;
    }
    public double getBalance() {
        return balance;
    }
}
```

```
BankAccount ba = new BankAccount("02-125-872", 0);
ba.deposit(100);
System.out.println(ba.getID() + ", " +
    ba.getBalance());
```

ตัวอย่าง : Dice

- เขียนคลาสเพื่อสร้างลูกเต๋า ให้บริการดังนี้
 - `int roll()` : โยนลูกเต๋า แล้วคืนหมายเลขหน้าที่ได้
 - `int getValue()` : คืนหมายเลขหน้าของลูกเต๋า

```
public class Dice {
    private int value;
    public Dice() {
        roll();
    }
    public int roll() {
        value = 1 + (int)(6 * Math.random());
        return value;
    }
    public int getValue() {
        return value;
    }
}
```



ตัวอย่าง : PiggyBank

- เขียนคลาสเพื่อสร้างกระปุกออมสินเก็บได้เฉพาะเหรียญ 1 บาท, 2 บาท, 5 บาท และ 10 บาท ให้บริการดังนี้
 - `PiggyBank()` : ตัวสร้างกระปุกออมสินเปล่า ๆ
 - `void add1(int c)` : หยอดเหรียญ 1 บาท c เหรียญ
 - `void add2(int c)` : หยอดเหรียญ 2 บาท c เหรียญ
 - `void add5(int c)` : หยอดเหรียญ 5 บาท c เหรียญ
 - `void add10(int c)` : หยอดเหรียญ 10 บาท c เหรียญ
 - `void clear()` : เทเหรียญทั้งหมดออกจากกระปุก
 - `int getTotal()` : คืนจำนวนเงินทั้งหมดที่เก็บในกระปุก

```
PiggyBank pb = new PiggyBank();
pb.add1(3); pb.add2(1); pb.add5(2); pb.add10(1);
pb.add1(2); System.out.println(pb.getTotal());
pb.clear(); System.out.println(pb.getTotal());
```

ตัวอย่าง : PiggyBank : ตัวแปรประจำอ็อบเจกต์

```
public class PiggyBank {
    private int one, two, five, ten;

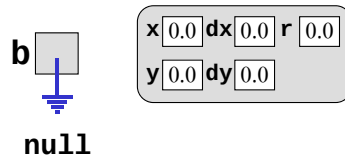
    public PiggyBank() {
        clear();
    }
    public void clear() {
        one = two = five = ten = 0;
    }
    public void add1(int coins) {one = one + coins;}
    public void add2(int coins) {two = two + coins;}
    public void add5(int coins) {five = five + coins;}
    public void add10(int coins) {ten = ten + coins;}
    public int getTotal() {
        return one + 2*two + 5*five + 10*ten;
    }
}
```



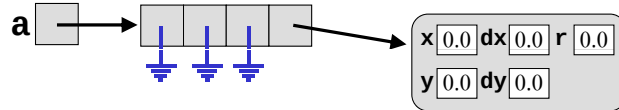
ลองปรับปรุงให้กระปุกมีความจุ
จะไม่เพิ่มให้ถ้าเต็มแล้ว

null : ค่าที่แทนการไม่อ้างอิงอ็อบเจกต์ใด

```
Ball b = new Ball();
b = null;
```



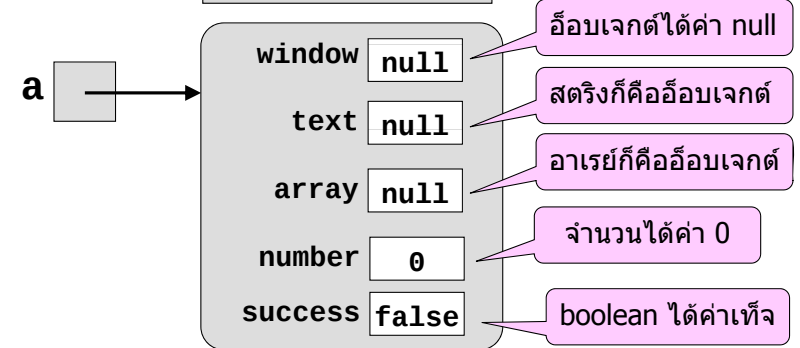
```
Ball[] a = new Ball[4]; // สร้างแค่อาเรย์
a[3] = new Ball();
```



ค่าเริ่มต้นของตัวแปรประจำอ็อบเจกต์

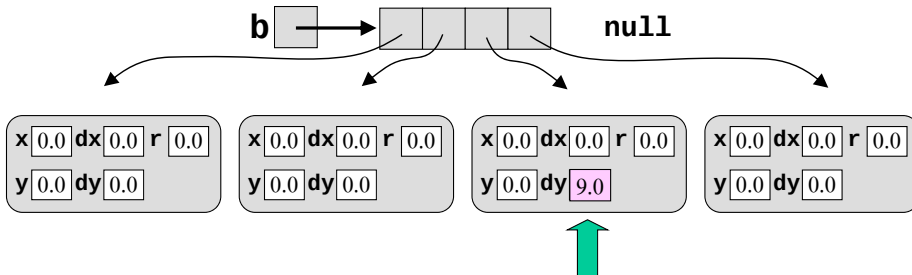
```
public class A {
    private DWindow window;
    private String text;
    private int[] array;
    private int number;
    private boolean success;
    ...
}
```

```
A a = new A();
```



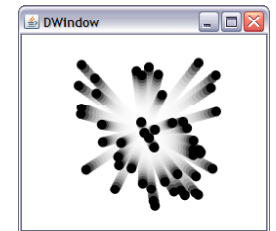
อาเรย์ของอ็อบเจกต์

```
Ball[] b = new Ball[4]; // สร้างแค่อาเรย์
for (int i = 0; i < b.length; i++) {
    b[i] = new Ball();
}
b[2].dy = 9;
```



ตัวอย่าง : ลูกบอล 50 ลูก (ใช้อาเรย์ 5 แถว)

