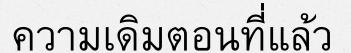
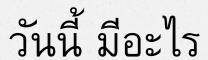
สัปดาห์ที่ 2 – ตัวดำเนินการ และประโยคเงื่อนไข

การเตรียมความพร้อมสู่ สอวน. คอมพิวเตอร์ ค่าย 1 ระยะที่ 1 ห้องกุหลาบเพชร โรงเรียนสวนกุหลาบวิทยาลัย ศิระ ทรงพลโรจนกุล



- > รู้จัก หน้าตาของภาษาซี่
- > รู้จัก ตัวแปร
- > ใช้ printf scanf ได้



- ตัวดำเนินการ
- ประโยคเงื่อนไข

ตัวดำเนินการ

Assignment (=)

เอาค่าทาง ขวา ไปใส่ไว้ทางซ้าย

```
a = 10 // correct
10 = a // error
```

$$a = b = c = 5$$

ตัวอย่าง

```
int a,b;

a = 10; a = 10 b = ?

b = 5; a = 10 b = 5

a = b; a = 5 b = 5

b = 8; a = 5 b = 8

printf("b=%d a=%d\n",b,a);

b = 8 a = 5
```

ตัวดำเนินการทางคณิตศาสตร์



ตัวดำเนินการทางคณิตศาสตร์

เครื่องหมาย	ชื่อ	ตัวอย่าง
+	บวก	A = 7 + 3; // $A = 10$
_	ลบ	A = 7 - 3; // $A = 4$
*	คูณ	A = 7 * 3; // A = 21
/	หาร	A = 7 / 3; // A = 2
ે	หารเอาเศษ (mod)	A = 7 % 3; // A = 1

สังเกตว่า **7/3** ไม่ใช่ **2.333...**



การดำเนินการทางคณิตศาสตร์

การดำเนินการทางคณิตศาสตร์ เป็นการกระทำแบบ binary และทำ คูณหาร ก่อนบวกลบ

$$2+3*2+3 \rightarrow 2+(3*2)+3 \rightarrow 2+6+3 \rightarrow (2+6)+3 \rightarrow 8+3 \rightarrow 11$$

การดำเนินการ จะได้ตัวแปรที่ใหญ่กว่า

int*int → int int*double → double





ตัวดำเนินการทางคณิตศาสตร์บนตัวเอง

เครื่องหมาย	ชื่อ	ตัวอย่าง
+=	บวก	A = 7; A += 3; // A = 10
-=	ลบ	A = 7; A -= 3; // A = 4
*=	คูณ	A = 7; A *= 3; // A = 21
/=	หาร	A = 7; A /= 3; // A = 2
%=	หารเอาเศษ (mod)	A = 7; A %= 3; // A = 1

