



เขียนโปรแกรมเบื้องต้น ด้วยภาษา **Python**

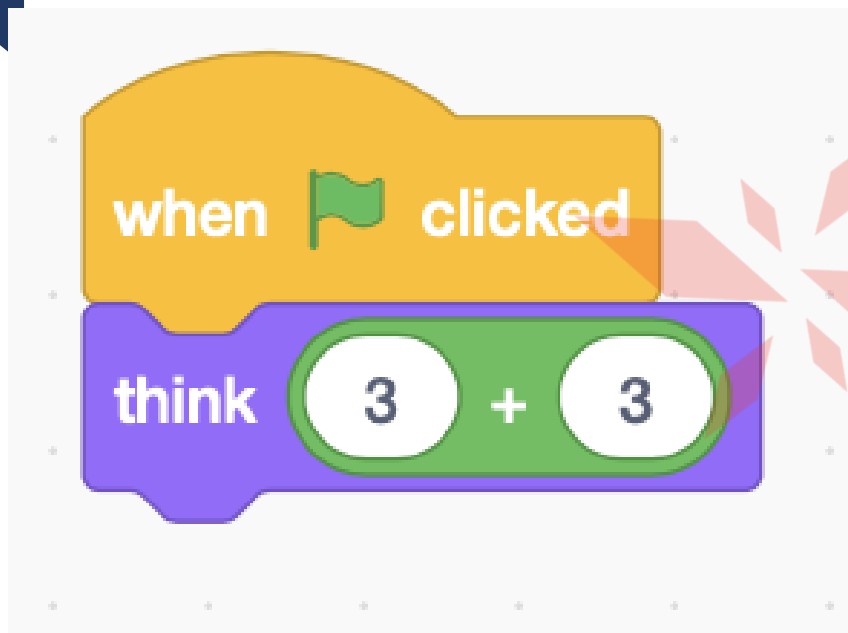
ธีรวัฒน์ ประกอบผล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



พื้นฐานการพัฒนาโปรแกรม
ด้วยภาษา Python

ภาษาโปรแกรม



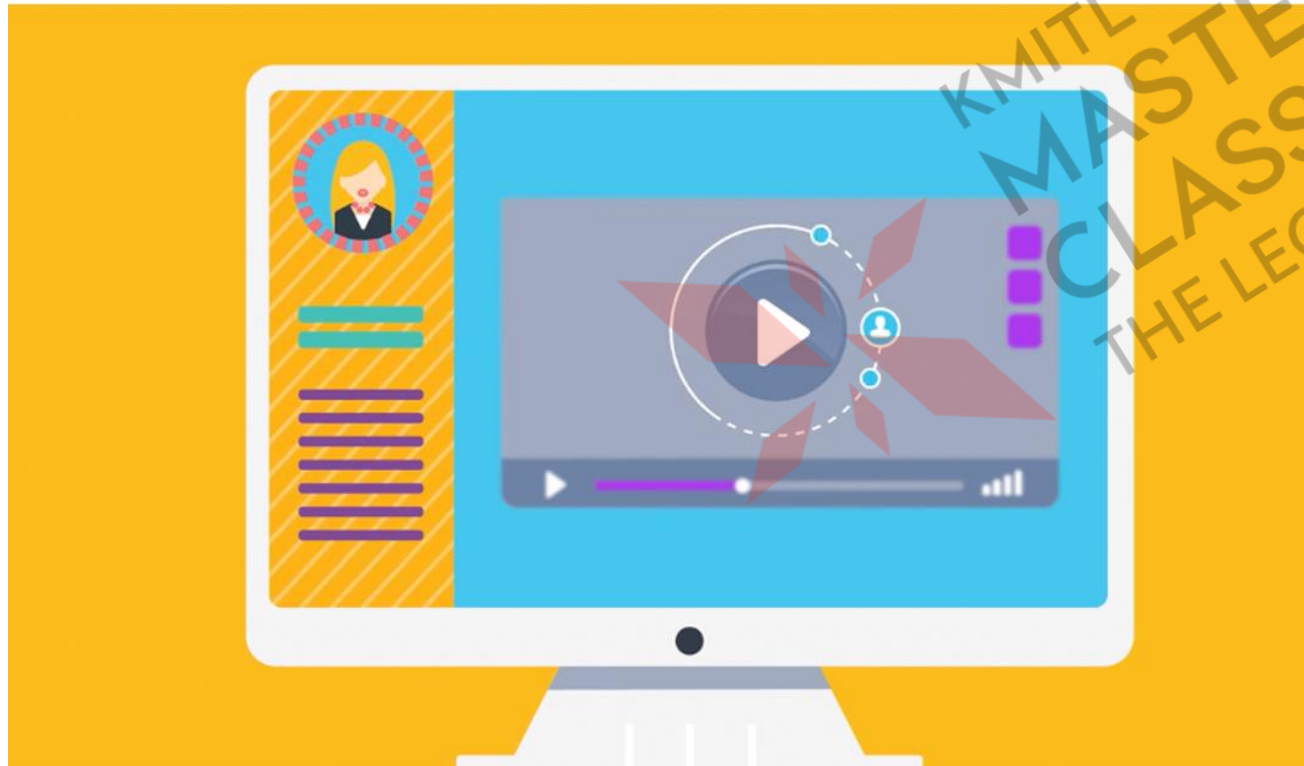
KMITL
MASTER
CLASS
THE LEGEND 2021

```
>>> 3 + 3
```

```
6
```

<https://www.python.org>

การเขียนโปรแกรมภาษาไพทอน



- เกิดขึ้นเมื่อปี ค.ศ. 1980
- ต้องการให้เป็นภาษาที่เขียนสะดวก
- ใช้ได้ฟรี



ทำงานร่วมกับภาษาอื่น ๆ ได้ดี

โปรแกรมที่ไม่ซับซ้อน -> Python ตัวเดียว

งานอื่น ๆ เช่น Data Science -> Python + packages

ตัวอย่างงานที่ใช้



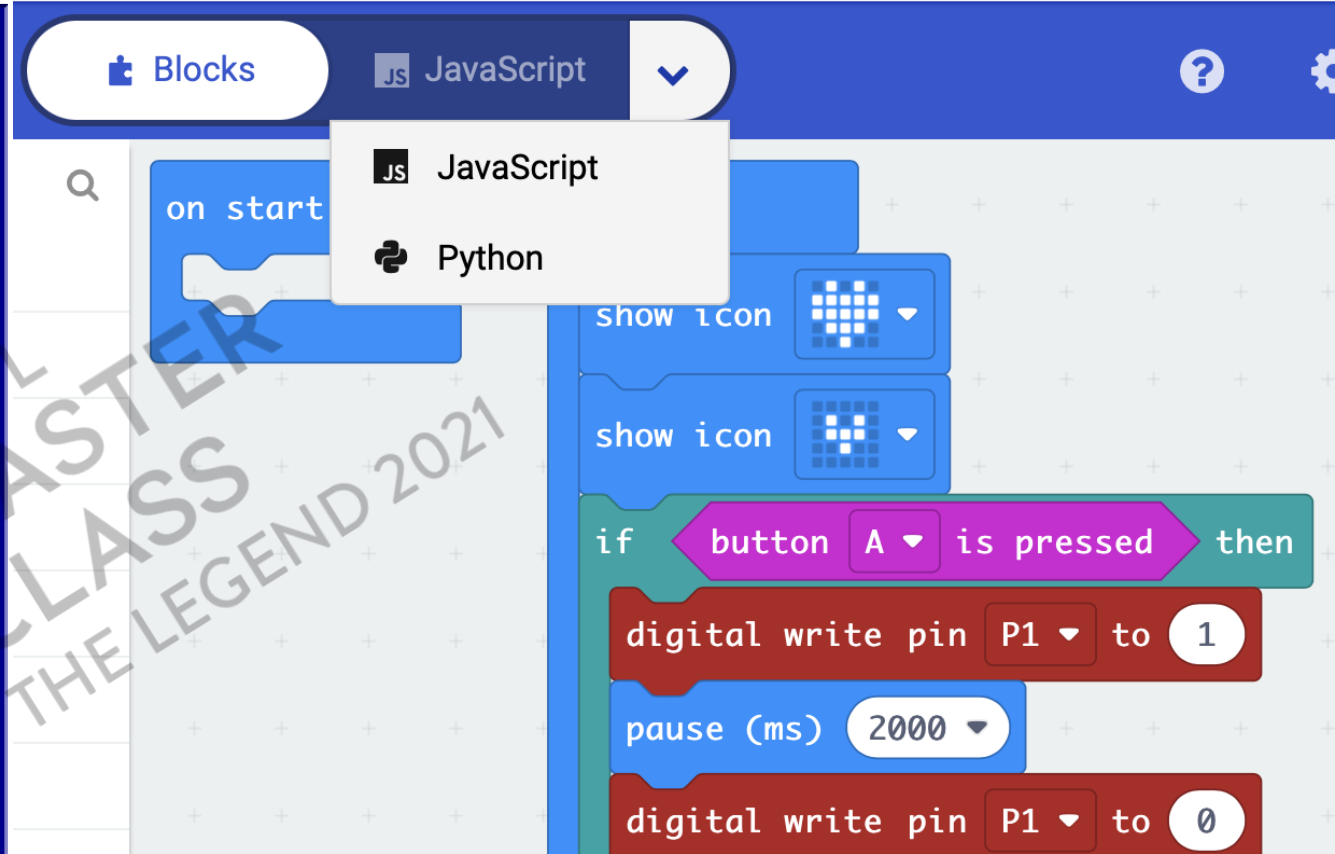
Google

YAHOO!

