

บทที่ 1 ทฤษฎีจำนวน (Number Theory)

ในบทนี้เรามาทำความรู้จักกับภาษา Python เบื้องต้น เพื่อเป็นพื้นฐานในการเขียนโปรแกรม และทำความเข้าใจวิชาคณิตศาสตร์ได้ดียิ่งขึ้น โดยเสนอเรื่องการเขียนข้อความอธิบายในโปรแกรม, การแสดงผลออกทางหน้าจอ, การสร้างและกำหนดค่าให้ตัวแปร, ตัวดำเนินการทางคณิตศาสตร์ และการแก้สมการเบื้องต้นโดยใช้ภาษา Python

หมายเหตุ การคำนวณ ค.ร.น., ห.ร.ม. อยู่ในบทฟังก์ชัน

1. การเขียนข้อความอธิบายในโปรแกรม

การเขียนข้อความอธิบายในโปรแกรม ใช้เครื่องหมาย # หรือ " (3 single quote) โดยข้อความจะไม่ถูกนำไปประมวลผล เครื่องหมาย # และ " ใช้งานแตกต่างกันดังนี้

ใช้กับการเขียนข้อความอธิบายในโปรแกรมเพียง 1 บรรทัด โดยข้อความที่อยู่หลังเครื่องหมาย # จะเปลี่ยนเป็นคำอธิบายทั้งหมด

" ใช้กับการเขียนข้อความอธิบายในโปรแกรมหลายบรรทัด โดยข้อความที่อยู่ระหว่างเครื่องหมาย " " จะเปลี่ยนเป็นคำอธิบายทั้งหมด

2. การแสดงผลออกทางหน้าจอ

การแสดงผลออกทางหน้าจอ นิยมใช้คำสั่ง print() สามารถใช้แสดงผลกับข้อมูลชนิดต่าง เช่น ข้อความ ตัวเลข, ตัวแปร หรือผสมกัน

ตัวอย่าง 1 แสดงข้อความ Hello world ออกทางหน้าจอ

```
1: print("Hello, World!")
```

ผลลัพธ์

```
Hello, World!
```

อธิบาย

บรรทัด 1 คำสั่ง print("Hello world!") แสดง Hello World! ออกทางหน้าจอ

หมายเหตุ เราสามารถใช้ ' ' หรือ " " ก็ได้

ตัวอย่าง 2 แสดงการขึ้นบรรทัดใหม่ โดย \n ในข้อความ

```
1: print("Hello, \nWorld!")
```

ผลลัพธ์

```
Hello,  
World!
```

อธิบาย

บรรทัด 1 คำสั่ง print("Hello, \nWorld!") แสดง Hello, World! ออกทางหน้าจอ โดย \n ถูกใส่ไว้หน้าข้อความ, ตัวเลข, ตัวแปรที่ต้องการขึ้นบรรทัดใหม่

หมายเหตุ \n ต้องอยู่ภายใน ' ' หรือ " " เท่านั้น

เราแสดงข้อความเดิมซ้ำกันได้ โดยใช้ * ตามด้วยจำนวนครั้งที่ต้องการแสดงซ้ำหลังเครื่องหมาย "

ตัวอย่าง 3 แสดง Hello world! 3 ครั้ง

```
1: print("Hello, World! " * 3)
```

ผลลัพธ์

```
Hello, World! Hello, World! Hello, World!
```

อธิบาย

บรรทัด 1 คำสั่ง print("Hello, world!" * 3) เพื่อแสดง Hello, World! 3 ครั้ง ออกทางหน้าจอ

การเชื่อมข้อความกับข้อความ ใช้เครื่องหมาย + ในกรณีที่มีข้อความ (String) เพียงอย่างเดียว

ตัวอย่าง 4 แสดงการเชื่อมระหว่างข้อความกับข้อความ

```
1: a = 'Python!'
2: print('Hello, ' + a)
```

ผลลัพธ์

```
Hello, Python!
```

อธิบาย

บรรทัด 1 สร้างตัวแปร a เก็บข้อความ 'Python!'

