

2110101 : การทำโปรแกรมคอมพิวเตอร์

ระบบคอมพิวเตอร์เบื้องต้น

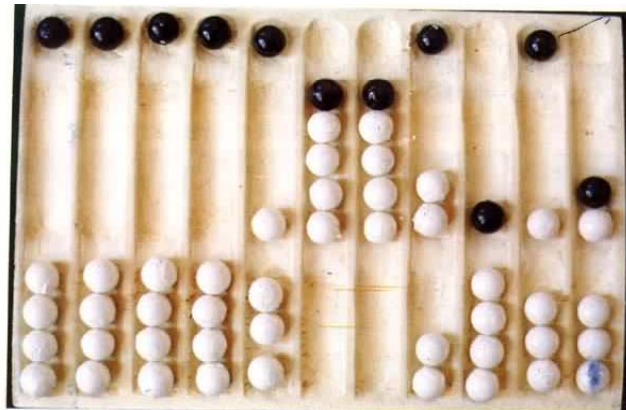
ภาควิชาวิศวกรรมคอมพิวเตอร์
จุฬาลงกรณ์มหาวิทยาลัย

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

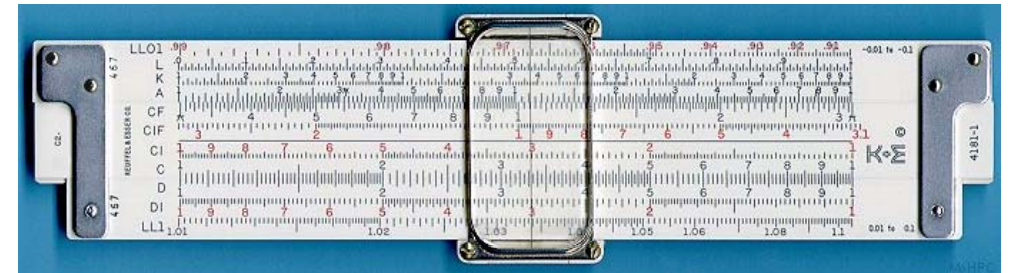
2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (25/05/52)

2

300 B.C. : Babylonians



1632 : Slide Rule



1642 : Blaise Pascal (Pascaline)

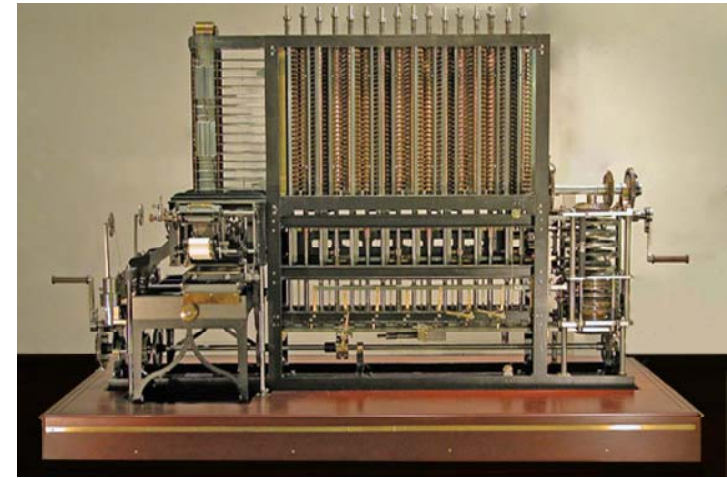


Blaise Pascal
1623-1662



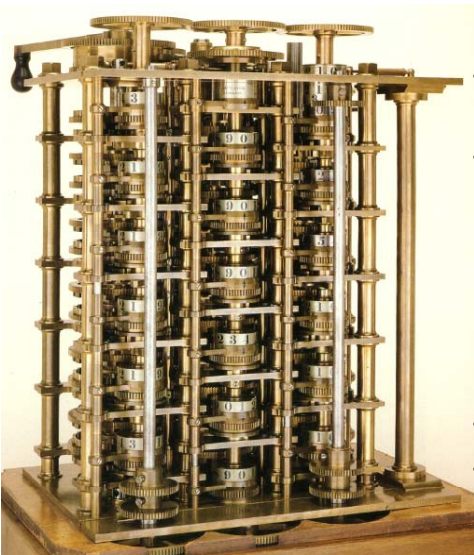
Mechanical calculator
Add and subtract

1822 : Charles Babbage (Difference Engine)



