

Basic I/O, Data Types, Variables, Operators, Expressions, Statements

ภาควิชาวิศวกรรมคอมพิวเตอร์

จุฬาลงกรณ์มหาวิทยาลัย

๒๕๖๒

โครงสร้างโปรแกรมและลักษณะการทำงาน

ทำงานทีละคำสั่ง
จากบนลงล่าง

บางคำสั่งก็เป็น
แบบเลือกทำ

บางคำสั่งก็
ทำงานเป็นวงวน

```
import matplotlib.pyplot as plt
import math
```

```
n = int(input()) # data points
```

```
if n < 10:
    n = 10
```

comment

```
x = []
```

```
y = []
```

```
for k in range(n):
    x.append(k*16*math.pi/n)
    y.append(0.1*k*math.sin(x[k]))
```

```
plt.plot(x, y)
```

```
plt.show()
```

ต้องเริ่มคำสั่งให้อยู่ในแนวเดียวกันกับคำสั่งอื่นในกลุ่ม

Basic Data Types: String and Number

- String

- ```
- "Hello" 'Hello'
- "Hello Python" 'Hello Python'
- "12345" '12345'
- "" ''
```

- Number

- integer
  - 1234, 0
- floating point
  - -1234.0, 1.5E-15      ←    1.5 × 10<sup>-15</sup>

## Operator: plus

- ข้อความ**บวก**กัน คือข้อความ**ต่อ**กัน
  - "Hello" + " " + "!" ได้ "Hello !"
  - "1" + "1" ได้ "11"
- จำนวน**บวก**กัน
  - 1 + 1 ได้ 2
  - 1 + 1.0 ได้ 2.0
- ข้อความบวกกับจำนวนไม่ได้
  - "1" + 1 ← ผิด

## print( ... ) แสดงผลทางจอภาพ

```
print("Hello")
print("Python")
print("Hello" + " " + "Python")
print("Hello", "Python")
print("a", "b", "c", 1, 2, 3)
print("1+1 =", (1+1))
```




program


```
Hello
Python
Hello Python
Hello Python
a b c 1 2 3
1+1 = 2
```

output

## อักขระพิเศษ (Escape Characters)

```
s = "Hello"
print(s) # Hello
s = "\"Hello\""
print(s) # "Hello"
s = "'Hello'"
print(s) # 'Hello'
s = "\"'\\Hello\\\""
print(s) # "\"'\\Hello\\"
s = "Hello\nPython"
print(s) # Hello
 # Python
```

 `\"` คือ `"`  
 `\'` คือ `'`  
 `\\` คือ `\`

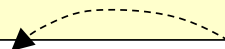
 `\n` คือ รหัสให้ขึ้นบรรทัดใหม่

## ตัวอย่าง: แทน "' / \ ( ) , . : ; ด้วยช่องว่าง

```
def blank(t):
 result = ""
 for c in t:
 if c in "\"\'/\\,.:;":
 result += "
 else:
 result += c
 return result
```

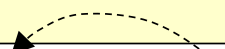
# ตัวแปร (Variables)

- ตัวแปรเป็นที่เก็บข้อมูล
  - ต้องมีชื่อกำกับ
  - เปลี่ยนค่าในที่เก็บได้



```
name = "Python"
```

นำค่า "Python" เก็บในตัวแปร name



```
lang = name
```

นำข้อมูลใน name เก็บในตัวแปร lang  
ทำให้ lang กับ name เก็บค่าเหมือนกัน

```
a = 1 # a = 1
b = 2 # a = 1; b = 2
c = a # a = 1; b = 2; c = 1
d = c + b # a = 1; b = 2; c = 1; d = 3
d = d + 5 # a = 1; b = 2; c = 1; d = 8
```

หลายคนเขียน 1+2 = a ผิด !!!



# ชื่อตัวแปร

- ประกอบด้วยตัวอักษร ตัวเลข หรือเครื่องหมายขีดเส้นใต้
- ตัวอักษรใหญ่กับเล็กไม่เหมือนกัน
- ห้ามขึ้นต้นชื่อด้วยตัวเลข
- ไม่ซ้ำกับคำสงวนของภาษา

```
first_name = "John"
last_name = "Wick"
```

|          |          |         |       |       |
|----------|----------|---------|-------|-------|
| and      | as       | assert  | break | class |
| continue | def      | del     | elif  | else  |
| except   | exec     | finally | for   | from  |
| global   | if       | import  | in    | is    |
| lambda   | nonlocal | not     | or    | pass  |
| raise    | return   | try     | while | with  |
| yield    | True     | False   | None  |       |

## Basic Data Types

- str ข้อความ
- int จำนวนเต็ม
- float จำนวนจริง (มีจุดทศนิยม)
- bool ค่าจริงกับเท็จเท่านั้น (True กับ False)
- list ชุดข้อมูล เก็บเรียงเป็นลำดับ

