

ตัวดำเนินการ (Operator)

ตัวดำเนินการ (Operator)

- ▶ ตัวดำเนินการทางคณิตศาสตร์ (Arithmetic Operators)
- ▶ ตัวดำเนินการเปรียบเทียบ (Comparison Operators)
- ▶ ตัวดำเนินการมอบหมายค่า (Assignment Operators)
- ▶ ตัวดำเนินการทางตรรกะ (Logical Operators)
- ▶ ตัวดำเนินการสำหรับข้อมูลชนิด BitWise
- ▶ ตัวดำเนินการเกี่ยวกับสมาชิก (Membership Operators)
- ▶ ตัวดำเนินการเกี่ยวกับบิตเล่น (Identity Operators)
- ▶ ตัวดำเนินการกับข้อความ (String Operators)
- ▶ ตัวดำเนินการกับรายการ (List Operators)

ตัวดำเนินการทางคณิตศาสตร์

| ตัวดำเนินการ | ตัวอย่าง | คำอธิบาย |
|--------------|----------------|--------------|
| + | $2 + 3 = 5$ | บวก |
| - | $5 - 2 = 3$ | ลบ |
| * | $3 * 4 = 12$ | คูณ |
| / | $10 / 2 = 5.0$ | หาร |
| // | $11 // 3 = 3$ | หารเอาส่วนลง |
| % | $11 \% 3 = 2$ | หารเอาเศษ |
| ** | $2 ** 3 = 8$ | ยกกำลัง |

ตัวดำเนินการเปรียบเทียบ

| ตัวดำเนินการ | ตัวอย่าง | คำอธิบาย |
|--------------------|---------------------------------|---------------------|
| <code>==</code> | <code>5 == 5</code> เป็นจริง | เท่ากับ |
| <code>!=</code> | <code>5 != 3</code> เป็นจริง | ไม่เท่ากับ |
| <code>></code> | <code>7 > 5</code> เป็นจริง | มากกว่า |
| <code><</code> | <code>3 < 7</code> เป็นจริง | น้อยกว่า |
| <code>>=</code> | <code>7 >= 7</code> เป็นจริง | มากกว่าหรือเท่ากับ |
| <code><=</code> | <code>3 <= 5</code> เป็นจริง | น้อยกว่าหรือเท่ากับ |

ตัวดำเนินการทางตรรกะ

| ตัวดำเนินการ | ตัวอย่าง | คำอธิบาย |
|------------------|---|----------|
| <code>and</code> | <code>(5 > 3) and (3 > 1)</code> เป็นจริง | และ |
| <code>or</code> | <code>(5 < 3) or (3 < 1)</code> เป็นจริง | หรือ |
| <code>not</code> | <code>not(5 > 3)</code> เป็นเท็จ | ปฏิเสธ |

ตัวดำเนินการสำหรับข้อมูลชนิด BitWise

| ตัวดำเนินการ | ตัวอย่าง | คำอธิบาย |
|-----------------------|---|----------------------|
| <code>&</code> | <code>0b1010 & 0b1100 = 0b1000</code> | AND บิต |
| <code>`</code> | <code>`</code> | <code>`0b1010</code> |
| <code>^</code> | <code>0b1010 ^ 0b1100 = 0b0110</code> | XOR บิต |
| <code>~</code> | <code>~0b1010 = -0b1011</code> | NOT บิต |
| <code><<</code> | <code>0b1010 << 2 = 0b101000</code> | เลื่อนบิตไปทางซ้าย |
| <code>>></code> | <code>0b1010 >> 2 = 0b0010</code> | เลื่อนบิตไปทางขวา |

ตัวดำเนินการกับข้อความ(String)

| ตัวดำเนินการ | ตัวอย่าง | คำอธิบาย |
|----------------|---|------------|
| <code>+</code> | <code>'Hello' + 'World' = 'HelloWorld'</code> | รวมข้อความ |
| <code>*</code> | <code>'Hi' * 3 = 'HiHiHi'</code> | ทวนข้อความ |

ตัวดำเนินการกับรายการ(List)

| ตัวดำเนินการ | ตัวอย่าง | คำอธิบาย |
|--------------|-----------------------------------|-----------|
| + | $[1, 2] + [3, 4] = [1, 2, 3, 4]$ | รวมรายการ |
| * | $[1, 2] * 3 = [1, 2, 1, 2, 1, 2]$ | ทวนรายการ |

ตัวดำเนินการมอบหมายค่า

| ตัวดำเนินการ | ตัวอย่าง | คำอธิบาย |
|------------------|---|--------------------------|
| <code>=</code> | <code>x = 5</code> | กำหนดค่า |
| <code>+=</code> | <code>x = 2; x += 3</code> ได้ <code>x = 5</code> | บวกแล้วกำหนดค่า |
| <code>-=</code> | <code>x = 5; x -= 3</code> ได้ <code>x = 2</code> | ลบแล้วกำหนดค่า |
| <code>*=</code> | <code>x = 2; x *= 3</code> ได้ <code>x = 6</code> | คูณแล้วกำหนดค่า |
| <code>/=</code> | <code>x = 6; x /= 3</code> ได้ <code>x = 2.0</code> | หารแล้วกำหนดค่า |
| <code>%=</code> | <code>x = 7; x %= 3</code> ได้ <code>x = 1</code> | หารเอาเศษแล้วกำหนดค่า |
| <code>**=</code> | <code>x = 2; x **= 3</code> ได้ <code>x = 8</code> | ยกกำลังแล้วกำหนดค่า |
| <code>//=</code> | <code>x = 7; x //= 3</code> ได้ <code>x = 2</code> | หารเอาส่วนลงแล้วกำหนดค่า |