Compute tables of numbers, such as logarithm tables

1833 : Charles Babbage (Analytical Engine)



- The first autonomic 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

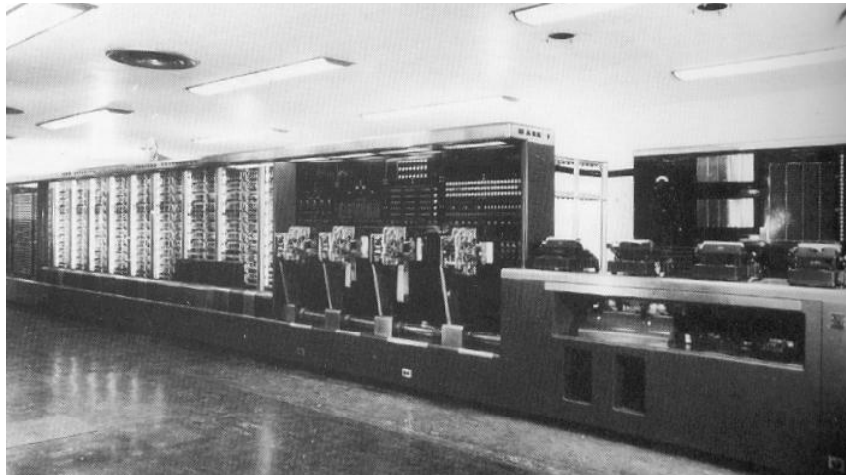
1890 : Herman Hollerith (Hollerith desk)