```
DWindow w = new DWindow(250, 200);
int n = 50;
double[] x = new double[n], y = new double[n];
double[] r = new double[n];
double[] dx = new double[n], dy = new double[n];
for (int i = 0; i < n; i++) {
    x[i] = 125; y[i] = 100; r[i] = 5;
    dx[i] = -4 + 8 * Math.random();
    dy[i] = -4 + 8 * Math.random();
}
while (true) {
    w.fade(0.3);
    for (int i = 0; i < n; i++) {
        w.fillEllipse(x[i], y[i], 2 * r[i], 2 * r[i]);
        x[i] = x[i] + dx[i]; y[i] = y[i] + dy[i];
        if (x[i] + r[i] > 250) dx[i] = -dx[i];
        if (x[i] - r[i] < 0) dx[i] = -dx[i];
        if (y[i] + r[i] > 200) dy[i] = -dy[i];
        if (y[i] - r[i] < 0) dy[i] = -dy[i];
    }
    w.sleep(30);
}
```



สร้างอาเรย์

ให้ค่าเริ่มต้น

วาดและปรับค่า

คลาส Ball อีกครั้ง

```
public class Ball {
    public double x, y, dx, dy, r;

    public Ball(double x1, double y1,
                double dx1, double dy1, double r1) {
        x = x1; y = y1; dx = dx1; dy = dy1; r = r1;
    }
    public void draw(DWindow w) {
        w.fillEllipse(x, y, 2 * r, 2 * r);
    }
    public void move(DWindow w) {
        x = x + dx;
        y = y + dy;
        if (x + r > w.getWidth() || x < r) dx = -dx;
        if (y + r > w.getHeight() || y < r) dy = -dy;
    }
}
```

ตัวอย่าง : ลูกบอล 50 ลูก (ใช้อาเรย์ของ Ball 1 แถว)

```
public static void main(String[] args) {
    DWindow w = new DWindow(250, 200);
    Ball[] b = new Ball[50];
    for (int i = 0; i < b.length; i++) {
        b[i] = new Ball(125, 100,
                        random(-5, 5), random(-5, 5), 10);
    }
    while (true) {
        w.fade(0.3);
        for (int i = 0; i < b.length; i++) {
            b[i].draw(w);
            b[i].move(w);
        }
        w.sleep(30);
    }
}
public static double random(int a, int b) {
    return a + (b - a) * Math.random();
}
```

เรียก move ของลูกบอล b[i]

ตัวอย่าง : Card (ไพ่ 1 ใบ)

```
public class Card {
    private String suit; // โพธิ์ดำ โพธิ์แดง หลามตัด หัวใจ
    private int rank;    // 2, 3, ..., 14

    public Card(int r, String s) {
        suit = s;
        rank = r;
    }
    public int getRank() {
        return rank;
    }
    public String getSuit() {
        return suit;
    }
}
```



ตัวอย่าง : Deck (ไพ่ 1 สำรับ)

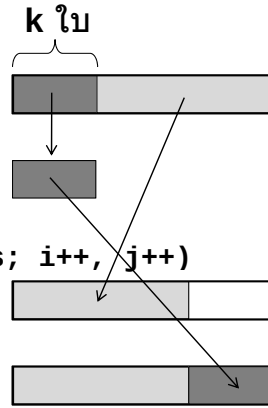
```
public class Deck {
    private Card[] deck;
    private int numberOfCards;

    public Deck() {
        deck = new Card[52];
        int k = 0;
        for (int i = 2; i <= 14; i++) {
            deck[k++] = new Card(i, "โพธิ์ดำ");
            deck[k++] = new Card(i, "โพธิ์แดง");
            deck[k++] = new Card(i, "หลามตัด");
            deck[k++] = new Card(i, "หัวใจ");
        }
        numberOfCards = 52;
    }
    public Card nextCard() {
        numberOfCards--;
        return deck[numberOfCards];
    }
}
```



ตัวอย่าง : Deck : ตัดไพ่

```
public class Deck {
    private Card[] deck;
    private int numberOfCards;
    // ...
    public void cut(int k) {
        Card[] temp = new Card[k];
        for (int i = 0; i < k; i++)
            temp[i] = deck[i];
        int j = 0;
        for (int i = k; i < numberOfCards; i++, j++)
            deck[j] = deck[i];
        for (int i = 0; i < k; i++, j++)
            deck[j] = temp[i];
    }
}
```



องค์ประกอบของคลาส : ตัวแปรประจำคลาส

- ตัวแปรประจำอ็อบเจกต์
 - แต่ละอ็อบเจกต์มีที่เก็บเป็นของตัวเอง เช่น ball.dx (ต้องมีอ็อบเจกต์ถึงจะมีตัวแปรแบบนี้)
- เมทอดประจำอ็อบเจกต์
 - บริการที่อ็อบเจกต์มีให้เรียกใช้ เช่น account.deposit(100) (ต้องมีอ็อบเจกต์ถึงจะเรียกเมทอดแบบนี้ได้)
- เมทอดประจำคลาส
 - เรียกใช้ได้เลยโดยไม่ต้องสร้างอ็อบเจกต์ Math.sin(x) (นำหน้าชื่อเมทอดด้วยชื่อคลาส)
- ตัวแปรประจำคลาส
 - ที่เก็บข้อมูลประจำตัวคลาส (ไม่ต้องสร้างอ็อบเจกต์ก็มีที่เก็บนี้ให้ใช้)

ใส่ static กำกับตัวแปรประจำคลาส

```
public class DayOfWeek {
    public static String getDayOfWeek(int d) {
        String[] dow = {"เสาร์", "อาทิตย์", "จันทร์",
                        "อังคาร", "พุธ", "พฤหัสบดี", "ศุกร์"};
        return dow[d];
    }
}
```

อาเรย์นี้ถูกสร้างทุกครั้งที่
getDayOfWeek ถูกเรียก

```
public class DayOfWeek {
    private static String[] dow =
        {"เสาร์", "อาทิตย์", "จันทร์", "อังคาร", "พุธ", "พฤหัสบดี", "ศุกร์"};

    public static String getDayOfWeek(int d) {
        return dow[d];
    }
}
```

อาเรย์นี้ถูกสร้างครั้งเดียวที่คลาส
DayOfWeek ถูกเรียกใช้

ใส่ static กำกับตัวแปรประจำคลาส

```
public class Coin {
    private static int numberOfCoins = 0;
    private int value;

    public Coin(int v) {
        value = v;
        numberOfCoins++;
    }

    public int getValue() {
        return value;
    }

    public static int getNumberOfCoins() {
        return numberOfCoins;
    }
    ...
}
```

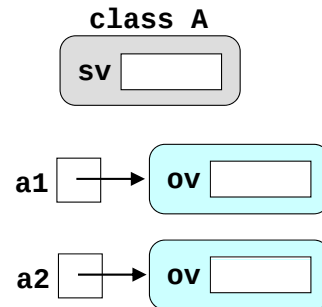
ถ้าลบ static ออก
จะเกิดอะไรขึ้น ?

นับจำนวนเหรียญที่ได้ผลิตมา

ตัวอย่างทดสอบความเข้าใจ : class & obj. var.