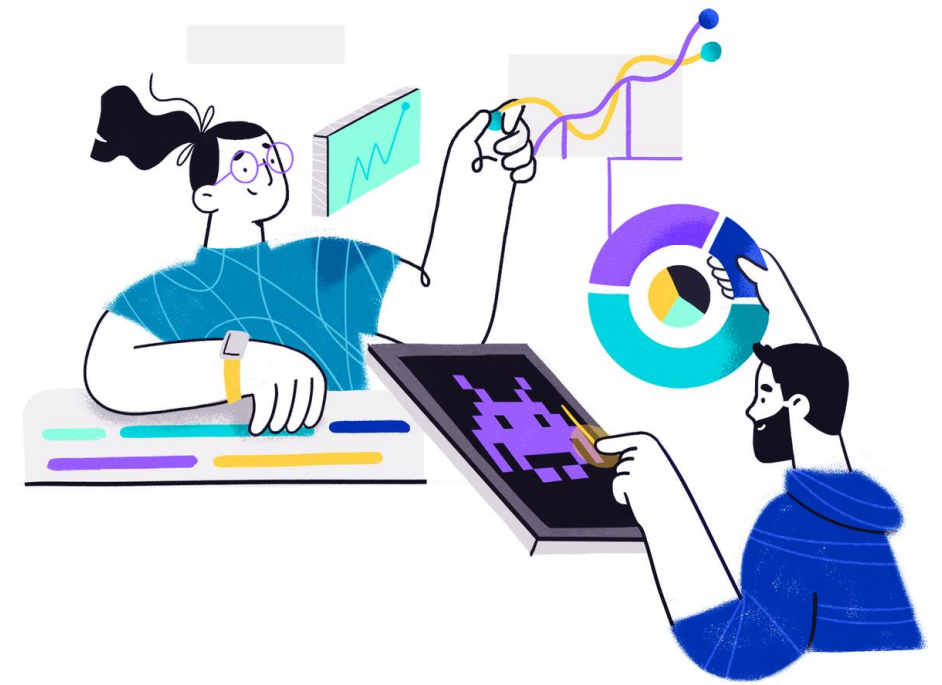
NASA

KMITL
MASTER
CLASS
THE LEGEND 2021

ตัวอย่างงานจาก Python



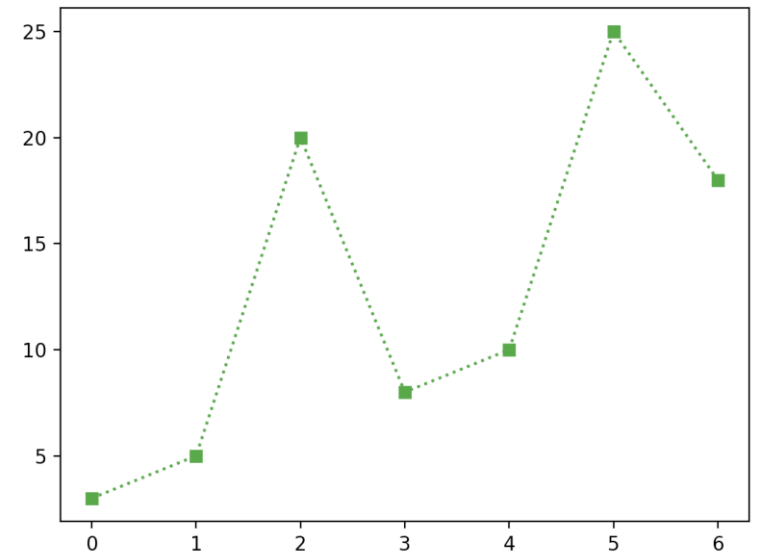
Data Visualization



บันทึกกราฟเป็นไฟล์

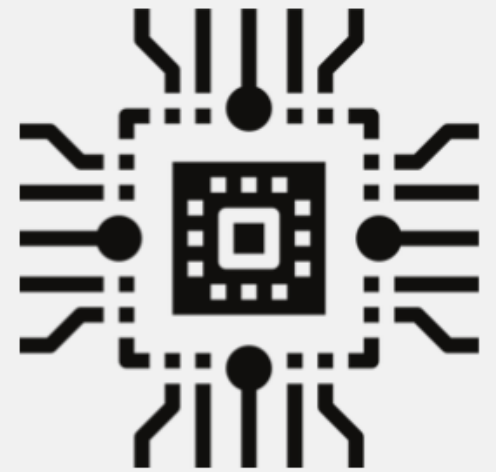
```
import matplotlib.pyplot as plt  
plt.plot([3,5,20,8,10,25,18], 's:C2')
```

```
plt.savefig('graphplot.png')  
plt.show()
```



การทำงาน ของคอมพิวเตอร์

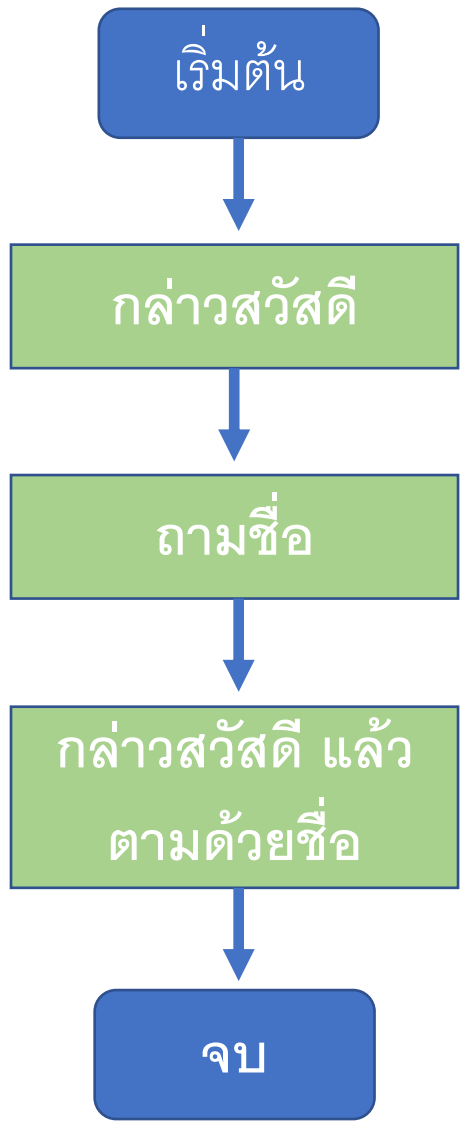
KMITL
MASTER
CLASS
THE LEGEND 2021



งานที่ คอมพิวเตอร์ทำ

คอมพิวเตอร์จะทำงานได้ดี มาจากความคิดที่ดีของมนุษย์
มนุษย์คิดแล้วให้คอมพิวเตอร์ทำ





ภาษาคอมพิวเตอร์



ภาษา Python

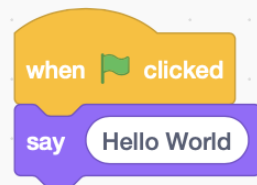


ข้อดีของภาษาไพทอน

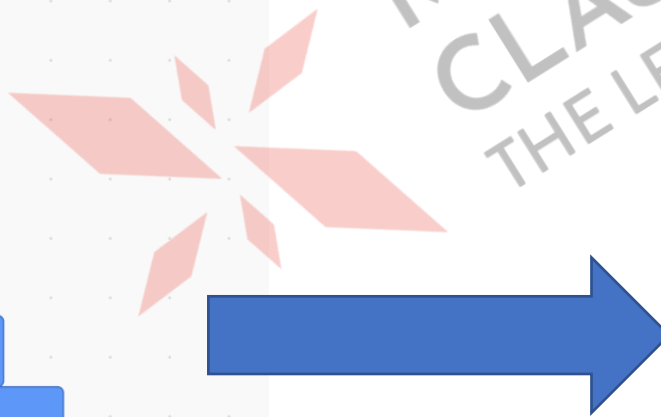
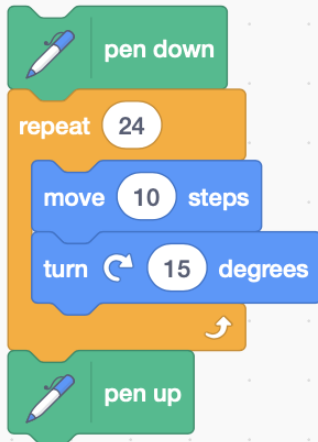
- เขียนได้ทั้งแบบโครงสร้างและโปรแกรมเชิงวัตถุ
- โครงสร้างง่าย ไม่ต้องกำหนดชนิดของข้อมูลล่วงหน้า
- ใช้ได้หลายระบบปฏิบัติการ
- มีไลบรารีกราฟิกติดมาด้วย
- มีไลบรารีจำนวนมาก เพิ่มไลบรารีพิเศษได้

เช่น จะเชื่อมต่อกับ MySQL เพิ่ม PBM เข้าไป

Scratch and Python



```
print('Hello World')
```

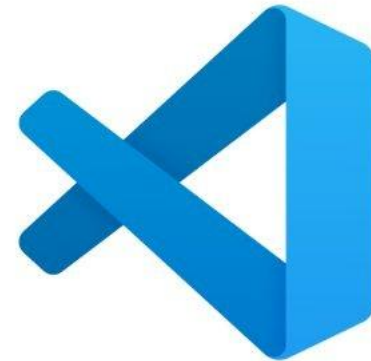


```
from turtle import *  
pendown()  
for n in range(24):  
    forward(10)  
    right(15)  
penup()
```