เครื่องอ่านบัตรเจาะรู
เพื่อนับข้อมูลทางสถิติ
สำมะโนครัวประชากร

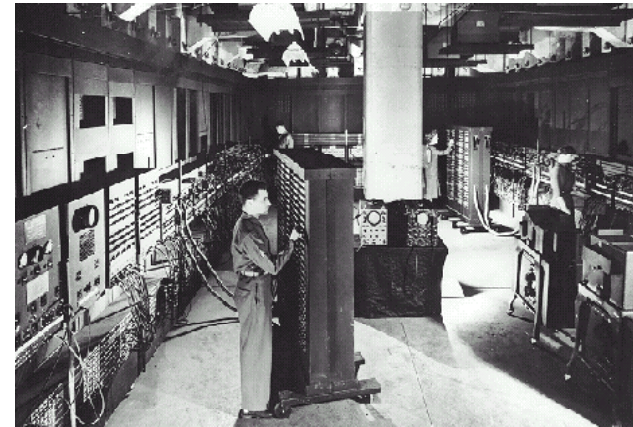


1940 : Harvard Mark I



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

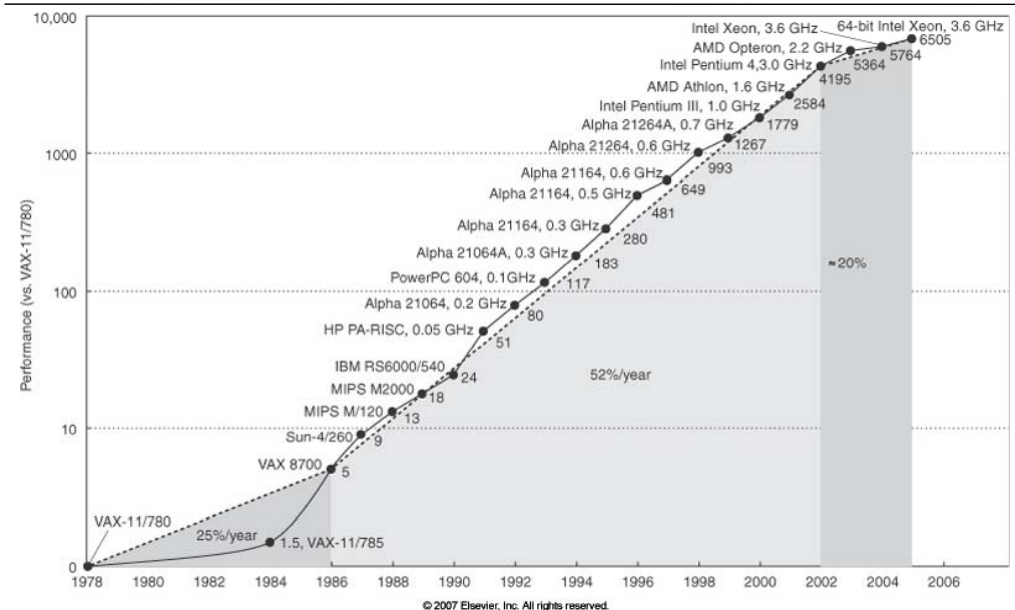
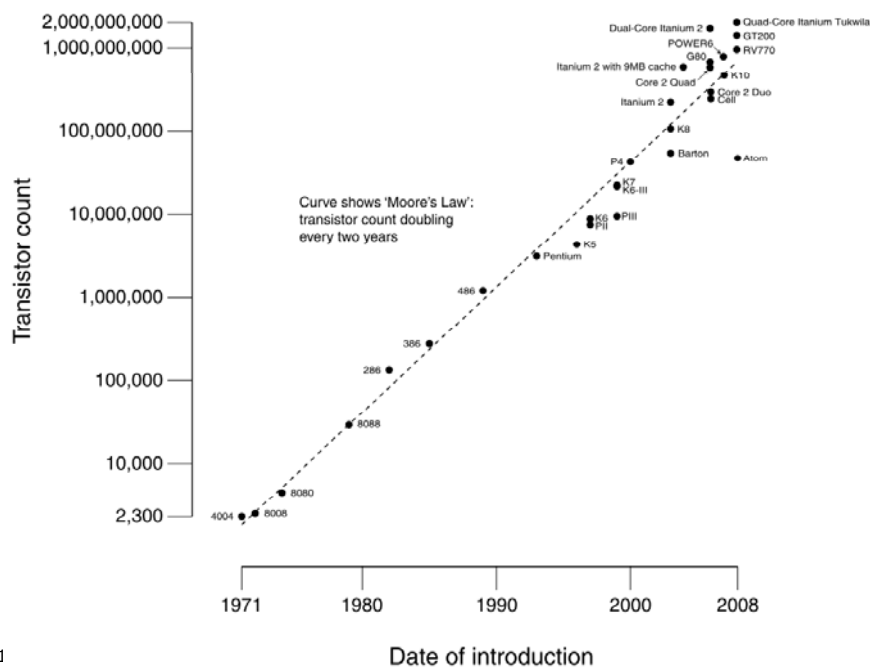


Apple Computers



70 กว่าปีที่ผ่านมา

Game Consoles



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



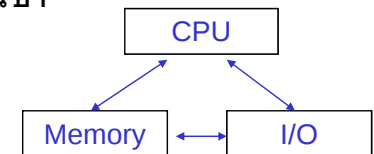
องค์ประกอบของคอมพิวเตอร์

- hardware เพื่อประมวลผล จัดเก็บ และสื่อสารข้อมูล
- software เพื่อสั่งให้ hardware ทำงานตามขั้นตอนที่กำหนดไว้