บรรทัด 2 คำสั่ง print('Hello, ' + a) เนื่องจาก 'Hello, ' เป็นข้อความ และตัวแปร a มีค่าเท่ากับ 'Python!' ซึ่งเป็นข้อความเช่นกัน ทำให้สามารถใช้เครื่องหมาย + เชื่อมกันได้

การเชื่อมข้อความและตัวเลข ใช้เครื่องหมาย , ในการเชื่อมระหว่างการเชื่อมข้อความและตัวเลข

ตัวอย่าง 5 แสดงการเชื่อมระหว่างข้อความกับตัวเลข

```
1: print('Input', 10, 'Number')
```

ผลลัพธ์

```
Input 10 Number
```

อธิบาย

บรรทัด 1 คำสั่ง print('Input', 10, 'Number') แสดงข้อความ Input 10 Number ทางหน้าจอ

ข้อควรระวัง รูปแบบดังต่อไปนี้ ทำให้เกิดข้อผิดพลาด (Error)

```
print('Input = '+10)
print(10+' = Number')
print('Input = '+10+'Number')
print('Input = '+(10+5)+'Number')
```

เนื่องจากในภาษา Python ไม่สามารถเชื่อมข้อความและตัวเลขได้

3. การสร้างและกำหนดค่าตัวแปร

การสร้างตัวแปร หรือการประกาศตัวแปร คือ การจองเนื้อที่ในหน่วยความจำของคอมพิวเตอร์เพื่อใช้สำหรับเก็บค่า ๆ หนึ่ง โดยผู้เขียนโปรแกรมจะกำหนดค่าให้ตัวแปรด้วยเครื่องหมาย (=) เช่น a = 13

ตัวอย่าง 6 แสดงการสร้าง และกำหนดค่าให้ตัวแปร

```
1: a = 7+3
2: print(a)
```

ผลลัพธ์

```
10
```

อธิบาย

- บรรทัด 1 สร้างตัวแปร a กำหนดค่า 7+3 ซึ่งเท่ากับ 10
บรรทัด 2 คำสั่ง print(a) แสดงค่าตัวแปร a ทางหน้าจอ

ตัวอย่าง 7 แสดงการเปลี่ยนค่าของตัวแปร

```
1: a = 5
2: a = 10
3: print(a)
```

ผลลัพธ์

10

อธิบาย

- บรรทัด 1 สร้างตัวแปร a กำหนดค่า 5 ให้ตัวแปร a
บรรทัด 2 กำหนดค่า 10 ให้ตัวแปร a
บรรทัด 3 คำสั่ง print(a) แสดงค่าตัวแปร a ทางหน้าจอ

4. ตัวดำเนินการทางคณิตศาสตร์

ตาราง 1- 1 ตัวดำเนินการทางคณิตศาสตร์

เครื่องหมาย	ความหมาย	ไวยากรณ์	ตัวอย่าง
+	บวก, addition	$a + b$	$5 + 10 = 15$
-	ลบ, subtraction	$a - b$	$14 - 5 = 9$
*	คูณ, multiplication	$a * b$	$5 * 8 = 40$
**	ยกกำลัง, exponents	$a ** b$	$4 ** 3 = 64$
/	หาร, division	a / b	$14 / 3 = 4.67$
//	หารเอาจำนวนเต็ม, floor division	$a // b$	$14 // 3 = 4$
%	หารเอาเศษ, modulo	$a \% b$	$8 \% 3 = 2$

โดยมีอันดับการคำนวณก่อนหลังเหมือนกับการคำนวณทางคณิตศาสตร์ มีรายละเอียดดังนี้

1. () (วงเล็บ, parentheses)
2. ** (ยกกำลัง, exponents)
3. *, /, //, % (คูณ, multiplication), (หาร, true division), (หารเอาจำนวนเต็ม, floor division), (หารเอาเศษ, modulo)
4. +, - (บวก, addition), (ลบ, subtraction)