```
public class A {
    private static int sv;
    private          int ov;
    ...
}
```

```
A.sv = 1;
A a1 = new A();
A a2 = new A();
A.sv++;
a1.ov++;
a2.ov++;
a1.sv++;
a2.sv++;
A.ov = 9 // wrong
```



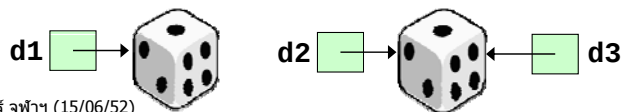
ตัวอย่างที่เราเคยใช้มา

- Class methods
 - Math.sin(x), Math.random(), ...
- Class variables
 - System.in, System.out, DWindow.RED, ...
- Object methods
 - kb.nextInt(), System.out.println(), w.sleep(30), ...
- Object variables
 - มีน้อย เพราะมักให้เป็น private

การใช้ == กับอ็อบเจกต์

- ถ้า a และ b เป็นตัวแปรอ้างอิงอ็อบเจกต์
- a == b เป็นการทดสอบว่า
 - a และ b อ้างอิงตำแหน่งเดียวกันหรือไม่
 - a และ b อ้างอิงอ็อบเจกต์ตัวเดียวกันหรือไม่
- การเปรียบเทียบความเท่ากันของสตริง จึงไม่ใช่ == ("a" == "A".toLowerCase() ได้ false)
 - ต้องใช้เป็นเมทอดแทน (s1.equals(s2))

```
Dice d1 = new Dice();
Dice d2 = new Dice();
Dice d3 = d2;
System.out.println( (d1 == d2) ); // false
System.out.println( (d2 == d3) ); // true
```



equals ของ Dice

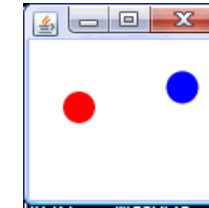
```
public class Dice {
    private int value;
    public Dice() { roll(); }
    public int getValue() { return value; }
    public int roll() {
        value = 1 + (int)(6 * Math.random());
        return value;
    }
    public boolean equals(Dice d) {
        return value == d.getValue();
    }
}
```

```
Dice d1 = new Dice();
Dice d2 = new Dice();
int c = 1;
while (!d1.equals(d2)) {
    d1.roll();
    d2.roll();
    c++;
}
System.out.println( c );
```

การสร้างคลาสใหม่ ด้วยการรับทอด

สร้างคลาสใหม่จากเก่า

- ทำสำเนา source code ของคลาสเดิม แล้วแก้ไข
- เขียนคลาสใหม่เก็บอ็อบเจกต์ของคลาสเดิม
- เขียนคลาสใหม่ให้ขยายความสามารถจากคลาสเดิม



วิธีที่ 1 : ลูกบอลสีดำ → ตั้งสีได้

```
import jlab.graphics.DWindow;

public class BallC {
    public double x, y;
    public double dx, dy;
    public double r;
    public int color;

    public BallC(double x1, double y1, double dx1, double dy1,
                  double r, int c) {
        x = x1; y = y1; dx = dx1; dy = dy1; r = r1; color = c;
    }
    public void draw(DWindow w) {
        w.fillEllipse(color, x, y, 2 * r, 2 * r);
    }
    public void move(DWindow w) {
        x = x + dx;
        y = y + dy;
        if (x + r > w.getWidth() || x < r) dx = -dx;
        if (y + r > w.getHeight() || y < r) dy = -dy;
    }
}
```

ทำสำเนาห้สตันฉบับของ
Ball แล้วแก้ไขได้ BallC

วิธีที่ 2 : ลูกบอลสีดำ → ตั้งสีได้

```
import jlab.graphics.DWindow;

public class BallC {
    public Ball ball;
    public int color;

    public BallC(double x1, double y1, double dx1, double dy1,
                  double r1, int c1) {
        ball = new Ball(x1, y1, dx1, dy1, r1);
        color = c1;
    }
    public void draw(DWindow w) {
        w.fillEllipse(color, ball.x, ball.y, 2*ball.r, 2*ball.r);
    }
    public void move(DWindow w) {
        ball.move(w);
    }
}
```

สร้างอ็อบเจกต์ของเดิมเก็บไว้ใช้งาน
เขียนเพิ่มเท่าที่จำเป็น

ถ้า Ball เปลี่ยนพฤติกรรมการ move
BallC ก็เปลี่ยนตามโดยอัตโนมัติ

วิธีที่ 3 : ลูกบอลสีดำ → ตั้งสีได้

```
import jlab.graphics.DWindow;
```

```
public class BallC extends Ball {
```

```
public Ball ball;
public int color;
```

```
public BallC(double x1, double y1, double dx1, double dy1,
             double r1, int c1) {
    super(x1, y1, dx1, dy1, r1);
    color = c1;
```

super(...) แทนตัวสร้างของ
ที่คลาสนี้ไป extends มา

```
public void draw(DWindow w) {
    w.fillEllipse(color, super.x, super.y, 2*super.r, 2*super.r);
}
```

```
public void move(DWindow w) {
    super.move(w);
}
```

super. แทนของที่
คลาสนี้ไป extends มา

ถ้าเหมือนของคลาสที่ extends มา
ทุกประการ ไม่ต้องเขียนก็ได้

เขียนคลาสใหม่ให้ "ขยาย"
ลักษณะจากคลาสเก่า

วิธีที่ 3 : ลูกบอลสีดำ → ตั้งสีได้