โฟตอนเป็นภาษาระดับสูง
การจัดการในเชิงลึกต้องอาศัยภาษาอื่น ๆ ผ่านการเรียกใช้โมดูล

from ชื่อโมดูล import คำสั่ง , กลุ่มคำสั่ง

เครื่องมือสำหรับเขียนโปรแกรม



KMITL
MASTER
CLASS
THE LEGEND 2021

การทดสอบโปรแกรม ทำได้ 2

- เขียนคำสั่งด้วย Python Shell
- เขียนคำสั่งในรูปแบบไฟล์ไป

สิ่งที่มีใหม่ควรระวัง เช่น
การใช้เครื่องหมาย
0, [], {}

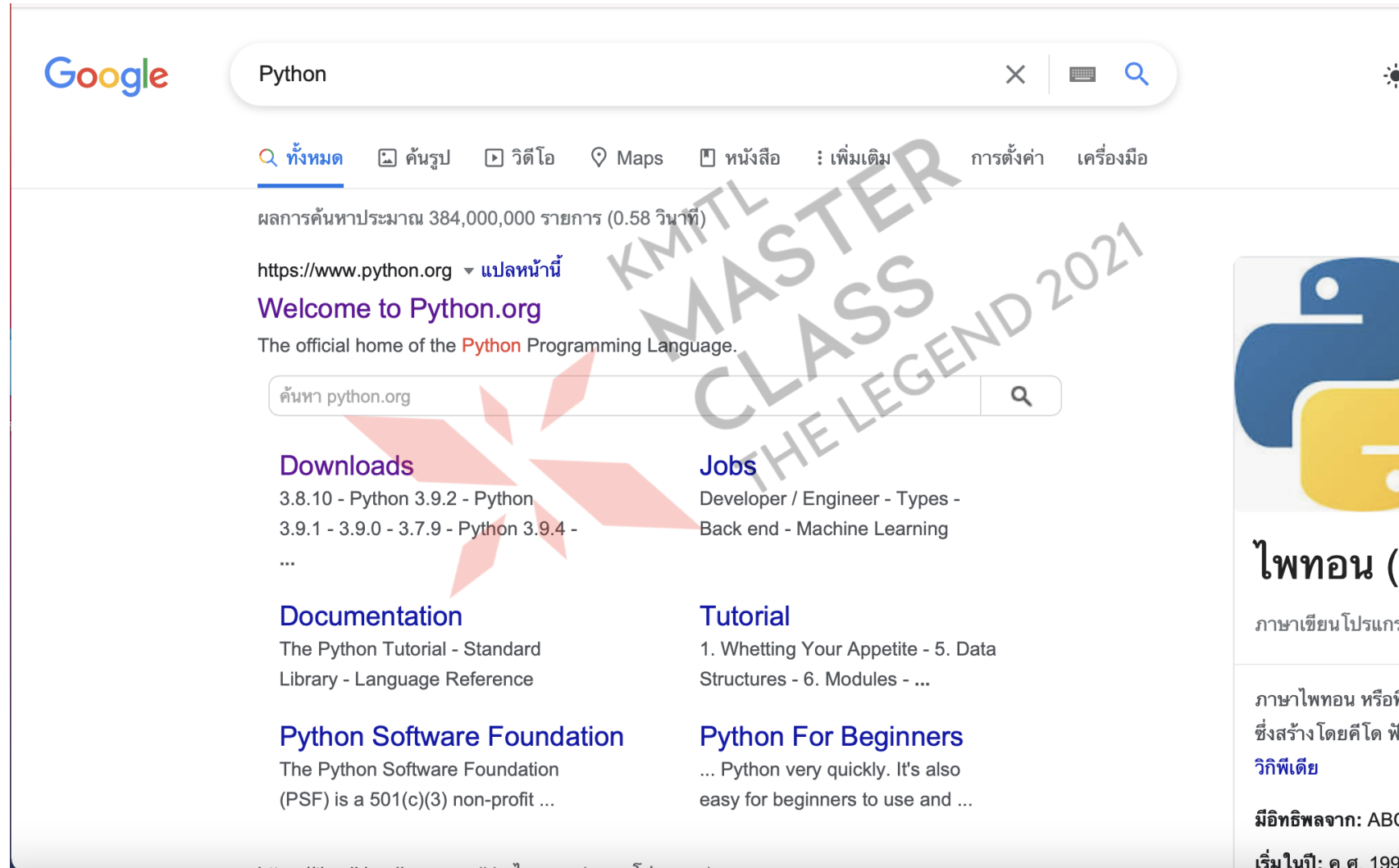
```
Python 3.7.3 Shell
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 16:39:00)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```

พิมพ์คำสั่งบรรทัด

MITL
MASTER
CLASS
THE LEGEND 2021

Ln: 4 Col: 4

เริ่มเขียน Python



Google Python

🔍 ทั้งหมด 🖼️ ค้นรูป 📺 วิดีโอ 📍 Maps 📖 หนังสือ : เพิ่มเติม การตั้งค่า เครื่องมือ

ผลการค้นหาประมาณ 384,000,000 รายการ (0.58 วินาที)

<https://www.python.org> ▾ [แปลหน้านี้](#)

Welcome to Python.org
The official home of the Python Programming Language.

ค้นหา python.org 🔍

Downloads 3.8.10 - Python 3.9.2 - Python 3.9.1 - 3.9.0 - 3.7.9 - Python 3.9.4 - ...	Jobs Developer / Engineer - Types - Back end - Machine Learning
Documentation The Python Tutorial - Standard Library - Language Reference	Tutorial 1. Whetting Your Appetite - 5. Data Structures - 6. Modules - ...
Python Software Foundation The Python Software Foundation (PSF) is a 501(c)(3) non-profit ...	Python For Beginners ... Python very quickly. It's also easy for beginners to use and ...

ไพทอน (Python)
ภาษาเขียนโปรแกรม

ภาษาไพทอน หรือที่
ซึ่งสร้างโดยคีโด ฟัน
[วิกิพีเดีย](#)

มีอิทธิพลจาก: ABC,
เริ่มในปี: ค.ศ. 1990

เขียน Python บนเว็บ

Google replit

ทั้งหมด ค้นรูป วิดีโอ ข่าวสาร Maps :เพิ่มเติม การตั้งค่า เครื่องมือ

ผลการค้นหาประมาณ 1,160,000 รายการ (0.44 วินาที)

<https://replit.com>

Replit: The collaborative browser based IDE
Replit is a simple yet powerful online IDE, Editor, Compiler, Interpreter, and REPL. Code, compile, run, and host in 50+ programming languages.

Python online editor, IDE ...
Write and run Python code using our Python online compiler ...

Log In
Replit is a simple yet powerful online IDE, Editor, Compiler ...

Sign Up
Replit is a simple yet powerful online IDE, Editor, Compiler ...

[ผลการค้นหาเพิ่มเติมจาก replit.com »](#)

Blog
Replit Apps - What is Blubbr? - Boosts - ...

C++
C++ online editor, IDE, compiler, interpreter, and REPL. Code ...

C online editor, IDE, compiler ...
C online editor, IDE, compiler, interpreter, and REPL. Code ...

repl.it
บริษัท

แปลจากภาษาอังกฤษ
ในซานฟรานซิสโกและ
REPL ซึ่งย่อมาจาก
Masad, Faris Masad
ในปี 2559 [วิกิพีเดีย](#)

ดูคำอธิบายต้นฉบับ

ก่อตั้ง: พ.ศ. 2559
ผู้ก่อตั้ง: Amjad Masad

วิดีโอ



Untitled1.ipynb ☆

Comment Share Settings Profile (T)

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM [Progress Bar] Disk [Progress Bar] Editing [Up Arrow]

Navigation icons: Up, Down, Link, Comment, Settings, Copy, Delete, More

```
for i in range(10):  
    print(i,end='')
```



✓ 0s completed at 8:59 PM



ทดลองเขียนโปรแกรม



MITL
MASTER
CLASS
THE LEGEND 2021

กิจกรรมที่ 1 (ต่อ)

2. ลองเขียนโปรแกรมดังต่อไปนี้ แล้วทดสอบการทำงาน

2.1 `x = 2`

`y = 3`

`print('x + y = ',x+y)`

`print('x * y = ',x*y)`

`print('y / x = ',y/x)`

2.2 `x = input("INPUT X : ")`

`y = input("INPUT Y : ")`

`z = x + y`

`print('z = ',z)`

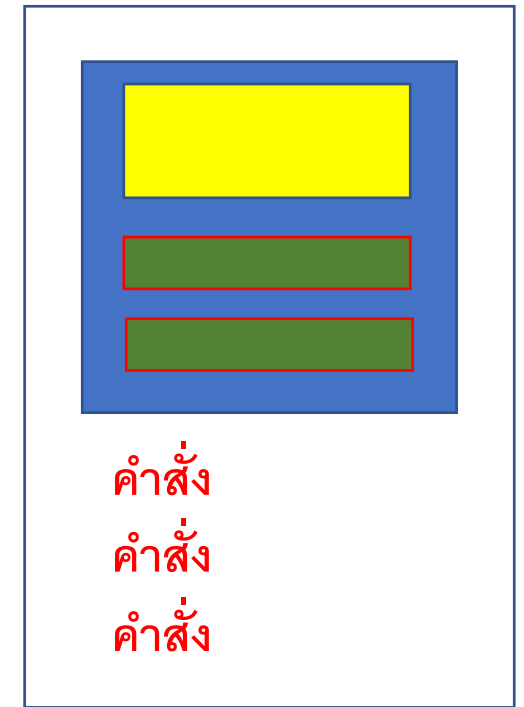
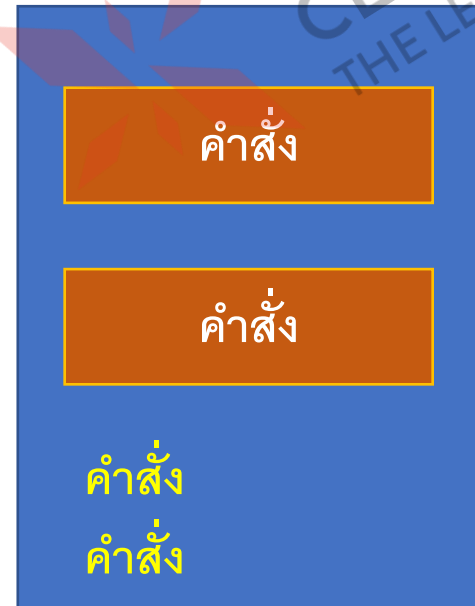
KMITL
MASTER
CLASS
THE LEGEND 2021

รูปแบบการเขียนโปรแกรม

แบบลำดับ

การเขียนโปรแกรมแบบโครงสร้าง

การเขียนโปรแกรมแบบ OOP



การประมวลผลกับตัวเลข

การคำนวณอย่างง่าย