ตัวดำเนินการนี้เป็นลักษณะ unary



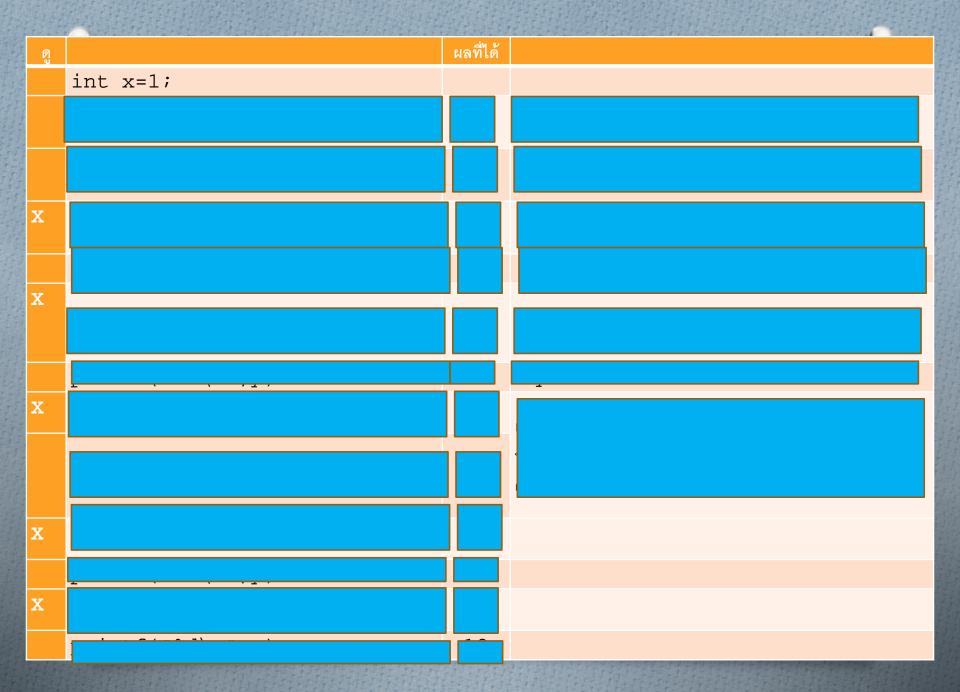
Increase and decrease

เครื่องหมาย	ชื่อ	ตัวอย่าง
++	เพิ่มค่าขึ้นหนึ่ง	A = 3; A++; // A = 4 ++A; // A = 5
	ลดค่าลงหนึ่ง	A = 3; A; // A = 2 A; // A = 1



ความแตกต่าง ระหว่าง a++ กับ ++a

- a++ -> เอาค่า a ไปใช้ก่อน แล้วค่อยเพิ่มค่าขึ้น 1
- ++a → บวกค่าเพิ่มขึ้น 1 ก่อน แล้วค่อยเอาค่าไปใช้





int x=3;		
printf("%d %d\n",x++,x++);	4 3	สังเกตว่า comma จะเป็นตัว แบ่ง คำสั่ง ใน printf โดยทำ จากคำสั่งหลังมาหน้า
<pre>printf("%d\n",x);</pre>	5	
int x=3;		
printf("%d %d\n",++x,++x);	5 4	
<pre>printf("%d\n",x);</pre>	5	

ตัวดำเนินการบิต

คอมพิวเตอร์ เก็บข้อมูลต่างๆ เป็นเลขฐานสอง เช่น

int a = 127 ข้อมูลที่เก็บก็คือ

a: 00000000 00000000 00000000 11111111

ตัวดำเนินการบิต จะดำเนินการบนเลขฐานสองนี้

ตัวดำเนินการบิต

เครื่องหมาย	ชื่อ	ตัวอย่าง
~	กลับบิต	char A=0xF1;// A:11110001 B = ~A; // B:00001110
&	And bit	A = 5 & 3; // $A = 1$
	Or bit	A = 5 3; // A = 7
^	Xor	$A = 5 ^3; // A = 6$
>>	เลื่อนบิตไปทางขวา	A = 5 >> 1; // $A = 2$
<<	เลื่อนบิตไปทางซ้าย	$A = 5 \ll 1;$ // $A = 10$

[~] กลับบิต

กลับบิตทุกตัว เรียกอีกอย่างหนึ่งว่า one's complement

```
ตัวอย่าง
```

```
char a = 0xAF; // 1010 1111
char b = \sima; // 0101 0000
printf("%x\n",b); // 50
```