```
import jlab.graphics.DWindow;
```

```
public class BallC extends Ball {
```

```
public int color;
```

```
public BallC(double x1, double y1, double dx1, double dy1,
             double r1, int c1) {
    super(x1, y1, dx1, dy1, r1);
    color = c1;
```

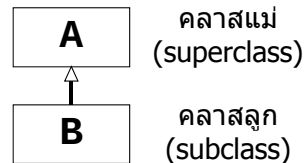
```
public void draw(DWindow w) {
    w.fillEllipse(color, super.x, super.y, 2*super.r, 2*super.r);
}
```

ถ้า Ball เปลี่ยนพฤติกรรม move
BallC ก็เปลี่ยนตามโดยอัตโนมัติ

เขียนคลาสใหม่ให้ "ขยาย"
ลักษณะจากคลาสเก่า

การรับทอด (Inheritance)

```
public class B extends A {
    ...
}
```



- B รับทอดลักษณะสมบัติจาก A
 - อ็อบเจกต์ของ B มีตัวแปรประจำอ็อบเจกต์ทุกตัวของ A
 - B ใช้ตัวแปรและเมทอด (ประจำคลาสและประจำอ็อบเจกต์) ทั้งหมดของ A ที่ไม่เป็น private ได้
 - B ไม่ได้ตัวสร้างของ A (แต่เรียกใช้ได้)
- B เพิ่มตัวแปรใหม่และเพิ่มเมทอดใหม่ได้
- B เขียนเมทอดใหม่แทนของที่ได้รับทอดได้

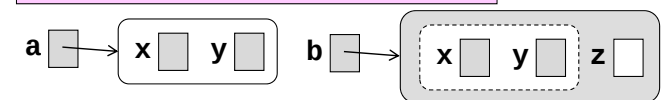
override

B extends A (แบบง่ายสุด)

```
public class A {
    public int x;
    private int y;
    public A() {}
    public A(int x1) {x = x1;}
    public void m1() {...}
    public int m2(double z) {...}
    private void m3() {...}
}
```

```
public class B extends A {
    ...
}
```

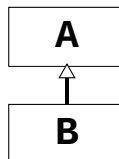
```
A a = new A();
B b = new B();
```



- อ็อบเจกต์ของ B มีตัวแปร x และ y (แต่ใช้ y ไม่ได้)
- เราเรียกเมทอด m1 และ m2 กับอ็อบเจกต์ของ B ได้
- คลาส B มีตัวสร้าง B(){} ที่ระบบเติมให้

B extends A

```
public class A {
    public int x;
    private int y;
    public A() {}
    public A(int x1) {x = x1;}
    public void m1() {...}
    public int m2(double z) {...}
    private void m3() {...}
}
```



```
public class B extends A {
    private int z;
    public B(int z1) {z = z1;}
    public void m1() {...}
    public void m4() {...}
}
```

m1 นี้ แทน
m1 ของแม่

เพิ่ม m4

- อ็อบเจกต์ของ B มีตัวแปร x, y, และ z (แต่ใช้ y ไม่ได้)
- เราเรียกเมทอด m1, m2, และ m4 กับอ็อบเจกต์ของ B ได้
- คลาส B มีตัวสร้าง B(int z1){...} ไม่มี B(){}

การเรียกตัวสร้างของแม่ : super(...)

- เราควรเขียนตัวสร้างเพื่อตั้งค่าเริ่มต้นของตัวแปรประจำอ็อบเจกต์ เมื่อมีการ new อ็อบเจกต์
- ลูกไม่มีตัวสร้างของแม่ แต่ลูกเรียกใช้ตัวสร้างของแม่ได้
- ใช้ super(...) เพื่อเรียกตัวสร้างของแม่
- ที่บรรทัดแรกในตัวสร้างของลูก ควรเรียกตัวสร้างของแม่ ถ้าไม่เรียก ระบบจะเติม super() ให้

```
public class A {
    private int x;

    public A(int x1) {
        x = x1;
    }
    ...
}
```

```
public class B extends A {
    private int z;

    public B(int z1) {
        super();
        z = z1;
    }
    ...
}
```

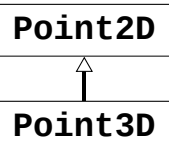
ผิด : แม่ไม่มี A()

super(0); ถูกต้อง

เขียน Point3D จาก Point2D

```
public class Point2D {
    public int x, y;

    public Point2D(int x1, int y1) {
        x = x1; y = y1;
    }
    ...
}
```



```
public class Point3D extends Point2D {
    public int z;

    public Point3D(int x1, int y1, int z1) {
        super(x1, y1);
        z = z1;
    }
    ...
}
```

public, private, protected

- **public** : ใคร ๆ ก็ใช้ได้
- **private** : ใช้ได้เฉพาะภายในคลาสตัวเอง
- **protected** : ให้คลาสลูก ๆ หลาน ๆ ใช้
- เขียนกำกับสมาชิกในคลาส

```
public class A {
    public int x;
    private int y;
    protected int z;

    protected A() {}
    public A(int x1) {x = x1;}

    public void m1() {...}
    private void m2() {...}
    protected void m3() {...}
}
```

เขียน CrookedDice จาก Dice

```
public class Dice {
    protected int value;
    public Dice() { roll(); }
    public int roll() {
        value = 1 + (int)(6 * Math.random());
        return value;
    }
    public int getValue() {
        return value;
    }
}