[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55]

["SU", "MO", "TU", "WE", "TH", "FR", "SA"]

[6230012021, [2110101, 2301107]]

[ ]

## ตัวอย่าง: Variables & Data Types

```
first_name = "Ranee" # str
last_name = "Campen" # str
aka = "Bella" # str
age = 29 # int
height = 1.65 # float
is_single = True # bool
birth_date = [24, 12, 1989] # list
tv_series = ["Roy Marn", "Plerng Boon",
 "Bubphe Sanniwat", "Krong Kam"]
```

```
print("Name:", first_name, last_name)
print("Age:", age)
print("Aka:", aka, "or", aka[0:4:1])
d = birth_date[0]
m = birth_date[1]
y = birth_date[2]
print("Born:", str(d) + "/" + str(m) + "/" + str(y))
```

จะได้ศึกษาการจัดการข้อมูล  
ประเภทต่าง ๆ ต่อไป

## Type Conversions

- ข้อมูลส่วนใหญ่แปลงเป็น string ได้ด้วย str
- ข้อมูลบางอย่างก็อาจแปลงเป็น int กับ float ได้

```
s1 = "123"
s2 = " 456"
n = int(s1) + int(s2) # 579
f = float(s1) + float(s2) # 579.0
print(s1+s2, n, f)
print(s1+s2 + ", " + str(n) + ", " + str(f))
```

```
a = int(1) # a = 1
b = int(1.9) # b = 1
c = int(1.9 + 0.5) # c = 2
d = float(1.1) # d = 1.1
e = float(1) # e = 1.0
f = str("string") # f = "string"
```

## input() รับสตริงจากแป้นพิมพ์

**input()** รอรับข้อมูลที่ป้อนทางคีย์บอร์ด  
พอกด enter ข้อมูลนั้นจะกลายเป็นสตริง  
นำมาเก็บในตัวแปร name

```
name = input()
print(name + " is a very easy language.")
print("We use " + name + " in our class.")
print("Hello " + name + ".")
```

program

หลังจากสั่งทำงานแล้วกดคำว่า **Python** ก็จะได้ผลเป็น

```
Python is a very easy language.
We use Python in our class.
Hello Python.
```

output

## รับจำนวน คือ รับสตริงแล้วเปลี่ยนเป็นจำนวน

```
x = input()
d = float(x)
perimeter = d + d + d + d
print("Perimeter of square =", perimeter)
```

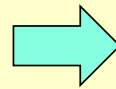
program

หลังจากสั่งทำงานแล้วกด **12** ก็จะได้ผลเป็น

```
Perimeter of square = 48.0
```

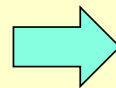
output

```
x = input()
d = float(x)
...
```



```
d = float(input())
... รับจำนวนจริง
```

```
x = input()
d = int(x)
...
```



```
d = int(input())
... รับจำนวนเต็ม
```

## Basic Arithmetic Operations

|    |              |
|----|--------------|
| +  | บวก          |
| -  | ลบ           |
| *  | คูณ          |
| /  | หาร          |
| // | หารปัดเศษ    |
| %  | เศษจากการหาร |
| ** | ยกกำลัง      |
| =  | กาให้ค่า     |

```
a = 5 + 2 # 7
b = 5 - 2 # 3
c = 5 * 2 # 10
d = 5 ** 2 # 25
e = 5 / 2 # 2.5
f = 5 // 2 # 2
g = 5 % 2 # 1
```

```
a,b,c = c,a,b # a = 10
 # b = 7
 # c = 3
```

```
a,b = b,a # สลับค่า a กับ b
```

## Tips

```
a = float(input())
a = -a # เปลี่ยนจำนวนลบเป็นบวก จำนวนบวกเป็นลบ
b = int(a) # บัดเศษของ a ทั้ง
c = int(a + 0.5) # บัดเศษของ a แบบมีขึ้นหรือลง
d = a - int(a) # ค่าหลังจุดทศนิยมของ a
d = a % 1 # ค่าหลังจุดทศนิยมของ a
e = b % 10 # เลขหลักหน่วยของ b
f = b // 10 % 10 # เลขหลักสิบของ b
g = b // 10**4 % 10 # เลขหลักหมื่นของ b

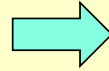
h = input() # h เก็บสตริง
h = int(h) # h เก็บจำนวนเต็ม (ใช้ตัวแปรซ้ำได้)
```



## แบบฝึกหัด: รับเซลเซียส แสดงฟาเรนไฮต์

Input

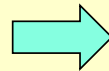
0



Output

32

100



312

```
celsius = float(input())
```

# Augmented Assignments

```
a = 10
a = a + 5 # 15
a = a - 1 # 14
a = a * 2 # 28
a = a % 10 # 8
a = a ** 2 # 64
a = a // 10 # 6
```

```
a = 10
a += 5 # 15
a -= 1 # 14
a *= 2 # 28
a %= 10 # 8
a **= 2 # 64
a //= 10 # 6
```