[&] and bit

a	b	a & b
0	0	0
0	1	0
1	0	0
1	1	1

และ ต้องเป็น จริงทั้งคู่ ถึงเป็นจริง

ตัวอย่าง

A = 0101 1011

B = 1100 1101

A&B= 0100 1001



[|] or bit

a	b	a b
0	0	0
0	1	1
1	0	1
1	1	1

หรือ มีอันไหนจริง ก็จริงแล้ว

ตัวอย่าง

$$A = 0101 1011$$

$$B = 1100 1101$$



[^] xor

a	b	a ^ b
0	0	0
0	1	1
1	0	1
1	1	0

มีอันจริง ได้แค่อันเดียว ถึงเป็นจริง

ตัวอย่าง

A = 0101 1011

 $B = 1100 \ 1101$

A^B= 1001 0110

Shift bit

เลื่อน บิตไปตามทิศทางของลูกศร เป็นจำนวนกี่สเตป

$$A = 27 \rightarrow 0001 \ 1011$$

$$A << 1 \rightarrow 0011 \ 0110 \rightarrow 54$$

$$A << 2 \rightarrow 0110 \ 1100 \rightarrow 108$$

$$A >> 1 \rightarrow 0000 \ 1101 \rightarrow 13$$

$$A >> 2 \rightarrow 0000\ 0110 \rightarrow 6$$

สังเกตว่า เลื่อนบิตไปทาง ซ้าย = คูณ 2 / ขวา = หาร 2

รู้ไว้เหอะ

สามารถเปลี่ยน เลขฐาน ระหว่าง ฐาน 2/8/16 ได้ง่ายๆ โดยการจัด กลุ่มบิต

Base 8 จัดกลุ่มบิต 3 ตัว

Base 16 จัดกลุ่มบิต 4 ตัว

ตัวอย่าง แปลง 01101011 (= 107) เป็นเลขฐาน 8 และ 16

 $01101011 \rightarrow 01/101/011 \rightarrow 153 \text{ (base 8)}$

 $01101011 \rightarrow 0110/1011 \rightarrow 6B \text{ (base 16)}$

ปัญหาที่ 1:16 2 8

แปลง AF09 (base16) เป็นเลขฐาน 8

คำตอบ

AF09 แปลงเป็นเลขฐานสองได้

1010 1111 0000 1001

จัดกลุ่มบิตสามตัวได้เป็น

1 010 111 100 001 001

แปลงเป็นเลขฐานแปดได้

127411 (base8)

ตัวดำเนินการบิต บนตัวเอง

เครื่องหมาย	ชื่อ	ตัวอย่าง	
&=	And bit	A = 5; A &= 3;	// A = 1
=	Or bit	A = 5; A = 3;	// A = 7
^=	Xor	A = 5; A ^= 3;	// A = 6
>>=	เลื่อนบิตไปทางขวา	A = 5; A >>= 1;	// A = 2
<<=	เลื่อนบิตไปทางซ้าย	A = 5; A <<= 1;	// A = 10

ตัวดำเนินการเปรียบเทียบ

ตัวดำเนินการเปรียบเทียบ จะบอกว่า การเปรียบนี้ ถูก (true) หรือ ผิด (false) ตัวอย่างเช่น

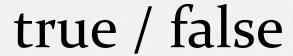
 $2 < 3 \rightarrow \text{true}$

 $2 > 3 \rightarrow false$



ตัวดำเนินการเปรียบเทียบ

เครื่องหมาย	ชื่อ	ตัวอย่าง
>	มากกว่า	4 > 3 → true
>=	มากกว่า หรือเท่ากับ	3 >= 3 → true
<	น้อยกว่า	6 < 5 → false
<=	น้อยกว่า หรือเท่ากับ	4 <= 2 → false
==	เท่ากับ	2 == 3 → false
! =	ไม่เท่ากับ	2 != 3 → true



ในภาษาซี

true มีค่าเป็น 1

false มีค่าเป็น 0

และ

0 คือ false

อะไรที่ไม่ใช่ 0 คือ true



true

false

true

ERROR

true

false

1 < 2 < 3 (1 < 2) < 3 (true) < 3 1 < 3 true

$$3 > 2 > 1$$

(3 > 2) > 1
(true) > 1

1 > 1 false

ตัวดำเนินการทางตรรกะ



เครื่องหมาย	ชื่อ	ตัวอย่าง
!	Not	$!(2 > 3) \rightarrow true$
&&	And	$3 >= 3 \&\& 0 \rightarrow false$
	Or	6 < 5 7 → true