public class CrookedDice extends Dice {
    public CrookedDice() {}
    public int roll() {
        if (Math.random() < 0.9)
            value = 1;
        else
            value = 2 + (int)(5 * Math.random());
        return value;
    }
}
```

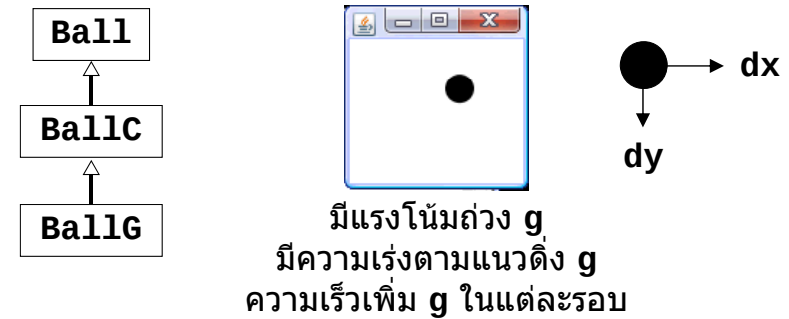
public ไม่ปลอดภัย
private ลูกใช้ไม่ได้
ต้องใช้ protected

ระบบเดิม
super();

roll นี้ แทน
roll ของแม่



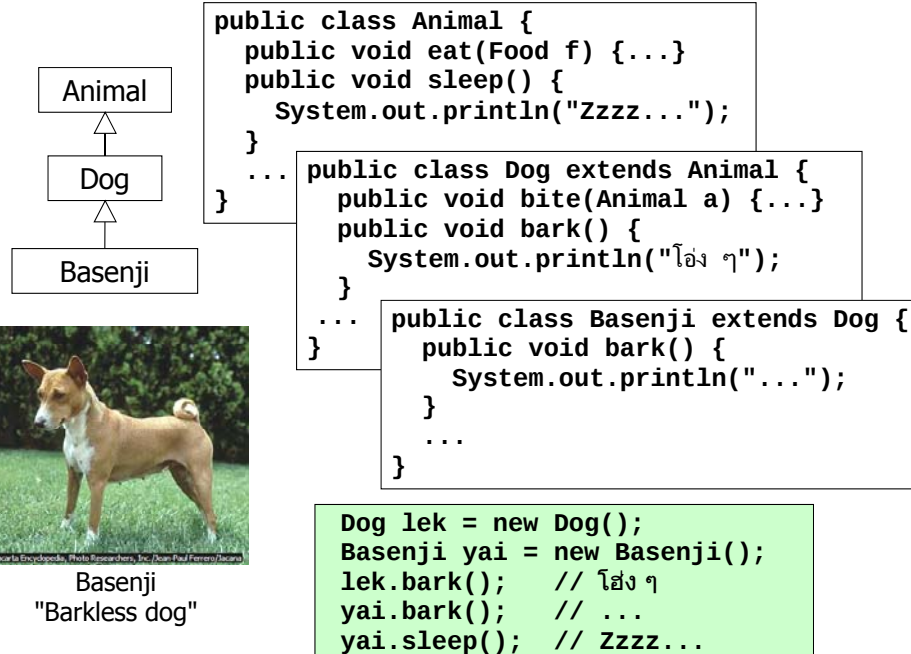
เขียน BallG จาก BallC



```
public class BallG extends BallC {
    private double gravity = 2;
    ...
    public void move(DWindow w) {
        dy += gravity;
        super.move(w);
        y = Math.min(y, w.getHeight() - r);
    }
}
```

ป้องกันลูกบอลจมน้ำ

Animal, Dog, Basenji



การสร้างคลาสใหม่แบบรับทอด

- เขียนคลาสใหม่ให้ extends จากคลาสเดิม
 - เขียน ตัวสร้าง ที่ต้องการให้บริการ
 - เพิ่ม ตัวแปรประจำอ็อบเจกต์ใหม่ที่เป็น
 - เพิ่ม เมทอดประจำอ็อบเจกต์ใหม่ ที่ superclass ไม่มี
 - แทน เมทอดประจำอ็อบเจกต์เดิมของ superclass ที่อยากเปลี่ยน

PiggyBank : กระปุกเก็บเหรียญ

```
public class PiggyBank {
    private int one, two, five, ten;

    public PiggyBank() {
        clear();
    }
    public void clear() {
        one = two = five = ten = 0;
    }
    public void add1(int coins) {one = one + coins;}
    public void add2(int coins) {two = two + coins;}
    public void add5(int coins) {five = five + coins;}
    public void add10(int coins) {ten = ten + coins;}
    public int getTotal() {
        return one + 2*two + 5*five + 10*ten;
    }
}
```



เขียน MoneyBox จาก PiggyBank

กล่องเก็บเงิน เก็บทั้งเหรียญและธนบัตร

```
public class MoneyBox extends PiggyBank {
    private int b20, b50, b100, b500, b1000;

    public MoneyBox() { super(); }
    public void clear() {
        super.clear();
        b20 = b50 = b100 = b500 = b1000 = 0;
    }
    public void add20(int b) { b20 = b20 + b; }
    public void add50(int b) { b50 = b50 + b; }
    public void add100(int b) { b100 = b100 + b; }
    public void add500(int b) { b500 = b500 + b; }
    public void add1000(int b) { b1000 = b1000 + b; }
    public int getTotal() {
        return super.getTotal() + 20*b20 + 50*b50 +
            100*b100 + 500*b500 + 1000*b1000;
    }
}
```



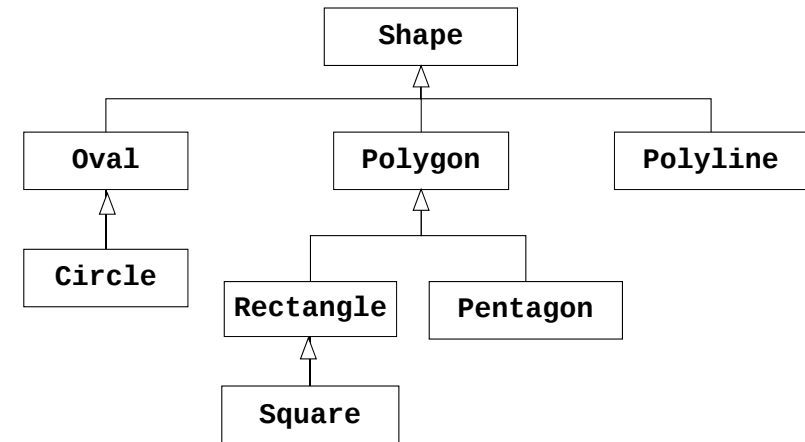
เขียน DWin จาก DWindow ให้มี fillCircle