# math module

```
import math
```

```
degree = float(input())
```

```
radian = degree * 3.14159 / 180
```

```
radian = degree * math.pi / 180
```

```
radian = math.radians(degree)
```

```
s = math.sin(radian)
```

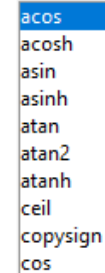
```
c = math.cos(radian)
```

```
g = math.log(1E100, 10) # g = 100.0
```

มีฟังก์ชันทางคณิตศาสตร์  
มากมายให้ใช้ใน math  
module

```
import math
```

```
a = math.
```



## Built-in Functions

```
a = abs(-2) # a = 2
b = round(2/3, 2) # b = 0.67
c = max([4,1,5,3]) # c = 5
d = min([4,1,5,3]) # d = 1
e = sum([4,1,5,3]) # e = 13
f = len([4,1,5,3]) # f = 4
g = str(1234) # g = "1234"
h = int("123") # h = 123
j = float("-123.4") # j = -123.4
k = input()
print(a,b,c)
```

จะได้เรียนวิธีการเขียนฟังก์ชันใหม่ ๆ ไว้ใช้เองต่อไป

## ใช้ฟังก์ชันซ้อน ๆ กันได้

```
x1 = input()
x2 = int(x1)
x3 = abs(x2)
x4 = 1 + x3
x5 = str(x4)
print("x5 = " + x5)
```

```
print("x5 = " + str(1 + abs(int(input()))))
```

เขียนซ้อนมากขึ้นเกินไป จะเข้าใจยาก

## Operator Precedence: $2 + 3 * 4 = ?$

- ลำดับการคำนวณ จากก่อนไปหลังเป็นดังนี้
  - ในวงเล็บ
  - ยกกำลัง
  - คูณ
  - $*$   $/$   $//$   $\%$
  - บวก และ ลบ
  - ถ้าพบหลายตัวที่สำคัญเท่ากัน ให้ทำตัวซ้ายไปขวา  
(ยกเว้นยกกำลัง ทำขวามาซ้าย เช่น  $2*3**2 = 2**9 = 512$ )

**PEMDAS:** **P**arentheses  
**E**xponential  
**M**ultiply  
**D**ivide  
**A**dd  
**S**ubtract

|   |   |   |   |   |   |          |   |         |
|---|---|---|---|---|---|----------|---|---------|
| 2 | * | 3 | + | 8 | / | -(2 - 4) | - | 2**2**3 |
| 2 | * | 3 | + | 8 | / | -(-2)    | - | 2**2**3 |
| 2 | * | 3 | + | 8 | / | -(-2)    | - | 2**8    |
| 2 | * | 3 | + | 8 | / | -(-2)    | - | 256     |
| 2 | * | 3 | + | 8 | / | 2        | - | 256     |
| 6 |   |   | + | 8 | / | 2        | - | 256     |
| 6 |   |   | + |   |   | 4.0      | - | 256     |
|   |   |   |   |   |   | 10.0     | - | 256     |
|   |   |   |   |   |   |          | - | 246.0   |

## ตัวอย่าง: คำนวณพื้นที่และเส้นรอบวงของวงกลม

```
radius = float(input())
area = math.pi * radius**2
circum = 2 * math.pi * radius
print("Area =", round(area, 2))
print("Circumference =", round(circum, 2))
```