ตัวอย่าง 8 แสดงลำดับการคำนวณของตัวดำเนินการ

$$5 * 2 + (9 - 4) = 5 * 2 + 5 = 10 + 5 = 15$$

อธิบาย

1. การดำเนินการจะเริ่มทำใน () (วงเล็บ, parentheses) ก่อน เนื่องจากวงเล็บมีอันดับการคำนวณทางคณิตศาสตร์เป็นอันดับแรก เช่น $5 * 2 + (9 - 4) = 5 * 2 + 5$
2. จากนั้น Python จะเลือกดำเนินการเครื่องหมาย * (คูณ, multiplication) ก่อนเครื่องหมาย + (บวก, addition) พบว่าคุณมีอันดับการคำนวณทางคณิตศาสตร์เป็นลำดับต่อมา เช่น $5 * 2 + 5 = 10 + 5$
3. ทำเครื่องหมาย + (บวก, addition) เป็นลำดับสุดท้าย เนื่องจากมีอันดับการคำนวณทางคณิตศาสตร์เป็นลำดับสุดท้าย เช่น $10 + 5 = 15$

ตัวดำเนินการแต่ละตัวมีรายละเอียดดังนี้

4.1 การบวก

a+b

จงหาผลบวกของจำนวนต่อไปนี้ (ให้หาคำตอบด้วยมือ หรือ เครื่องคิดเลข)

- | | | | |
|----|----------|---|----------------------|
| 1) | 36 + 64 | = | <input type="text"/> |
| 2) | 47 + 52 | = | <input type="text"/> |
| 3) | 71 + 108 | = | <input type="text"/> |

การเขียนโปรแกรมเพื่อหาผลบวกของ 15 + 5

<pre>a = 15 + 5 print(a)</pre>	ผลลัพธ์ 20
--------------------------------	----------------------

- | | | | |
|----|----------|---|----------------------|
| 1) | 72 + 45 | = | <input type="text"/> |
| 2) | 72 + 652 | = | <input type="text"/> |
| 3) | 91 + 409 | = | <input type="text"/> |
| 4) | 99 + 12 | = | <input type="text"/> |
| 5) | 91 + 108 | = | <input type="text"/> |

4.2 การลบ

a-b

จงหาผลลบของจำนวนต่อไปนี้ (ให้หาคำด้วยมือ หรือ เครื่องคิดเลข)

- | | | | |
|----|-----------|---|----------------------|
| 1) | $51 - 1$ | = | <input type="text"/> |
| 2) | $62 - 27$ | = | <input type="text"/> |
| 3) | $95 - 36$ | = | <input type="text"/> |

การเขียนโปรแกรมเพื่อหาผลลบของ $15 - 5$

<pre>a = 15 - 5 print(a)</pre>	ผลลัพธ์ 10
--------------------------------	----------------------

- | | | | |
|----|-------------|---|----------------------|
| 1) | $108 - 47$ | = | <input type="text"/> |
| 2) | $111 - 12$ | = | <input type="text"/> |
| 3) | $425 - 26$ | = | <input type="text"/> |
| 4) | $265 - 112$ | = | <input type="text"/> |
| 5) | $442 - 22$ | = | <input type="text"/> |

4.3 การคูณ

$a \times b$

จงหาผลคูณของจำนวนต่อไปนี้ (ให้หาคำด้วยมือ หรือ เครื่องคิดเลข)

- | | | | |
|----|----------------|---|----------------------|
| 1) | 4×10 | = | <input type="text"/> |
| 2) | 7×5 | = | <input type="text"/> |
| 3) | 11×11 | = | <input type="text"/> |

การเขียนโปรแกรมเพื่อหาผลคูณของ 15×5

<pre>a = 15 * 5 print(a)</pre>	ผลลัพธ์ 75
--------------------------------	----------------------

- | | | | |
|----|----------------|---|----------------------|
| 1) | 5×229 | = | <input type="text"/> |
| 2) | 8×139 | = | <input type="text"/> |
| 3) | 23×50 | = | <input type="text"/> |
| 4) | 27×37 | = | <input type="text"/> |
| 5) | 29×59 | = | <input type="text"/> |

4.4 การหารทศนิยม

$a \div b$ หรือ a/b

จงหาผลหารของจำนวนต่อไปนี้ (ให้นักสิตทำด้วยมือ หรือ เครื่องคิดเลข)