```
>>> x = 6
>>> y = x * 7
>>> print(y)
42
```

ลองเปลี่ยนค่า

```
>>> x = 30
>>> print(y)
42
```

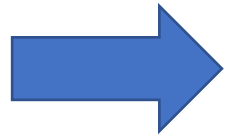
ตรวจสอบข้อมูล

```
>>> x = 10
>>> y = x * 7
>>> print(y)
70
```

สัญลักษณ์	การดำเนินการ
+	add
-	subtract
*	multiply
/	divide

Program flow

INPUT



PROCESSING



OUTPUT



Input command

Keyboard.

Mouse



Variables

Maths

Loops

Function



Print command

Screen

Graphics

KMITL
MASTER
CLASS
THE LEGEND 2021

นิพจน์ (Expressions)

- ตัวดำเนินการทางคณิตศาสตร์

$$Z = (A + B) / C + (B * D)$$

สัญลักษณ์	ความหมาย	ตัวอย่าง
+	บวก (Addition)	$C = A + B$
-	ลบ (Subtraction)	$C = A - B$
*	คูณ (Multiplication)	$C = A * B$
/	หาร (Division)	$C = A / B$
%	หารเอาเศษ (Modulo)	$C = A \% B$
**	ยกกำลัง (Exponent)	$C = A ** B$
//	หารปัดเศษ (Floor Division)	$C = A // B$

ตัวอย่าง

- $A = 3$

$$B = 2$$

$$C = A * B + 5 \quad - > \quad 3 * 2 + 5 = 11$$

$$C = (A+1) * 3 + 2 \quad - > \quad (3 + 1) * 3 + 2 = 14$$

$$C = (2 * A + 1) + 3 * (A + B) \quad - > \quad (2 * 3 + 1) + 3 * (3 + 2) \quad - > \quad (6 + 1) + 3 * 5 = 22$$

คำสั่งรับและแสดงผลข้อมูล

- ฟังก์ชันรับข้อมูล

`input()`

EX. `msg = input("ป้อนข้อมูลครับ : ")`

`x = input()`

- ฟังก์ชันแสดงผลข้อมูล

`print()`

EX. `print("ข้อมูลที่ป้อนเข้ามาคือ ",msg)`



* การเขียนโปรแกรม ภาษา PYTHON



KMITL
MASTER
CLASS
THE LEGEND 2021

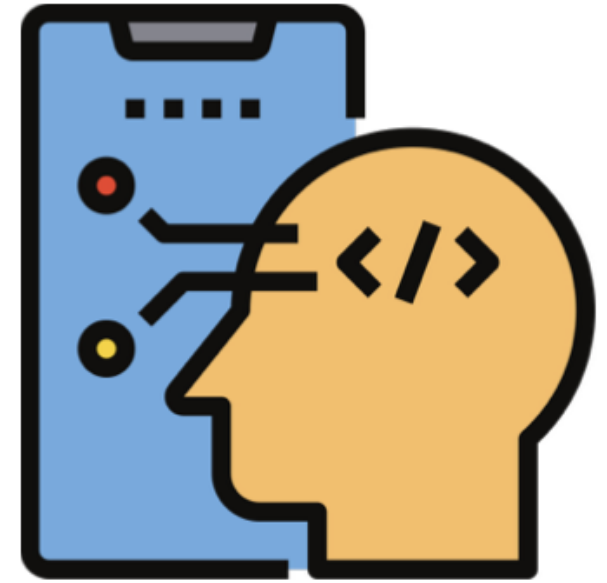
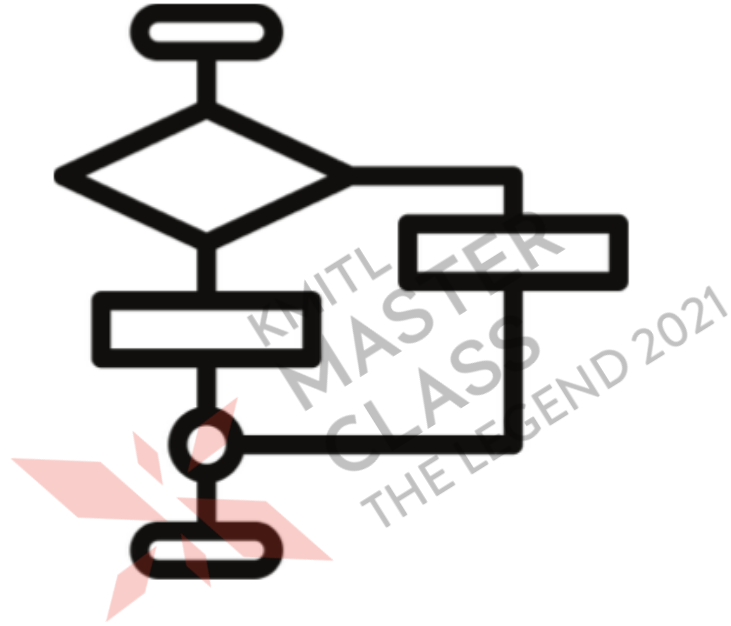
ตอนที่ 2

เขียนโปรแกรม แก้ปัญหา



เขียนโปรแกรมเพื่อ
แก้โจทย์ปัญหา

KMITL
MASTER
CLASS
THE LEGEND 2021



เข้าใจงาน. แล้วคิด

ขั้นตอนการแก้ปัญหา

Steps in Problem Solving

0
1

Identification of problem
and opportunities

0
2

Definition of goals

0
3