a	b	!a	a && b	a b
0	0	1	0	0
0	1	1	0	1
1	0	0	0	1
1	1	0	1	1

 $0 \rightarrow \text{false}$ $1 \rightarrow \text{true}$

ตัวอย่าง

int
$$a=5,b=-2;$$

3 < 5 && a > 3 true

b>=a || a+b > a*b true

a && b true

a &&!b false

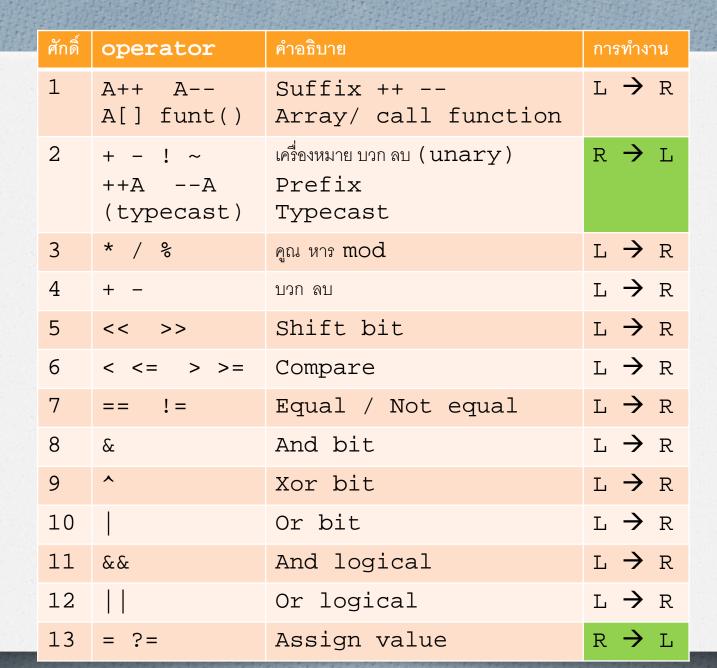
a > b && b > -10 true

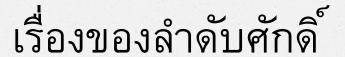
!!a true

true && true | | false true

true | | true && false true

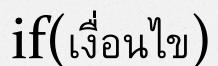
คิด not \rightarrow and \rightarrow or 34





คิดอะไรไม่ออก ก็ใส่วงเล็บครอบไป จะได้ทำงานตามที่เราคิด อย่างถูกต้อง

ประโยคเงื่อนไข



if (เงื่อนไข) คำสั่งที่ทำเมื่อเงื่อนไขเป็นจริง

ตัวอย่าง

if(x>10) printf("x more than $10\n$ ");

```
if(x<2)
    printf("A");
    printf("B");
printf("C");

x=1 → ?
x=2 → ?</pre>
```

if(เงื่อนไข) กับ หลายคำสั่ง

```
if (เงื่อนไข)
{
    คำสั่งที่ทำเมื่อเงื่อนไขเป็นจริง
    •
    •
}
```

```
if(x<2)
{
    printf("A");
    printf("B");
}
printf("C");

x=1 → ABC
x=2 → C</pre>
```

```
อย่าลืม
ใช้ วงเล็บปิกกา คลุม ทั้งก้อน
ถ้ามีหลายคำสั่ง
(มีคำสั่งเดียวก็ใช้ปิกกาคลุมได้)
```





ตัวอย่าง 1: เลขคู่ เลขคื่

input : จำนวนเต็ม N

output : บอกว่า N เป็นเลขคู่ หรือ เลขคี่

```
if(N%2==0)
   printf("%d is even\n",N);
else
   printf("%d is odd\n",N);
```