ตัวดำเนินการเกี่ยวกับสมาชิก

| ตัวดำเนินการ | ตัวอย่าง | คำอธิบาย |
|---------------------|--|---------------------------|
| <code>in</code> | <code>'H' in 'Hello'</code> เป็นจริง | ตรวจสอบหาสมาชิกในวัตถุ |
| <code>not in</code> | <code>'x' not in 'Hello'</code> เป็นจริง | ตรวจสอบไม่พบสมาชิกในวัตถุ |

ลำดับความสำคัญของ ตัวดำเนินการ

| ลำดับความสำคัญ | ตัวดำเนินการ | คำอธิบาย |
|----------------|---|---------------------------------------|
| 1 | <code>**</code> | ยกกำลัง |
| 2 | <code>+x</code> , <code>-x</code> , <code>~x</code> | การกระทำครั้งเดียว (unary operation) |
| 3 | <code>*</code> , <code>/</code> , <code>//</code> , <code>%</code> | คูณ, หาร, หารเอาส่วนลง, หารเอาเศษ |
| 4 | <code>+</code> , <code>-</code> | บวก, ลบ |
| 5 | <code><<</code> , <code>>></code> | เลื่อนบิตไปทางซ้าย, เลื่อนบิตไปทางขวา |
| 6 | <code>&</code> | AND บิต |
| 7 | <code>^</code> | XOR บิต |
| 8 | <code>~</code> | ~ |
| 9 | <code><</code> , <code><=</code> , <code>></code> , <code>>=</code> , <code>!=</code> , <code>==</code> | เปรียบเทียบ |
| 10 | <code>not x</code> | ปฏิเสธ |
| 11 | <code>and</code> | และ |
| 12 | <code>or</code> | หรือ |

หมายเหตุ

- ▶ ตัวดำเนินการที่มีลำดับความสำคัญเดียวกันจะถูกประเมินจากซ้ายไปขวา
- ▶ การใช้วงเล็บ () สามารถเปลี่ยนลำดับความสำคัญของตัวดำเนินการได้ เช่น $(2 + 3) * 4$ จะคำนวณเป็น 20

นิพจน์ (expression)

ในภาษาโปรแกรม หมายถึง กลุ่มของค่าคงที่ (Literals) ตัวแปร (Variables) และตัวดำเนินการ (Operators) ที่ถูกนำมารวมกัน เพื่อคำนวณหรือประมวลผลค่าใหม่ออกมา

องค์ประกอบของนิพจน์

- ▶ ค่าคงที่ (Literals) เช่น ตัวเลข 42 สตริง "Hello" ค่าบูลีน True/False
- ▶ ตัวแปร (Variables) เช่น x, name, is_valid
- ▶ ตัวดำเนินการ (Operators) เช่น +, -, *, /, **, and, or
- ▶ ฟังก์ชัน (Functions) เช่น len(), sum(), print()
- ▶ นิพจน์ย่อย (Subexpressions) เช่น $(x + y) * 2$, `not (x or y)`

ตัวอย่างนิพจน์

- ▶ $2 + 3 \rightarrow$ เป็นนิพจน์ของค่าคงที่และตัวดำเนินการบวก ผลลัพธ์คือ 5
- ▶ $x * y \rightarrow$ เป็นนิพจน์ของตัวแปรและตัวดำเนินการคูณ
- ▶ $\text{len}(\text{"Hello"}) + 42 \rightarrow$ เป็นนิพจน์ที่ประกอบด้วยฟังก์ชัน ค่าคงที่ และตัวดำเนินการบวก
- ▶ $(x + y) * z \rightarrow$ เป็นนิพจน์ที่มีนิพจน์ย่อย $(x + y)$ และใช้วงเล็บเพื่อควบคุมลำดับการคำนวณ

แบบฝึกหัด

1. จงเขียนโปรแกรมคำนวณค่าของนิพจน์ต่อไปนี้ $c * d - x / y$
กำหนดให้ตัวแปรแต่ละตัวมีค่าดังนี้

$$c = 16$$

$$d = 10$$

$$x = 5$$

$$y = 2$$

แบบฝึกหัด

2. จงเขียนโปรแกรมคำนวณค่าของนิพจน์ต่อไปนี้

$-(-5) * (x \% y - x * (y+1))$ กำหนดให้ตัวแปรแต่ละตัวมีค่าดังนี้

$$c = 16$$

$$d = 10$$

$$x = 5$$

$$y = 2$$

แบบฝึกหัด

3. จงเขียนโปรแกรมคำนวณค่าของนิพจน์ต่อไปนี้

$c // d + y \% x \neq 2 * c - d$ กำหนดให้ตัวแปรแต่ละตัวมีค่าดังนี้

$$c = 16$$

$$d = 10$$

$$x = 5$$

$$y = 2$$