Hardware

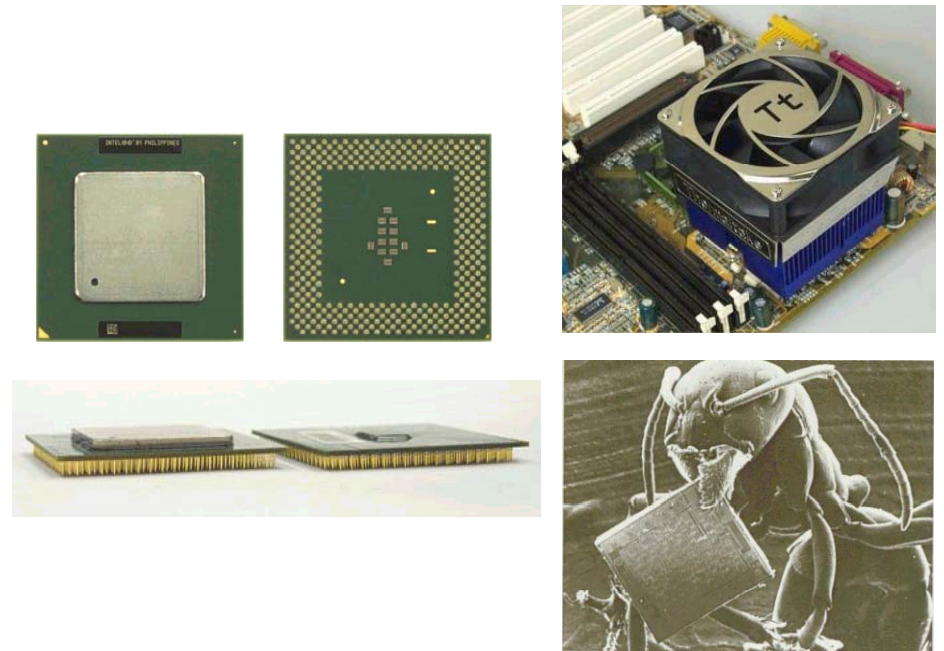
- Central Processing Unit (CPU)
 - เพื่อประมวลผลทางคณิตศาสตร์, ตรรกะ, การทดสอบจริงเท็จ, การทำงานซ้ำ
- Memory
 - เพื่อจัดเก็บและค้นคืนข้อมูล
- I/O
 - เพื่อรับส่งสื่อสารข้อมูล (จอภาพ แป้นพิมพ์ เมาส์ โมเด็ม เครื่องพิมพ์ กล้อง ...)