- | | | | |
|----|-------------|---|---|
| 1) | $12 \div 3$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 2) | $4 \div 6$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 3) | $45 \div 5$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |

การเขียนโปรแกรมเพื่อหาผลหาร $15 \div 5$

```
a = 15 / 5
print(a)
```

ผลลัพธ์

3.0

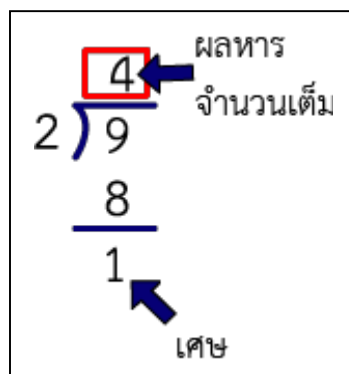
- | | | | |
|----|----------------|---|---|
| 1) | $63 \div 7$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 2) | $72 \div 4$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 3) | $121 \div 11$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 4) | $625 \div 25$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 5) | $1024 \div 16$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |

4.5 การหารเอาผลหารที่เป็นจำนวนเต็ม

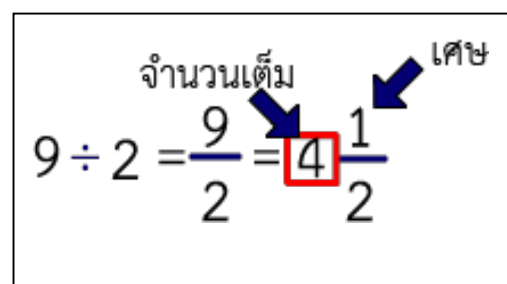
$a // b$

การหารเอาจำนวนเต็ม เป็นการดำเนินการกับจำนวนเต็มเพื่อเอาเฉพาะผลหารที่เป็นจำนวนเต็มเท่านั้น โดยที่ a และ b เป็นจำนวนเต็ม และ b มีค่ามากกว่า 1

ตัวอย่าง $99 // 2 = 4$



หรือ



จงหาผลหารจำนวนเต็มต่อไปนี้ต่อไปนี้ (ให้นักสิตทำด้วยมือ หรือ เครื่องคิดเลข)

- | | | | |
|----|------------|---|---|
| 1) | $4 // 2$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 2) | $6 // 3$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 3) | $7 // 2$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |
| 4) | $-16 // 5$ | = | <div style="border: 1px solid black; width: 50px; height: 20px;"></div> |

การเขียนโปรแกรมเพื่อผลลัพธ์ของ $77//7$

```
a = 77 // 7
print(a)
```

ผลลัพธ์

11

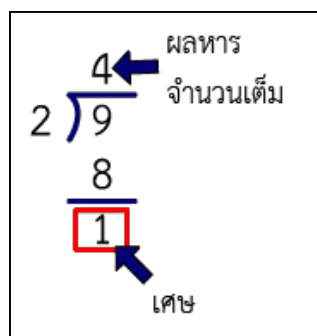
- | | | | |
|----|---------|---|----------------------|
| 1) | 88//8 | = | <input type="text"/> |
| 2) | 199//7 | = | <input type="text"/> |
| 3) | 212//4 | = | <input type="text"/> |
| 4) | 315//15 | = | <input type="text"/> |
| 5) | -58//7 | = | <input type="text"/> |
| 6) | 498//83 | = | <input type="text"/> |

4.6 การหารเอาเศษ

$a \bmod b$

mod หรือ modulo เป็นการดำเนินการกับจำนวนเต็มเพื่อหาเศษที่เหลือจากการหาร โดยที่ a และ b เป็นจำนวนเต็ม และ b มีค่ามากกว่า 1

ตัวอย่าง $10 \div 9 \bmod 2 = 1$



หรือ

$$9 \div 2 = \frac{9}{2} = 4 \frac{1}{2}$$

The fraction $\frac{1}{2}$ is highlighted with a red box, and the numerator 1 is labeled 'เศษ' (remainder).