```
import jlab.graphics.DWindow;

public class DWin extends DWindow {
    public DWin(double w, double h) {
        super(w, h);
    }
    public void fillCircle(double x, double y,
                           double r) {
        fillEllipse(x, y, 2 * r, 2 * r);
    }
    public void fillCircle(int color, double x,
                           double y, double r) {
        fillEllipse(color, x, y, 2 * r, 2 * r);
    }
}
```

ไม่ต้องมีรหัสต้นฉบับ
ของ DWindow

Inheritance Hierarchy



เมทอดแบบเรียกซ้ำ (recursive)

ความสัมพันธ์เวียนเกิด (Recurrences)

- การเขียนความสัมพันธ์ของจำนวนเต็มในลำดับ

$$0, 1, 2, 3, 4, \dots \quad a_n = a_{n-1} + 1 \quad \text{เมื่อ } n > 0, a_0 = 0$$

$$3, 5, 7, 9, 11, \dots \quad a_n = a_{n-1} + 2 \quad \text{เมื่อ } n > 0, a_0 = 3$$

$$0, 1, 3, 6, 10, 15, \dots \quad a_n = a_{n-1} + n \quad \text{เมื่อ } n > 0, a_0 = 0$$

$$0, 1, 1, 2, 3, 5, 8, \dots \quad f_n = f_{n-1} + f_{n-2} \quad \text{เมื่อ } n > 1, f_0 = 0, f_1 = 1$$

$$a_n = 0, 1, 3, 6, 10, 15, \dots$$

- รู้ว่า $a_n = (0+1+2+\dots+n)$

```
public static int a(int n) {  
    int s = 0;  
    for (int i = 0; i <= n; i++)  
        s += i;  
    return s;  
}
```

แบบที่ 1

- รู้ว่า $a_n = a_{n-1} + n$ เมื่อ $n > 0, a_0 = 0$

```
public static int a(int n) {  
    if (n <= 0)  
        return 0;  
    else  
        return a(n-1) + n;  
}
```

แบบที่ 2

$$f_n = 0, 1, 1, 2, 3, 5, 8, 13, \dots$$

```
public static int fib(int n) {  
    if (n < 2) return n;  
    int[] f = new int[n + 1];  
    f[0] = 0; f[1] = 1;  
    for (int i = 2; i <= n; i++)  
        f[i] = f[i-1] + f[i-2];  
    return f[n];  
}
```

แบบที่ 1

```
public static int fib(int n) {  
    if (n < 2) return n;  
    int fn = 0, fn1 = 0, fn2 = 1;  
    for (int i = 2; i <= n; i++) {  
  
    }  
    return fn;  
}
```

แบบที่ 2

$$f_n = 0, 1, 1, 2, 3, 5, 8, 13, \dots$$

```
public static int fib(int n) {
    if (n < 2) return n;
    int fn2 = fib(n-2);
    int fn1 = fib(n-1);
    return fn1 + fn2;
}
```

แบบที่ 3

```
public static int fib(int n) {
    if (n < 2) return n;
    return fib(n-1) + fib(n-2);
}
```

เมท็อดหาค่า $n!$

```
public static int fac(int n) {
    int fac = 1;
    for (int i = 1; i <= n; i++) {
        fac = fac * i;
    }
    return fac;
}
```

แบบที่ 1

$$n! = n \times (n-1) \times (n-2) \times \dots \times 2 \times 1$$

```
public static int fac(int n) {
    if (n == 0) return 1;
    int f = fac(n-1);
    return n * f;
}
```

แบบที่ 2

กรณีเล็กสุด

ตามนิยามของแฟกทอเรียล

$$n! = \begin{cases} 1 & n = 0 \\ n(n-1)! & n \geq 1 \end{cases}$$

โปรแกรมแบบ recursive

- โปรแกรมที่มีการเรียกตัวเอง
- การทำงานแบ่งเป็นกรณี ๆ
 - กรณีพื้นฐาน ทำเสร็จได้ทันที
 - กรณีอื่น เรียกตัวเอง โดยขนาดของปัญหาต้องเล็กลง
- การเรียกแต่ละครั้ง จะมีการสร้างตัวแปรในเมท็อดชุดใหม่ในระบบ

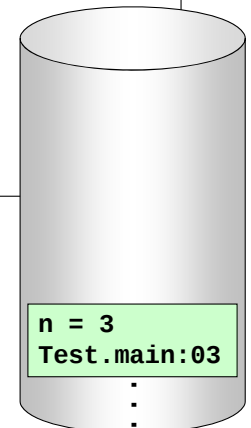
```
public static void main(String[] args) {
    jeng3(10);
}
public static void jeng3(int n) {
    System.out.println(n);
    jeng3(n-1);
}
```

ลองสังเกตว่า เกิดอะไรขึ้น

ตัวแปรระหว่างการเรียกซ้ำ

```
01: public class Test {
02:     public static void main(String[] args) {
03:         System.out.println(fac(3));
04:     }
05:     public static int fac(int n) {
06:         if (n == 0) return 1;
07:         int f = fac(n-1);
08:         return n * f;
09:     }
10: }
```

เมื่อระบบเรียกเมท็อด fac
ระบบจะสร้างตัวแปรของ fac



ตัวแปรระหว่างการเรียกซ้ำ

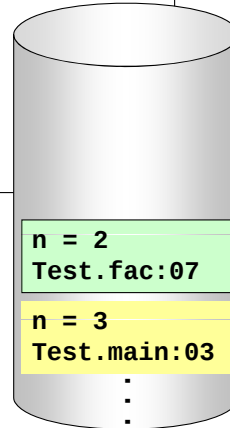
```

01: public class Test {
02:     public static void main(String[] args) {
03:         System.out.println(fac(3));
04:     }
05:     public static int fac(int n) {
06:         if (n == 0) return 1;
07:         int f = fac(n-1);
08:         return n * f;
09:     }
10: }

```



เมื่อเรียกเมทอด fac อีก
ระบบจะสร้างตัวแปรชุดใหม่ของ fac
ชุดใหม่ทับชุดเก่า
(จะนำตัวแปรชุดเก่ากลับมาใช้ใหม่
เมื่อ return)

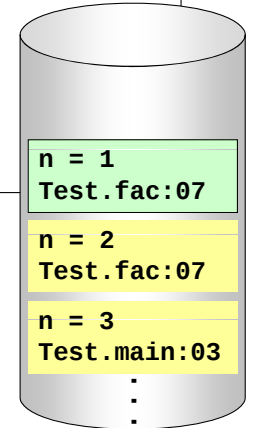


ตัวแปรระหว่างการเรียกซ้ำ