ตัวอย่างของ if อีกนิด

แต่ละข้อ อยากรู้ว่า พิมพ์ X หรือเปล่า

```
int a=5, b=6;
if(a > 5) printf("x\n");
if(true) printf("x\n");
if(0) printf("x\n");
                        X
if(1) printf("x\n");
                        X
if(-1) printf("x\n");
if(a + b) printf("x \n");
                            X
if(a ^a) printf("x\n");
if(a & b - 4) printf("x\n");
if(3 > 2 > 1) printf("x\n");
if (3 > 2 \&\& 2 > 1) printf("x\n");
                                    X
if(a>0 && b>0 && a<b) printf("x\n");
                                    X
if(a==b-1) printf("x\n");
                                    X
if(a=b-1) printf("x\n");
                                    X
if(a=b-6) printf("x\n");
```

skgrader: 366

รับปี ค.ศ. เข้ามา ถ้าเป็นปี อธิกสุรทิน (มี 366 วัน) ให้พิมพ์ **TAK** มิเช่นนั้น พิมพ์ **NIE**



if-else if

```
if(เงื่อนไข1)
    คำสั่งที่ทำเมื่อเงื่อนไข1 เป็นจริง
else if(เงื่อนไข2)
    คำสั่งที่ทำเมื่อเงื่อนไข1 ไม่จริง แต่ เงื่อนไข2 จริง
else if(เงื่อนไข3)
    คำสั่งที่ทำเมื่อเงื่อนไขที่ผ่านมาไม่จริง แต่เงื่อนไข3 จริง
                   จะไม่มี else ตัวสุดท้ายก็ได้
else
    คำสั่งที่ทำเมื่อไม่เข้าเงื่อนไขใดเลย
```



ตัวอย่าง 2: มากกว่า น้อยกว่า หรือว่าพอดีเป็ะ

รับจำนวนเต็มสองตัว a b

ให้ตอบว่า a > b หรือ a < b หรือ a = b

```
if(a > b)
    printf("%d > %d\n",a,b);
else if(a < b)
    printf("%d < %d\n",a,b);
else if(a == b)
    printf("%d = %d\n",a,b);</pre>
```



หลากหลาบวิธีการ

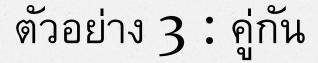
```
if(a > b)
    printf("%d > %d\n",a,b);
else if(a < b)
    printf("%d < %d\n",a,b);
else if(a == b)
    printf("%d = %d\n",a,b);</pre>
```

```
if(a > b)
    printf("%d > %d\n",a,b);
else if(a < b)
    printf("%d < %d\n",a,b);
else
    printf("%d = %d\n",a,b);</pre>
```



if ซ้อนๆ

```
if(เงื่อนไข1)
  if(เงื่อนไข2)
     if(เงื่อนไข3)
        คำสั่งที่ทำเมื่อเงื่อนไข1,2,3 จริง
     else
        คำสั่งที่ทำเมื่อเงื่อนไข1,2 จริง 3 เท็จ
  else
     คำสั่งที่ทำเมื่อเงื่อนไข1 จริง 2 เท็จ (ไม่สนใจ เงื่อนไข3)
else
 คำสั่งที่ทำเมื่อเงื่อนไข1 เท็จ (ที่เหลือไม่สนใจ)
```



รับจำนวนเต็มสองตัว a b ถ้า a เป็นคี่ พิมพ์ alone ถ้า a เป็นคู่ ให้ดูที่ b ถ้า b เป็นคี่ ให้พิมพ์ alone มิเช่นนั้น พิมพ์ yeh

```
if(a%2==1)
   printf("alone");
else if(a%2==0 && b%2==1)
   printf("alone");
else if(a%2==0 && b%2==0)
   printf("yeh");
```

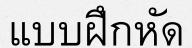


```
if(a%2==1)
    printf("alone");
else
{
    if(b%2==1)
        printf("alone");
    else
        printf("yeh");
}
```

ถ้าอ่านขาด จะได้

```
if(a%2==0 && b%2==0)
   printf("yeh");
else
   printf("alone");
```

?? คำถาม ??



> skgrader : น้อยสุดในสาม

> skgrader : สมการเส้นตรง

> skgrader : กำลังของสอง

จบสัปดาห์ที่ 2

สวัสดี