จงหาเศษต่อไปนี้ (ให้นักเรียนทำด้วยมือ หรือ เครื่องคิดเลข)

- | | | | |
|----|----------------|---|----------------------|
| 1) | $4 \bmod 2$ | = | <input type="text"/> |
| 2) | $7 \bmod 4$ | = | <input type="text"/> |
| 3) | $102 \bmod 12$ | = | <input type="text"/> |
| 4) | $-74 \bmod 9$ | = | <input type="text"/> |

การเขียนโปรแกรมการหาเศษด้วย mod (%)

```
a = 77 % 6
print(a)
```

ผลลัพธ์

5

- 1) $66 \bmod 9 =$
- 2) $73 \bmod 2 =$
- 3) $56 \bmod 5 =$
- 4) $165 \bmod 7 =$
- 5) $1852 \bmod 6 =$
- 6) $-190 \bmod 6 =$

4.7 การหารลงตัว

a | b

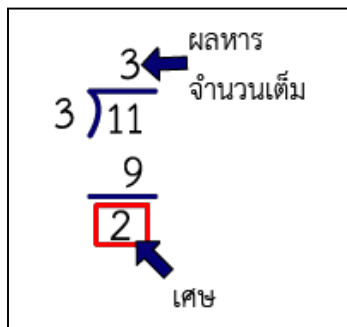
(อ่านว่า a หาร b ลงตัว)

หาก a หาร b ลงตัวแล้ว เราสามารถเขียนสัญลักษณ์ได้ ดังนี้ $a \mid b$ ซึ่ง a หาร b ลงตัว

เมื่อ $b = \text{เลขจำนวนเต็ม} \times a$, หรือ a หาร b แล้วไม่มีเศษ (เศษเป็น 0 หรือ $b \bmod a$ แล้วจะเท่ากับ 0)

ตัวอย่าง $11 \mid 3 \mid 11$ เป็นจริงหรือไม่

$3 \mid 11$ ไม่เป็นจริง เพราะได้เศษ 2



หรือ

$$11 \div 3 = \frac{11}{3} = 3 \frac{2}{3}$$

จำนวนเต็ม

เศษ

ตรวจสอบข้อต่อไปว่าเป็นจริงหรือไม่ (ให้นิสิตทำด้วยมือ หรือ เครื่องคิดเลข)

ตาราง 1 - 2 ตรวจสอบการหารลงตัวด้วยมือ

โจทย์	หารลงตัวหรือไม่	ผลหารที่เป็นจำนวนเต็ม	เศษ
$6 \mid 56$	หารไม่ลงตัว	9	2
$5 \mid 5$	หารลงตัว	1	0
$7 \mid 77$			
$4 \mid 82$			

การเขียนโปรแกรมเพื่อตรวจสอบว่า 5 | 50 หรือไม่

<pre>a = 50 b = 5 print(a % b) print(a // b)</pre>	<p>ผลลัพธ์</p> <p>0</p> <p>10</p>
---	-----------------------------------

จงเขียนโปรแกรมคำนวณว่าข้อใดบ้างเป็นการหารลงตัว

ตาราง 1 - 3 ตรวจสอบการหารลงตัวด้วยโปรแกรม

โจทย์	หารลงตัวหรือไม่	ผลหารที่เป็นจำนวนเต็ม	เศษ
6 84			
11 111			
15 625			

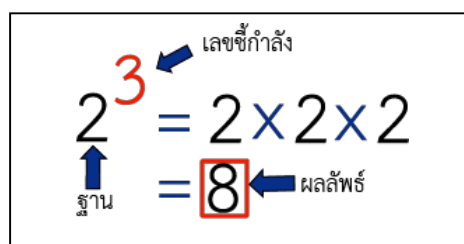
4.8 การยกกำลัง

ab

(เรียก a ว่าฐาน และเรียก b ว่าเลขชี้กำลัง)

การยกกำลัง คือ การคูณซ้ำ ๆ กันของฐาน ตามจำนวนเลขชี้กำลัง

ตัวอย่าง $2^3 = 8$



จงหาผลลัพธ์ของการยกกำลังต่อไปนี้ (ให้หาคำตอบด้วยมือ หรือ เครื่องคิดเลข)