```

01: public class Test {
02:     public static void main(String[] args) {
03:         System.out.println(fac(3));
04:     }
05:     public static int fac(int n) {
06:         if (n == 0) return 1;
07:         int f = fac(n-1);
08:         return n * f;
09:     }
10: }

```

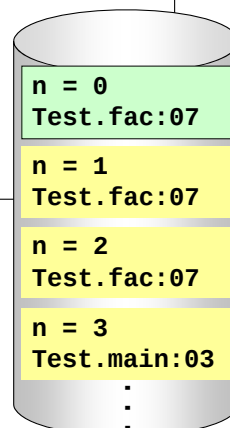


ตัวแปรระหว่างการเรียกซ้ำ

```

01: public class Test {
02:     public static void main(String[] args) {
03:         System.out.println(fac(3));
04:     }
05:     public static int fac(int n) {
06:         if (n == 0) return 1;
07:         int f = fac(n-1);
08:         return n * f;
09:     }
10: }

```

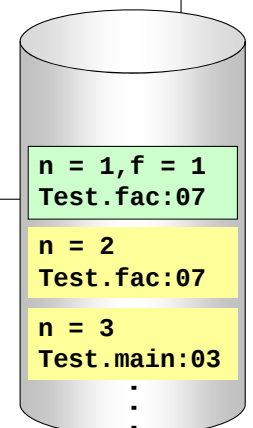


ตัวแปรระหว่างการเรียกซ้ำ

```

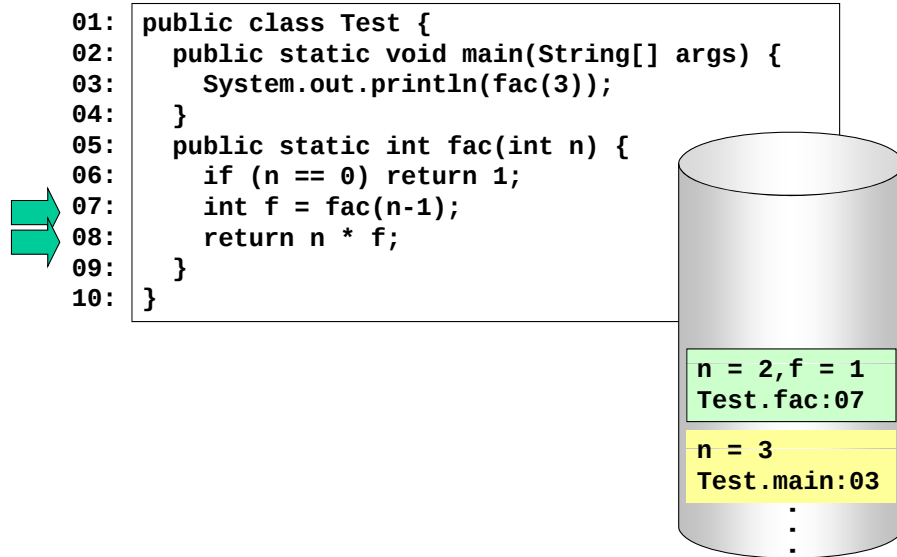
01: public class Test {
02:     public static void main(String[] args) {
03:         System.out.println(fac(3));
04:     }
05:     public static int fac(int n) {
06:         if (n == 0) return 1;
07:         int f = fac(n-1);
08:         return n * f;
09:     }
10: }

```

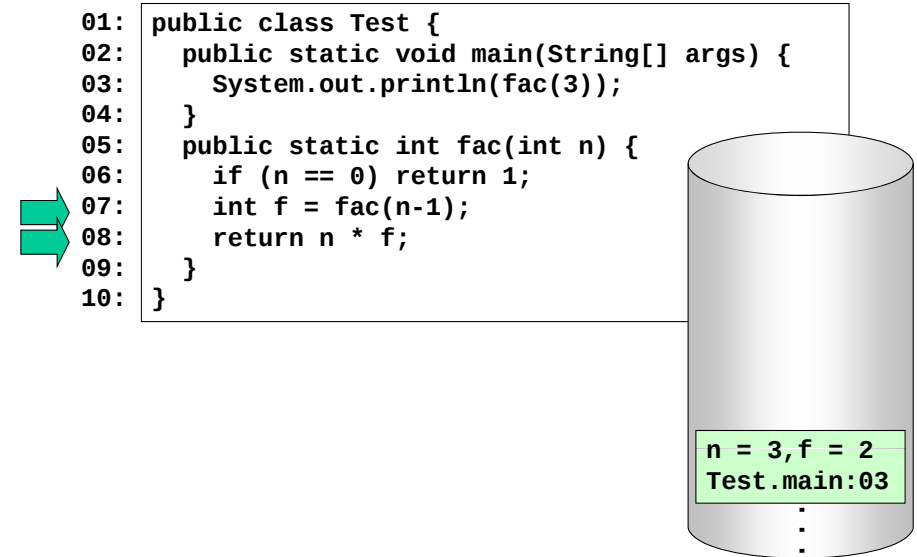


เมื่อ return
ระบบลบตัวแปรของเมทอดออก
และนำตัวแปรชุดล่าสุดมาใช้ต่อ

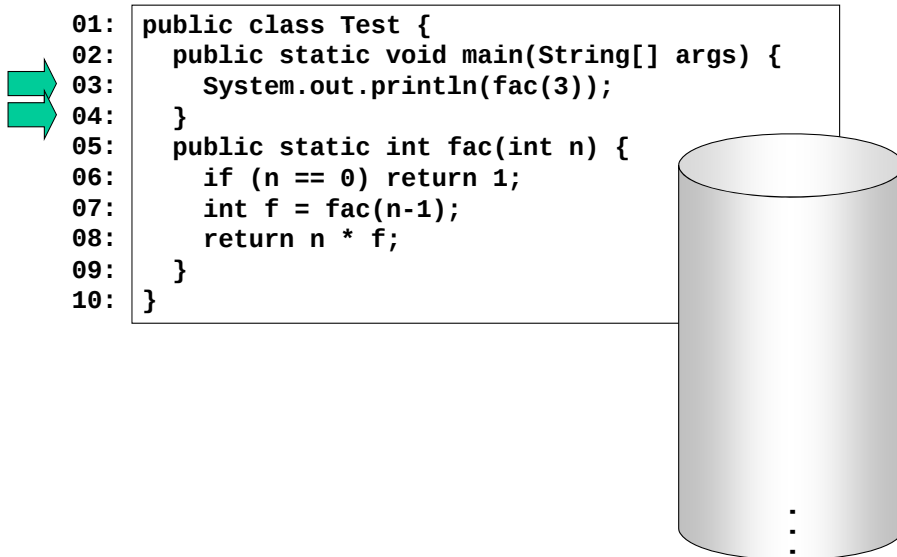
ตัวแปรระหว่างการเรียกซ้ำ



ตัวแปรระหว่างการเรียกซ้ำ



ตัวแปรระหว่างการเรียกซ้ำ



เมื่อดตรวจสอบว่าอาร์เรย์เรียงลำดับหรือไม่