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

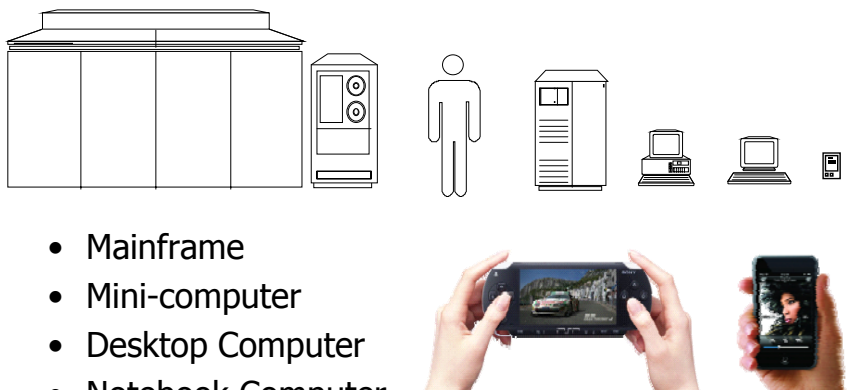
bps – bits per second

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



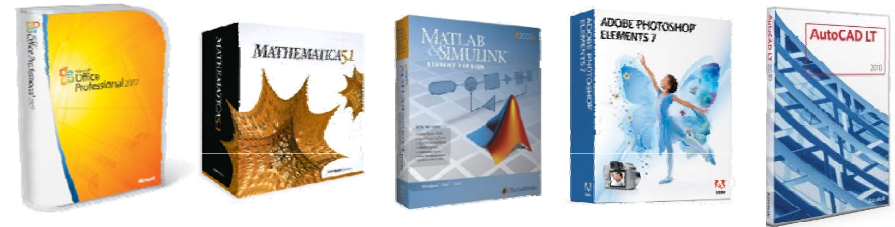
20,000 บาท

- 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

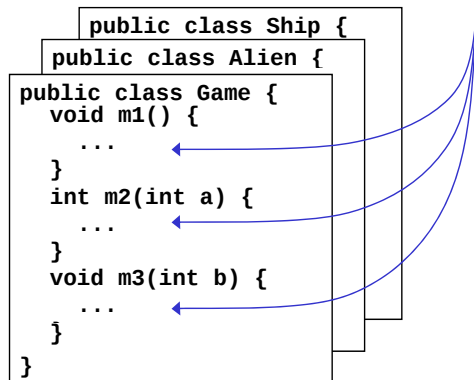
2110101 วิศวกรรมคอมพิวเตอร์ จุฬาฯ (25/05/52)

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

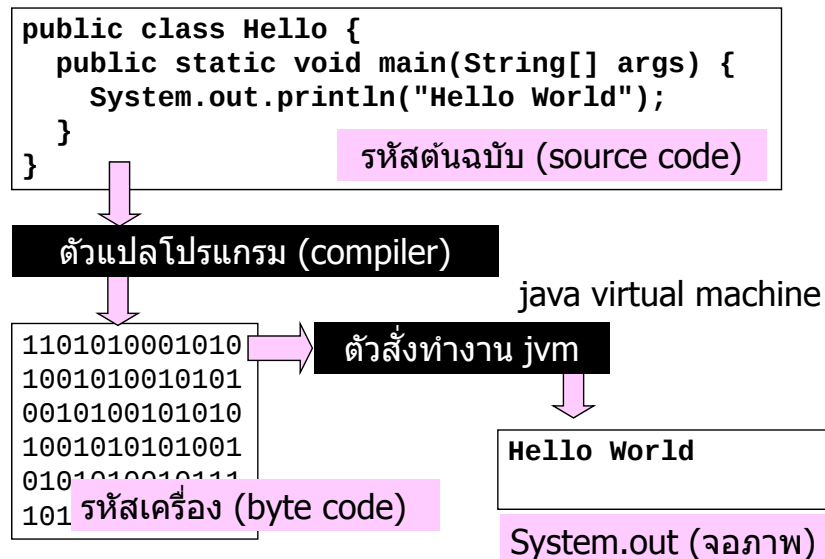
```
public class Hello {  
    public static void main(String[] args) {  
        System.out.println("Hello World");  
        System.out.print("Hello ");  
        System.out.println("Java");  
        System.out.print( "4+5 = " );  
        System.out.println( 4+5 );  
    }  
}
```

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("Java");  
        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

สวายเป็นไปอาจผิดกฎ

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

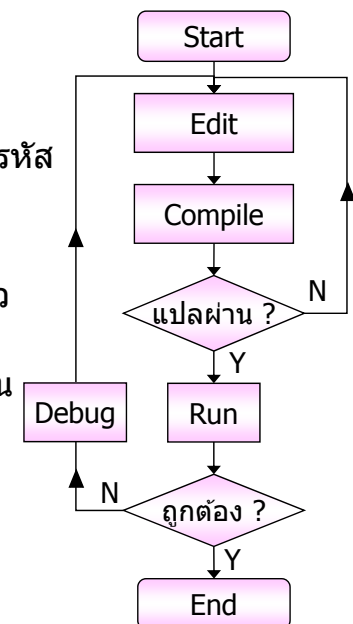
เขียนได้หรือไม่

```
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