- | | | | |
|----|--------|---|----------------------|
| 1) | 2^5 | = | <input type="text"/> |
| 2) | 5^3 | = | <input type="text"/> |
| 3) | 10^4 | = | <input type="text"/> |

การเขียนโปรแกรมเพื่อหาผลลัพธ์ของ 2^3

<pre>a = 2 b = 3 print(a**b)</pre>	ผลลัพธ์ 8
------------------------------------	--------------

1)	2^{10}	=	
2)	3^5	=	
3)	5^4	=	
4)	7^3	=	
5)	11^3	=	

5. การทดสอบการเป็นจำนวนเฉพาะ

ตัวอย่าง 13 จงตรวจสอบว่า 1573 เป็นจำนวนเฉพาะหรือไม่

```
1:     n = 1573
2:     if n > 1:
3:         for i in range(2, n):
4:             if (n % i) == 0:
5:                 print(n, "is not a prime number")
6:                 print(i, "times", n // i, "is", n)
7:                 break
8:         else:
9:             print(n, "is a prime number")
10:    else:
11:        print(n, "is not a prime number")
```

ผลลัพธ์

```
1573 is not a prime number
11 times 143 is 1573
```

อธิบาย

บรรทัด 1	สร้างตัวแปร n กำหนดค่าเท่ากับ 1573
บรรทัด 2	ตรวจสอบเงื่อนไขว่า n มีค่ามากกว่า 1 หรือไม่
บรรทัด 3	ถ้า n มีค่ามากกว่า 1 จริง จะทำการหาตัวประกอบของ n โดยเริ่มจากตัวประกอบ(i) ที่มีค่าเท่ากับ 2 ไปจนถึง n-1
บรรทัด 4	ตรวจสอบเงื่อนไขว่า i เป็นตัวประกอบของ n (n % i มีค่าเท่ากับ 0) หรือไม่
บรรทัด 5-6	ถ้า i เป็นตัวประกอบของ n แสดงข้อความว่า n is not a prime number (n ไม่เป็นจำนวนเฉพาะ) และแสดงข้อความว่า i คูณกับจำนวนใดมีค่าเท่ากับ n
บรรทัด 7	สิ้นสุดการทำงานซ้ำและไปยังคำสั่งถัดไปจากคำสั่ง for
บรรทัด 8-9	เมื่อสิ้นสุดการทำงานซ้ำแล้ว แสดงข้อความ n is a prime number
บรรทัด 10-11	ถ้า n มีค่าไม่มากกว่า 1 แสดงข้อความ n is not a prime number

6. การแยกตัวประกอบของจำนวนเต็ม

ตัวอย่าง 14 จงหาแยกตัวประกอบของ 1573

```
1:     n = 1573
2:     d = 2
3:     while n > 1:
4:         if n % d == 0:
5:             n = n/d
6:             print(d)
7:         else:
8:             d = d+1
```

ผลลัพธ์

```
11
11
13
```

อธิบาย

- | | |
|------------|--|
| บรรทัด 1 | สร้างตัวแปร n กำหนดค่าเท่ากับ 1573 |
| บรรทัด 2 | สร้างตัวแปร d กำหนดค่าเท่ากับ 2 |
| บรรทัด 3 | ทำคำสั่งในย่อหน้าซ้ำในขณะที่ n มีค่ามากกว่า 1 |
| บรรทัด 4 | ตรวจสอบเงื่อนไขว่า d เป็นตัวประกอบของ n ($n \% d$ มีค่าเท่ากับ 0) หรือไม่ |
| บรรทัด 5 | ถ้า d เป็นตัวประกอบของ n ให้นำค่าผลหารของ n/d ไปกำหนดให้กับตัวแปร n |
| บรรทัด 6 | แสดงค่าตัวแปร d ทางหน้าจอ |
| บรรทัด 7-8 | ถ้า d ไม่เป็นตัวประกอบของ n ให้นำค่า $d+1$ ไปกำหนดให้กับตัวแปร d |