```

public static boolean isSorted(int[] d) {
    for (int i = 1; i < d.length; i++) {
        if (d[i - 1] > d[i]) return false;
    }
    return true;
}

```

แบบที่ 1

```

public static boolean isSorted(int[] d) {
    return isSorted(d, d.length);
}
private static boolean isSorted(int[] d, int n) {
    if (n <= 1) return true;
    if (d[n - 2] > d[n - 1]) return false;
    return isSorted(d, n - 1);
}

```

แบบที่ 2

การคำนวณ $a^k \bmod m$

- $a^k \bmod m$ เป็นการคำนวณที่ใช้บ่อยในการเข้ารหัสลับ
- ตัวอย่างที่ 1 : $2^{20} \% 31 = ?$
 - คำนวณ 2^{20} ได้ 1048576 จากนั้น $\% 31$ ได้ 1
- ตัวอย่างที่ 2 : $2^{101} \% 31 = ?$
 - คำนวณ 2^{101} ได้ 2535301200456458802993406410752 จากนั้น $\% 31$ ได้ 2
 - ทำอีกแบบ :

$$a^k \% m = \begin{cases} 1 & k = 0 \\ (a^{\lfloor k/2 \rfloor} \% m)^2 \% m & k \text{ is even} \\ a(a^{\lfloor k/2 \rfloor} \% m)^2 \% m & k \text{ is odd} \end{cases}$$

$$\begin{aligned} 2^{60} \bmod 10 &= 4^2 \bmod 10 = 6 \\ 2^{30} \bmod 10 &= 8^2 \bmod 10 = 4 \\ 2^{15} \bmod 10 &= 2 \times 8^2 \bmod 10 = 8 \\ 2^7 \bmod 10 &= 2 \times 8^2 \bmod 10 = 8 \\ 2^3 \bmod 10 &= 2 \times 2^2 \bmod 10 = 8 \\ 2^1 \bmod 10 &= 2 \times 1^2 \bmod 10 = 2 \\ 2^0 \bmod 10 &= 1 \end{aligned}$$

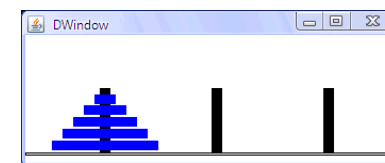
การคำนวณ $a^k \bmod m$

$$a^k \bmod m = \begin{cases} 1 & k = 0 \\ (a^{\lfloor k/2 \rfloor} \bmod m)^2 \bmod m & k \text{ is even} \\ a(a^{\lfloor k/2 \rfloor} \bmod m)^2 \bmod m & k \text{ is odd} \end{cases}$$

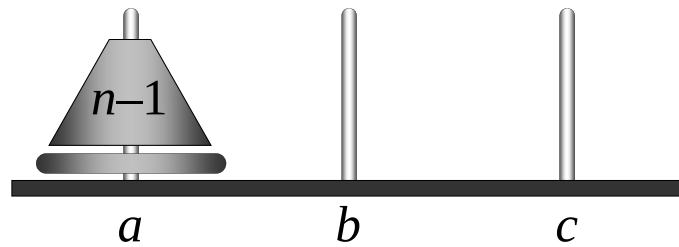
```
public static int powerMod(int a, int k, int m) {
    if (k == 0) return 1;
    int p = powerMod(a, k / 2, m);
    p = (p * p) % m;
    if ((k % 2) == 1) p = (a * p) % m;    // p is odd
    return p;
}
```

ปัญหา Tower of Hanoi

- มีเสา 3 ต้น
- ใส่จานกลมมีรูตรงกลางขนาดต่างกัน n ใบ ในเสาต้นซ้ายกองเรียงตามขนาด ใบใหญ่สุดอยู่ล่างสุด
- ต้องการ
 - ย้ายจานทุกใบจากเสาซ้ายไปใส่ที่เสาขวา
 - ย้ายได้ครั้งละใบ
 - อนุญาตให้ใบเล็กทับใบใหญ่ได้เท่านั้น
- ย้ายอย่างไร ?



hanoi(n, a, b, c)

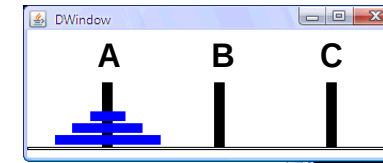


hanoi(n - 1, a, c, b)

move n : a → c

hanoi(n - 1, b, a, c)

โปรแกรมแก้ปัญหา Tower of Hanoi



```
move 1 : A -> C
move 2 : A -> B
move 1 : C -> B
move 3 : A -> C
move 1 : B -> A
move 2 : B -> C
move 1 : A -> C
```

```
public class HanoiTower {
    public static void main(String[] args) {
        hanoi(3, "A", "B", "C");
    }
    private static void hanoi(int n,
                               String from, String tmp, String to) {
        if (n == 0) return;
        hanoi(n - 1, from, to, tmp);
        System.out.println("move " + n + " : " + from +
                           " -> " + to);
        hanoi(n - 1, tmp, from, to);
    }
}
```