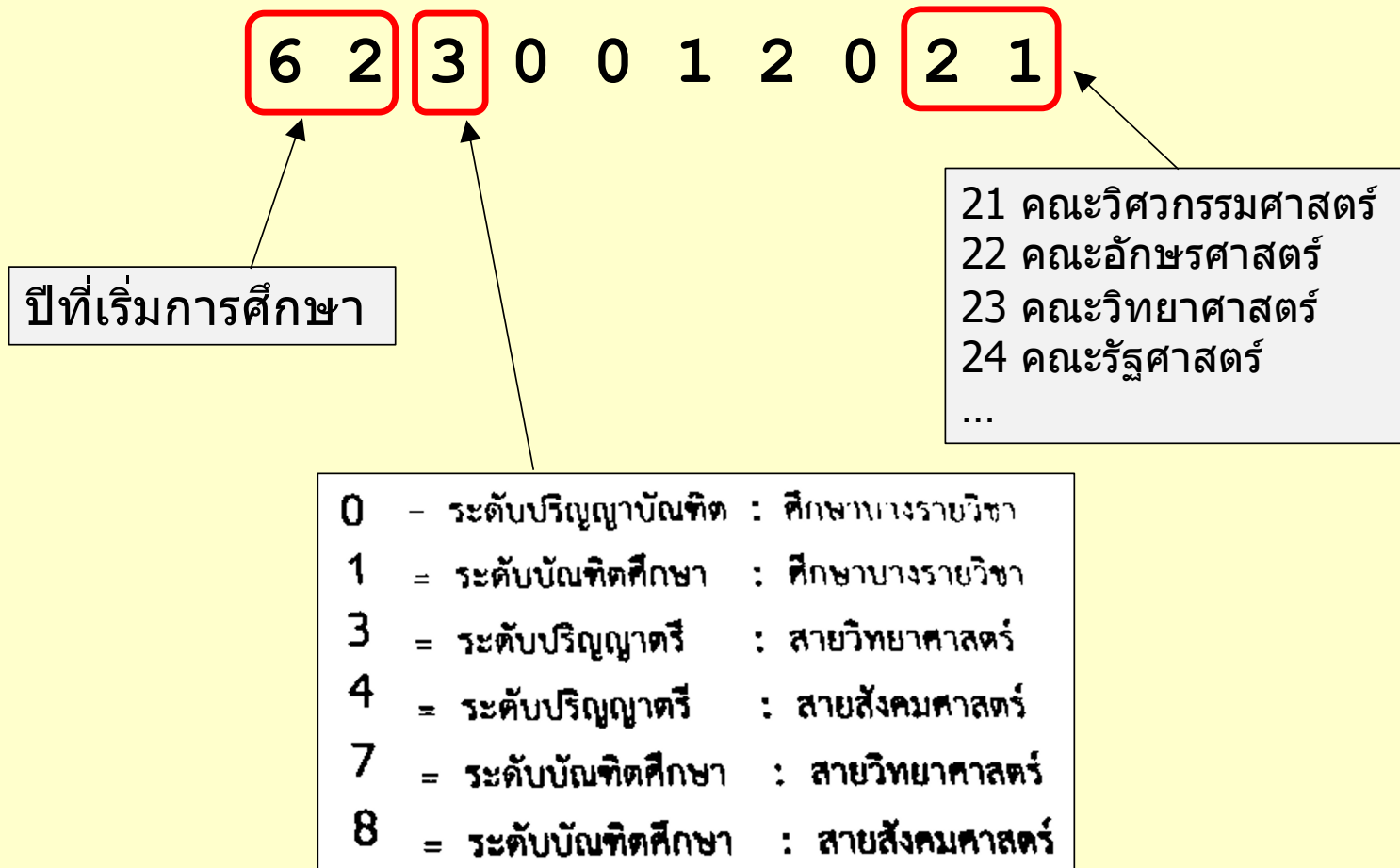
program

หลังจากสั่งทำงานแล้วป้อน 4.5 จะได้ผลเป็น

```
Area = 63.62
Circumference = 28.27
```

output

# เลขประจำตัวนิสิต





## ตัวอย่าง: รหัสนิสิต → คณะอะไร เข้าปีอะไร ระดับใด

```
stu_id = int(input())
print("Student ID:", stu_id)
fac_code = stu_id % 100 # เลือกสองตัวขวา
year_in = 2500 + stu_id//10**8 # ตัด 8 ตัวขวา
deg_code = stu_id//10**7 % 10
print("Faculty code:", fac_code)
print("Enrollment year:", year_in)
print("Academic degree:", deg_code)
```

program

หลังจากสั่งทำงานแล้วป้อน 6230012021 จะได้ผลเป็น

```
Student ID: 6230012021
Faculty code: 21
Enrollment year: 2562
Academic degree: 3
```

output

## แสดงเลขประจำตัวนิสิตที่ละหลัก หลักละบรรทัด

```
sid
sid = int(input()) # 6231020121
d = sid % 10; print(d); sid //= 10 # 623102012
d = sid % 10; print(d); sid //= 10 # 62310201
d = sid % 10; print(d); sid //= 10 # 6231020
d = sid % 10; print(d); sid //= 10 # 623102
d = sid % 10; print(d); sid //= 10 # 62310
d = sid % 10; print(d); sid //= 10 # 6231
d = sid % 10; print(d); sid //= 10 # 623
d = sid % 10; print(d); sid //= 10 # 62
d = sid % 10; print(d); sid //= 10 # 6
d = sid % 10; print(d)
```

เครื่องหมาย `;` คั่นคำสั่งที่ต้องการเขียนต่อกันในบรรทัดเดียวกัน

## แบบฝึกหัด: Stirling Formula

$$n! \sim \sqrt{2\pi n} \left(\frac{n}{e}\right)^n$$

```
import math
n = int(input())
```

## แบบฝึกหัด: :รากของ $ax^2+bx+c = 0$

รับ  $a$   $b$   $c$  จากแป้นพิมพ์  
คำนวณและแสดงสองรากจากสูตร

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