Exploration of
possible strategies

0
4

Anticipation of
outcome and action

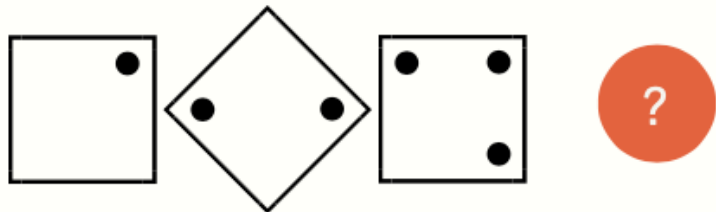
0
5

Leaning through
retrospection

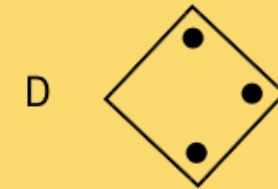
KMITL
MASTER
CLASS
THE LEGEND 2021

1. การระบุปัญหา

มองปัญหาให้ชัดเจน



มองให้ออกว่าต้องการทำอะไร



KMITL
MASTER
CLASS
THE LEGEND 2021



2. มองความหมาย ของปัญหา

อินพุต คืออะไร
ประมวลผล อย่างไร
เอาต์พุต คืออะไร รูปแบบใด



ตัวอย่างการคำนวณ

ข้อมูล



ผลลัพธ์



KMITL
MASTER
CLASS
THE LEGEND 2021

3. สำรองความเป็นไปได้

หาแนวทางการแก้ปัญหา
พิจารณาปัญหาและข้อจำกัด
คำนึงถึงกลยุทธ์ที่นำมาใช้



หาวิธีการคิดให้ได้ว่าจะทำ
อย่างไร



EXPLORE



4. คาดการณ์ ผลลัพธ์ และ ดำเนินการ

วิเคราะห์ผลลัพธ์ที่เป็นไปได้ทุกประการ
ปรับเปลี่ยนวิธีการให้เหมาะสมได้
วิธีการที่ไม่เหมาะสมให้เปลี่ยน

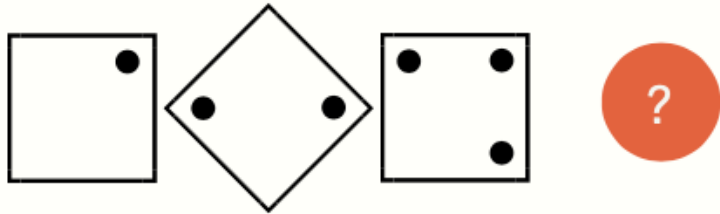
มองแล้วคิดทั้งผิดและถูก

KMITL
MASTER
CLASS
THE LEGEND 2021



5. ทดสอบปรับปรุง

หาวิธีการที่ดีที่สุด
หาวิธีการที่จะนำมาใช้ในอนาคต



ตรวจสอบว่าถูกหรือไม่
ลองป้อนค่าเข้าไป
หาวิธีการใหม่ที่ดีกว่า

KMITL
MASTER
CLASS
THE LEGEND 2021



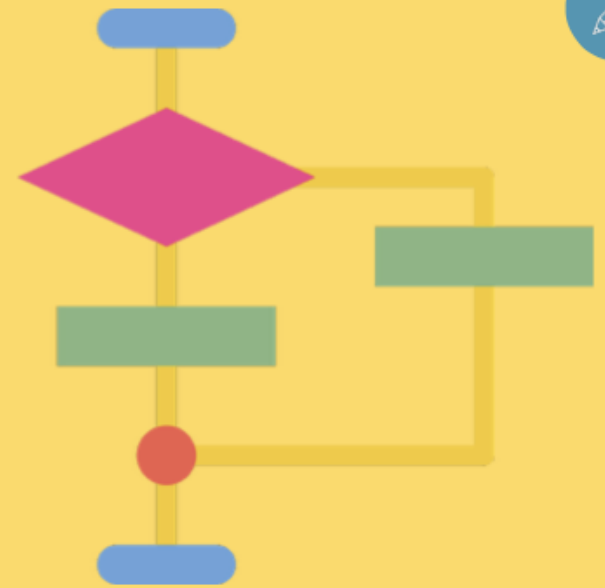
Algorithms

ขั้นตอนการแก้ปัญหา

เริ่มต้น

1. รับข้อมูล
2. ประมวลผล
 - 2.1
 - 2.2
3. แสดงผล

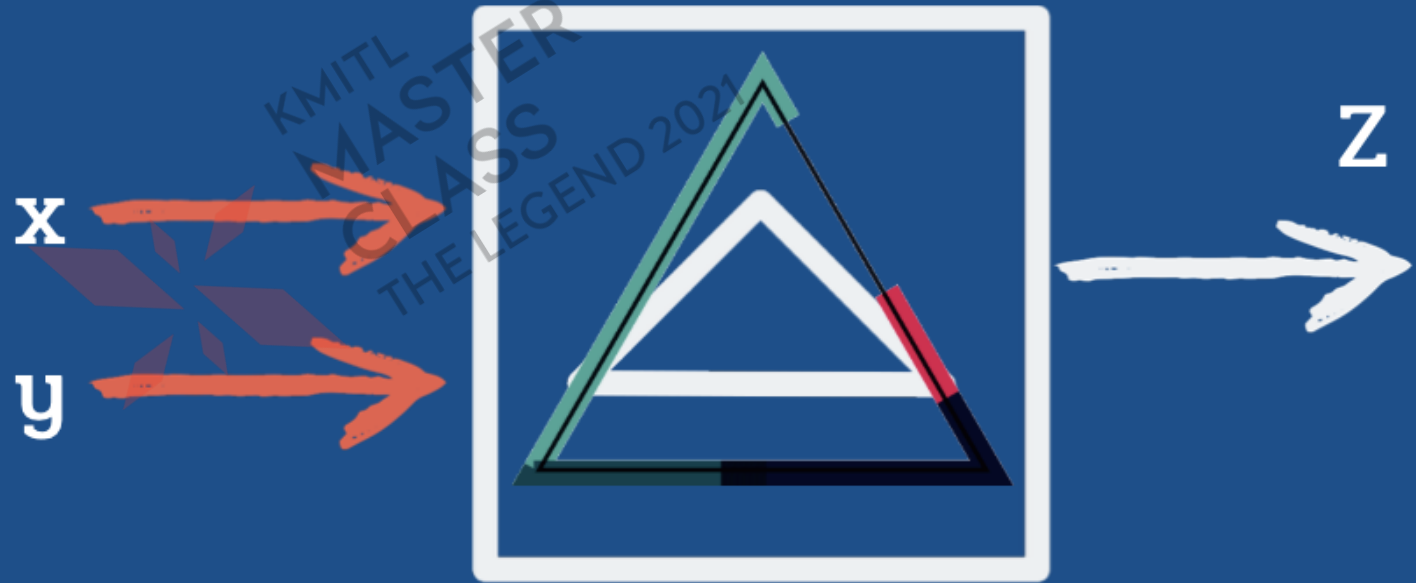
สิ้นสุด



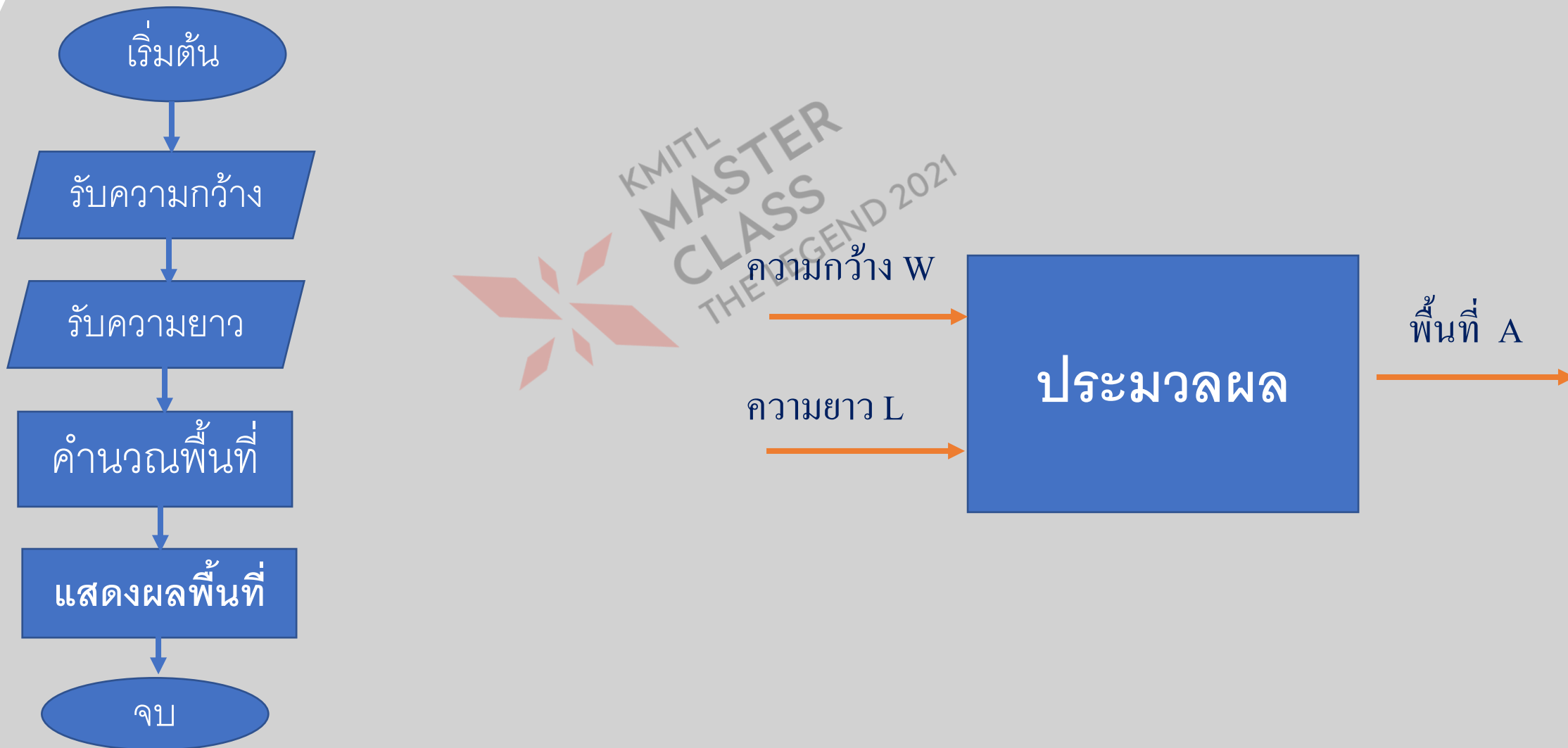
KMITL
MASTER
CLASS
THE LEGEND 2021

การคำนวณพื้นที่

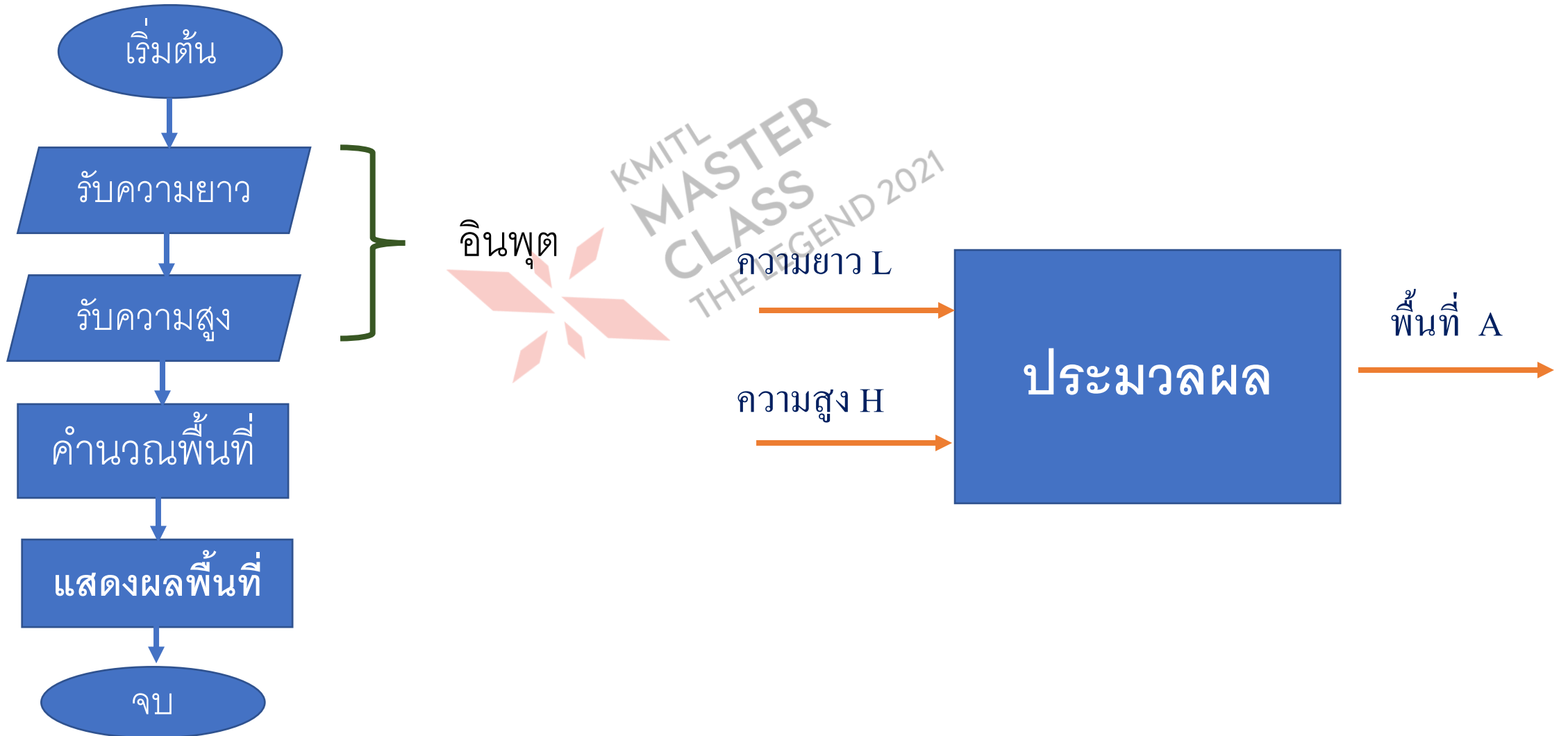
ต้องมีความรู้อื่นด้วย



การคำนวณพื้นที่สี่เหลี่ยม



การคำนวณพื้นที่สามเหลี่ยม



ตัวแปร ชนิดข้อมูล



แทนตำแหน่งหน่วยความจำ

การประกาศตัวแปร

- เป็นตัวอักษร ตัวพิมพ์เล็กพิมพ์ใหญ่จะต่างกัน
- เริ่มต้นด้วยตัวอักษรต้องไม่ซ้ำกับคำสั่ง
- ไม่มีช่องว่าง ใช้เครื่องหมาย underscore (_) คั่น

```
msg = "TEERAWAT PRAKOBPHON"
```

```
num1 = 2342
```

```
Temp = 37.5
```

```
Pi = 3.141
```

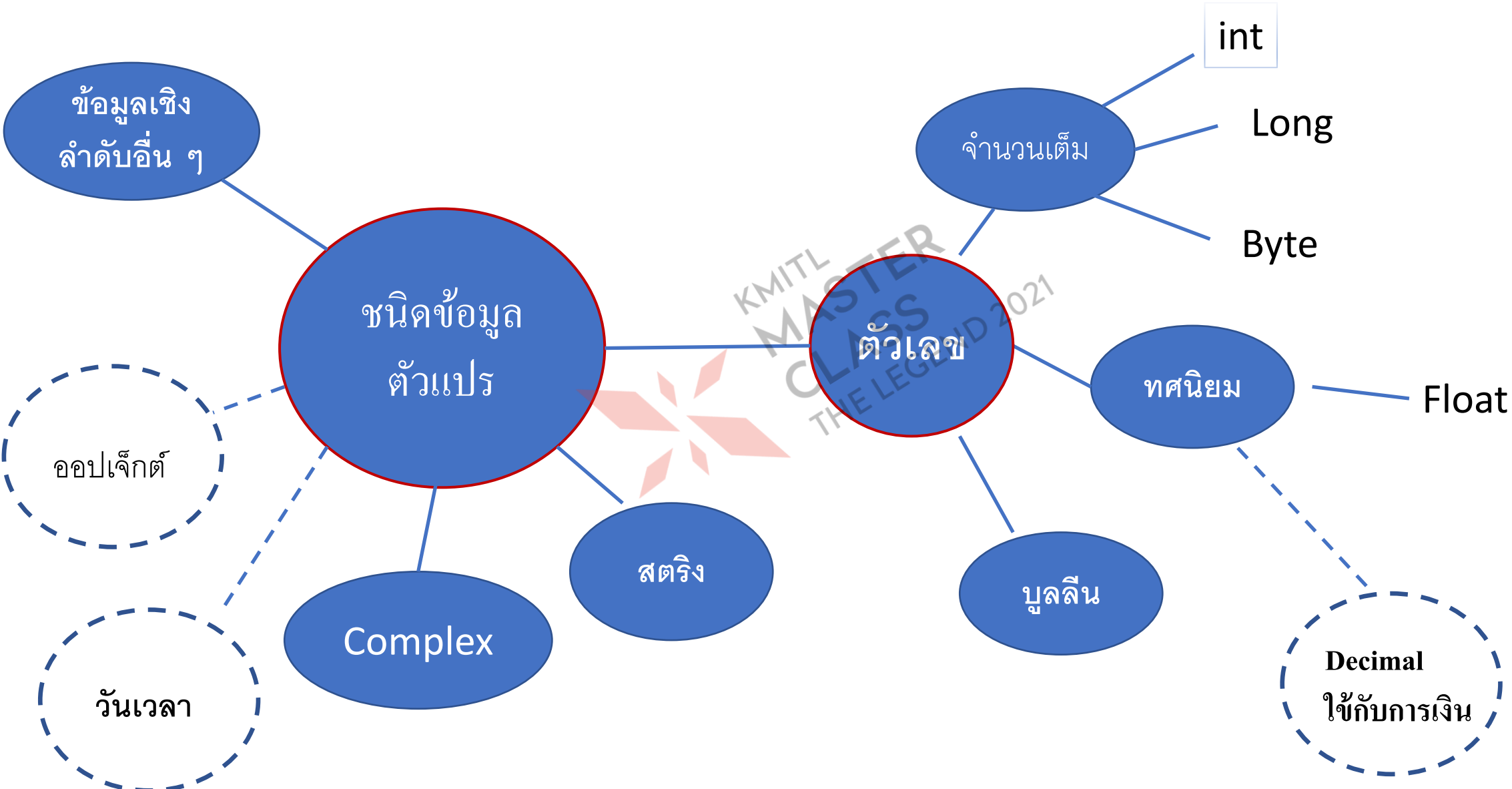
```
X = 0b01000001
```

ชนิดของข้อมูล

- ค่าคงที่ - $\rightarrow \text{Pi} = 3.141$
- ข้อมูลจำนวนเต็ม
- ข้อมูลจำนวนทศนิยม
- ข้อมูลเลขเชิงซ้อน
- ข้อมูลสตริง
- ข้อมูลตรรกะ

KMITL
MASTER
CLASS
THE LEGEND 2021

ชนิดของตัวแปร



บอกผลลัพธ์จากการประมวลผลต่อไปนี้

```
>>> A = 3
```

```
>>> B = 2
```

```
>>> C = (A + B*2) + (2*A + B*3 + 2)
```

```
>>> print(C)
```

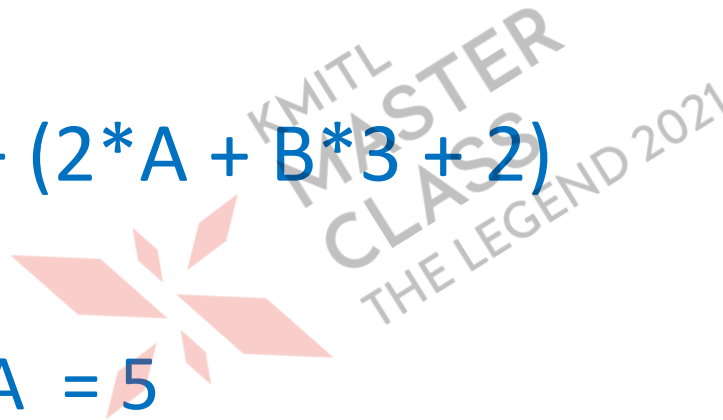
```
>>> A = 5
```

```
>>> B = 3
```

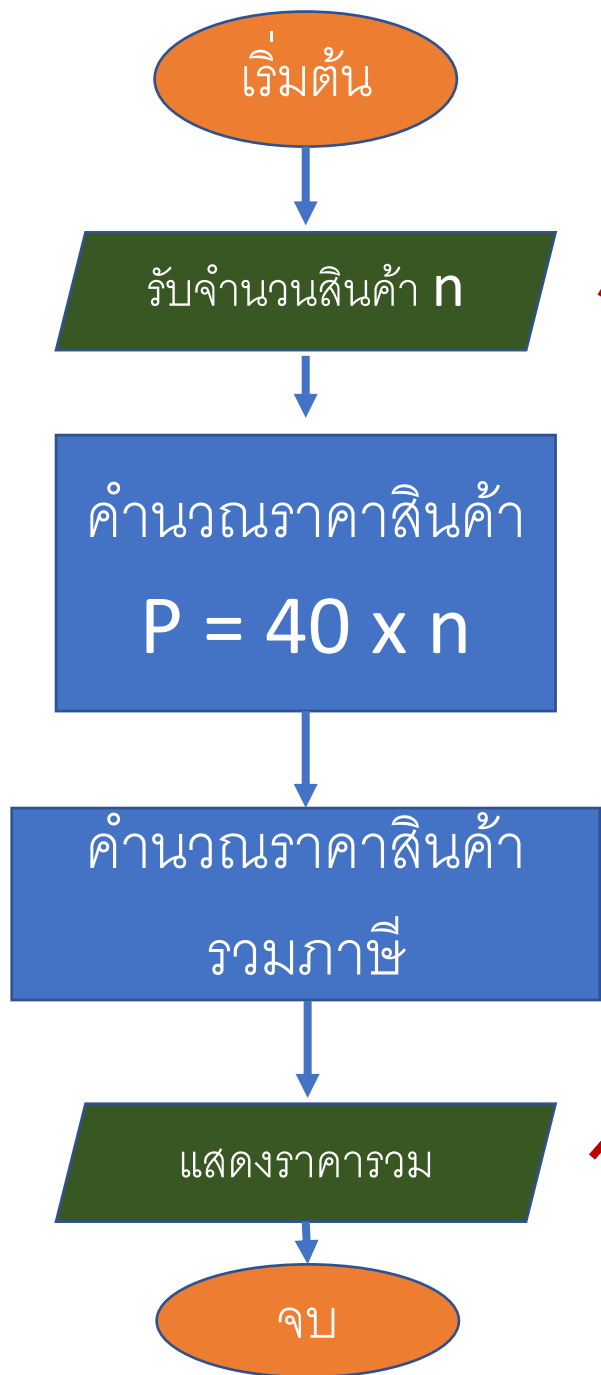
```
>>> C = A/B
```

```
>>> C = A%B
```

```
>>> C = A//B
```



การคำนวณราคาสินค้า



```
n = int(input("ป้อนจำนวนสินค้า: "))  
p = n*40  
total = p + p*(7/100)  
print("ราคาสินค้ารวม คือ :",total)
```

โจทย์

เครื่องกดเงิน ATM เครื่องหนึ่งบรรจุแบงค์ 1000
แบงค์ 500 และแบงค์ 100

ถ้าหากกดเงิน 4800 บาท จะมีแบงค์ออกมาดังนี้

แบงค์ 1000 = 4

แบงค์ 500 = 1

แบงค์ 100 = 3

จงเขียนโปรแกรมสำหรับเครื่อง ATM โดยรับจำนวนเงินเข้าไป
แล้วให้เครื่องแจ้งออกมาว่ามีแบงค์แต่ละชนิดจำนวนเท่าใด



วิธีคิด

- อินพุต เงินที่จะถอน
- ประมวลผล หาจำนวน
- เอาต์พุต จำนวนแบงค์

เริ่มต้น

1. รับจำนวนเงินที่จะถอน X
2. B1000 เท่ากับ X หาร 1000
3. B500 เท่ากับเศษที่เหลือ หารด้วย 500
4. B100 เท่ากับเศษที่เหลือ หารด้วย 100
5. แสดงผลจำนวนแบงค์

จบ

กิจกรรมที่ 2

1. จงหาค่าของการคำนวณต่อไปนี้

$$Z = 3 * A + (2 * B / 3 + 4 * 2) * 2$$

2. เขียนอัลกอริทึมและโปรแกรมคำนวณพื้นที่วงกลม

3. เขียนอัลกอริทึมและโปรแกรมคำนวณหาค่าเฉลี่ยของตัวเลข 4 ค่า



แบบฝึกหัด

- จงเขียนอัลกอริทึม และโปรแกรมสำหรับคำนวณราคาสินค้ารวม ภาษีมูลค่าเพิ่ม 7% โดยมีสินค้า 2 รายการ ให้ป้อนราคาและจำนวนของสินค้าแต่ละรายการ จากนั้นให้โปรแกรมแสดงราคารวมภาษีออกมา
- อินพุต
- ประมวลผล
- เอาต์พุต



KMITL
MASTER
CLASS
THE LEGEND 2021

STRING

```
>>> st1 = "Python"
```

```
>>> st2 = "Playing with "
```

```
>>> st3 = "JAVA"
```

```
>>> st4 = st2
```



KMITL
MASTER
CLASS
THE LEGEND 2021

```
>>> st4 = st2+st1
```

```
>>> print(st4)
```

```
Playing with Python
```

```
>>> print(st2 + st3)
```

```
Playing with JAVA
```

```
>>>
```



ข้อมูลแบบ ลิสต์ และทูเพิล

ธีรวัฒน์ ประกอบผล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ข้อมูลแบบลิสต์ (List)

ข้อมูลเก็บเป็นโครงสร้างมีลักษณะเป็นช่อง
ใช้ตัวชี้ หรือ index ในการอ้างถึงข้อมูล

การประกาศตัวแปร

ชื่อตัวแปร = [ข้อมูล 1, ข้อมูล 2, ..., ข้อมูล n]

ตัวอย่าง

data1 = [1, 3, 6, 9]

data2 = [2, 3.25, "TEE", 8, 4]

data3 = [25]

ตัวอย่างการดำเนินการกับลิสต์

```
>>> a = [1,2,3]
```

```
>>> b = [4,5]
```

การต่อลิสต์

```
>>> a+b
```

```
[1, 2, 3, 4, 5]
```

การทำสำเนาแล้วต่อลิสต์

```
>>> 3*a+b
```

```
[1, 2, 3, 1, 2, 3, 1, 2, 3, 4, 5]
```

```
>>> c = 10*[3]
```

```
>>> c
```

```
[3, 3, 3, 3, 3, 3, 3, 3, 3, 3]
```

```
>>> |
```

```
>>> a = [2,4,6,8,10]
```

```
>>> a[0]
```

```
2
```

```
>>> a[3]
```

```
8
```

```
>>> |
```



การทำอินเด็กซิง (indexing)

ตัวอย่าง

สร้างลิสต์ของตัวเลข

```
>>> a = list(range(1,20,2))
```

```
>>> a
```

```
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19]
```

```
>>> a[0:4]
```

```
[1, 3, 5, 7]
```

```
>>> a[3:5]
```

```
[7, 9]
```

```
>>>
```

การทำ สไลซิง (slicing)

KMITL
MASTER
CLASS
THE LEGEND 2021

ข้อมูลแบบทูเพิล (Tuple)

ข้อมูลเก็บเป็นโครงสร้างเหมือนลิสต์
แต่แก้ไขข้อมูลไม่ได้

การประกาศตัวแปร

ชื่อตัวแปร = (ข้อมูล 1, ข้อมูล 2, ..., ข้อมูล n)

ตัวอย่าง

```
data = ('Name', 'Address', age, 'ID')  
Book = ('Java', 'C++', 'Python')  
data3 = (25)
```

การประมวลผลข้อมูลในลิสต์และทูเพิล

• ดำเนินการผ่านเมธอด

เมธอด	การประมวลผล
append()	เพิ่มข้อมูลต่อท้ายลิสต์
clear()	ใช้ลบข้อมูลทั้งหมดออกจากลิสต์
count()	ใช้นับจำนวนข้อมูลในลิสต์
index()	ใช้ค้นหาตำแหน่งของข้อมูลที่เก็บในลิสต์
pop()	ลบข้อมูลโดยระบุตำแหน่งที่ต้องการลบ
len()	เป็นฟังก์ชันแสดงข้อมูลที่อยู่ในลิสต์ เช่น len(str)

ตัวอย่าง

data1.append(x)

เพิ่มข้อมูล X ลงไป

data1.clear()

data1.count(x)

X คือข้อมูลที่จะนับ

ทิวเปิด เปลี่ยนแปลงข้อมูลไม่ได้ แต่ลบได้

- ดำเนินการผ่านเมธอด และฟังก์ชัน

เมธอด	การประมวลผล
index()	ใช้แสดงตำแหน่งของข้อมูล
count()	ใช้นับจำนวนข้อมูล
len()	เป็นฟังก์ชันแสดงจำนวนข้อมูล

data.index(x)
data.count(x)
len(data)

เมื่อ **x** คือข้อมูล

ข้อมูลแบบ Dictionaries

```
>>> age = {'Mary':10 , 'Sanjay':8}
```

```
>>> print(age)
```

```
{'Sanjay' : 8, 'Mary' : 10}
```

ใช้คีย์ (**key**) ในการอ้างอิงถึงข้อมูล



ข้อมูลแบบเซต

เก็บข้อมูลที่แต่ละรายการไม่ซ้ำกัน

```
>>> A = {1,2,2,3,4,4,5,6,7,7,8,8,8}
```

```
>>> print(A)
```

```
A = {'A','B','B','C'}
```

```
B = {'A','B','C'}
```

```
print(A==B)
```

```
if 'B' in A:
```

```
    print("Yes")
```



A = {1,2,3,4,6,7}

B = {2,4,6,8}

A.add(9)

print("set A = ",A)

print("set B = ",B)

C = A.intersection(B)

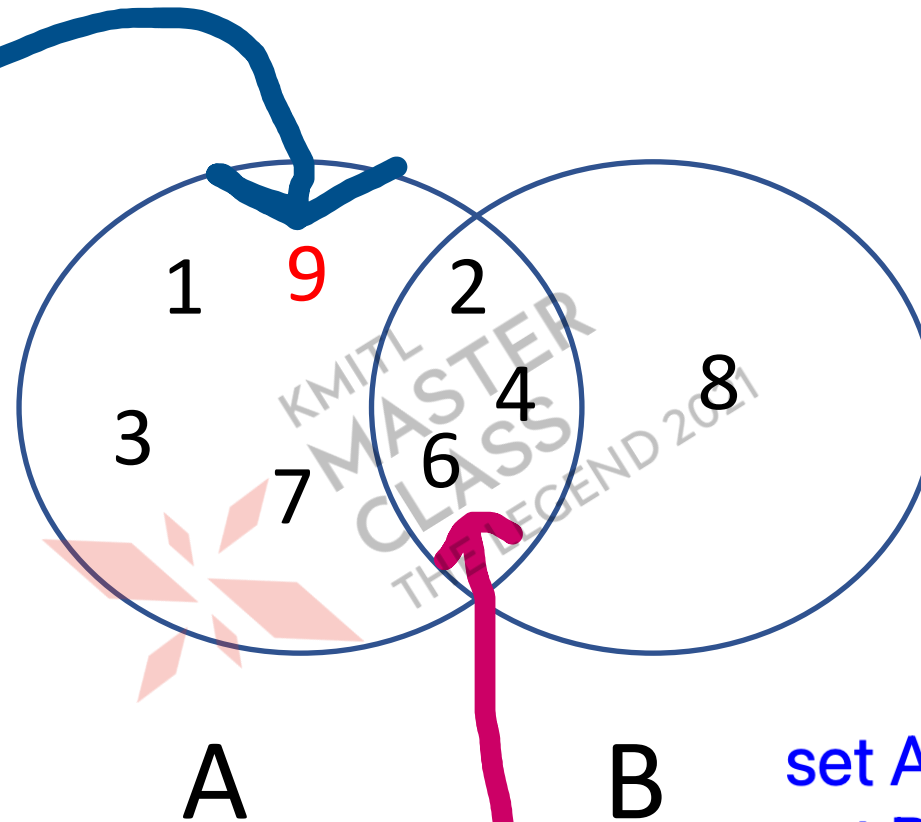
print("A ^ B = ",C)

C = A.union(B)

print("A U B = ",C)

C = A.difference(B)

print("A - B = ",C)



ผลการรันโปรแกรม

set A = {1, 2, 3, 4, 6, 7, 9}

set B = {8, 2, 4, 6}

A ^ B = {2, 4, 6}

A U B = {1, 2, 3, 4, 6, 7, 8, 9}

A - B = {1, 3, 9, 7}



การเขียนโปรแกรมตัดสินใจ

ตัวดำเนินการเปรียบเทียบ (Comparison Operators)

- ให้ผลลัพธ์เป็นจริง กับเท็จ

พื้นฐาน {

สัญลักษณ์	ความหมาย	ตัวอย่าง
>	มากกว่า	$X > Y$
<	น้อยกว่า	$X < Y$
==	เท่ากับ	$X == Y$
!=	ไม่เท่ากับ	$X != Y$
>=	มากกว่าหรือเท่ากับ	$X >= Y$
<=	น้อยกว่าหรือเท่ากับ	$X <= Y$

ตัวดำเนินการทางลอจิก

```
>>> toys = 10
```

```
>>> toys == 1
```

```
False
```

```
>>> toys > 1
```

```
True
```

```
>>> toys < 1
```

```
False
```

```
>>> toys == 9 or toys == 10
```

```
True
```

```
>>> not toys == 1
```

```
True
```

STRING

```
>>> dog = "Woof woof"
```

```
>>> dog == "Woof woof"
```

```
True
```

```
>>> dog == "woof woof"
```

```
False
```

```
>>> dog == "Woof woof "
```

```
False
```



KMITL
MASTER
CLASS
THE LEGEND 2021

ตัวดำเนินการตรรกศาสตร์ (Logical Operators)

- ให้ผลลัพธ์เป็นจริง กับเท็จ

สัญลักษณ์	ความหมาย	ตัวอย่าง
and	การ AND ผลลัพธ์เป็นจริงเมื่อเป็นจริงทั้งคู่	X and Y
or	การ OR ผลลัพธ์เป็นเท็จเมื่อเป็นเท็จทั้งคู่	X or Y
not	กลับค่าทางลอจิก	not X

คำสั่งมีทางเลือก (ตัดสินใจ)

- การเลือกทำแบบทางเดียว
- การเลือกทำสองทิศทาง
- การเลือกทำแบบหลายทิศทาง

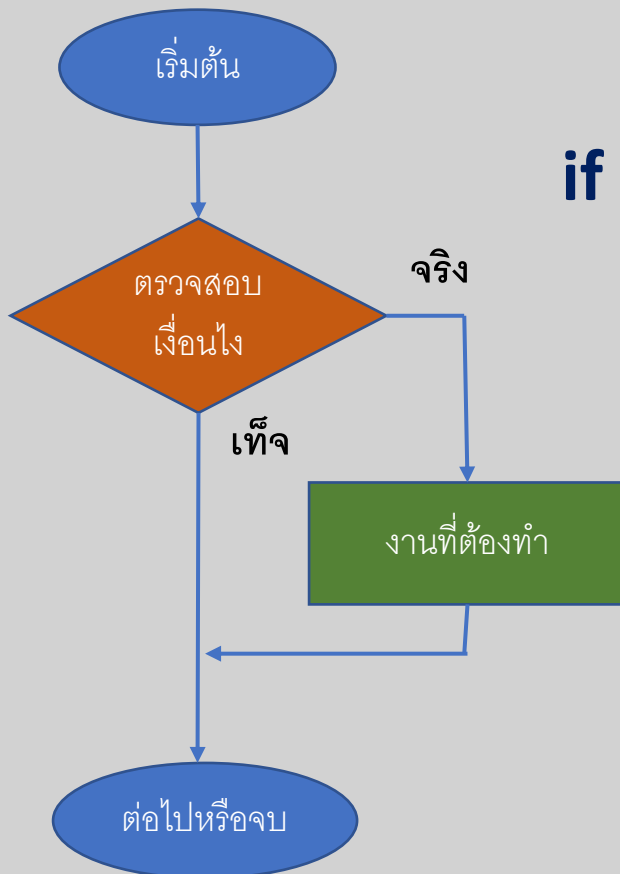


KMITL
MASTER
CLASS
THE LEGEND 2021

คำสั่งตัดสินใจ

- ใช้การตรวจสอบเงื่อนไขในการตัดสินใจ

เลือกทำแบบทางเดียว



if (condition) :
statement()

```
sc = int(input('กรุณาป้อนคะแนนของคุณ = '))
```

```
if (sc >= 50) :
```

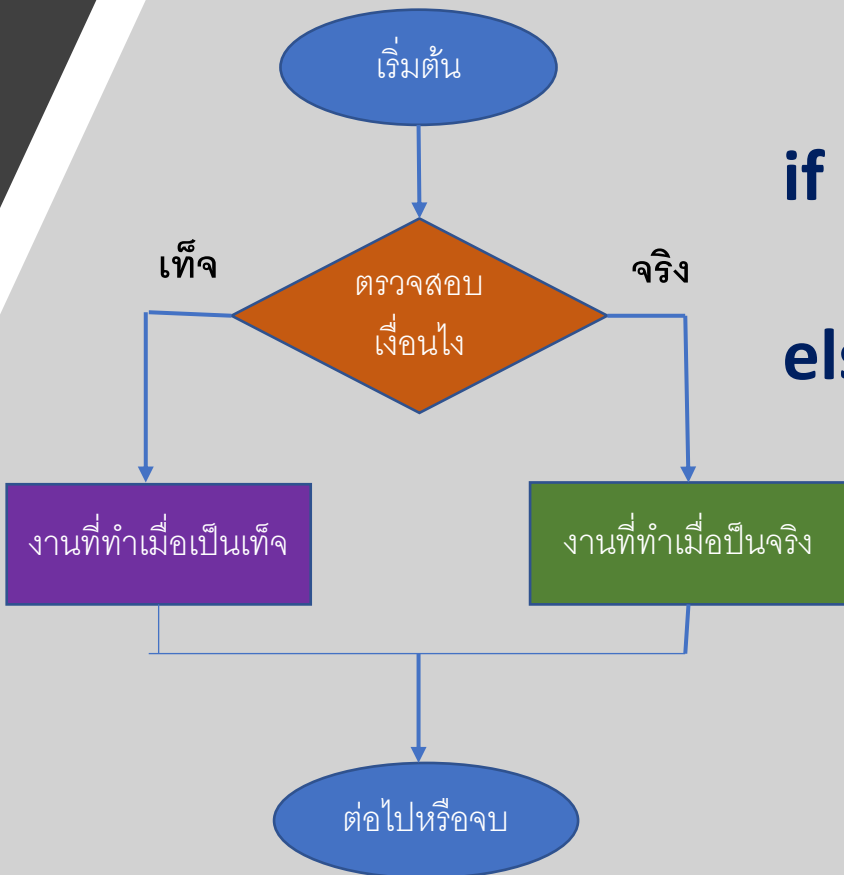
```
    print('คุณสอบผ่าน')
```

```
    print('ให้กำลังใจนะครับ')
```

คำสั่งตัดสินใจ

- ใช้การตรวจสอบเงื่อนไขในการตัดสินใจ

เลือกทำแบบสองทางเลือก

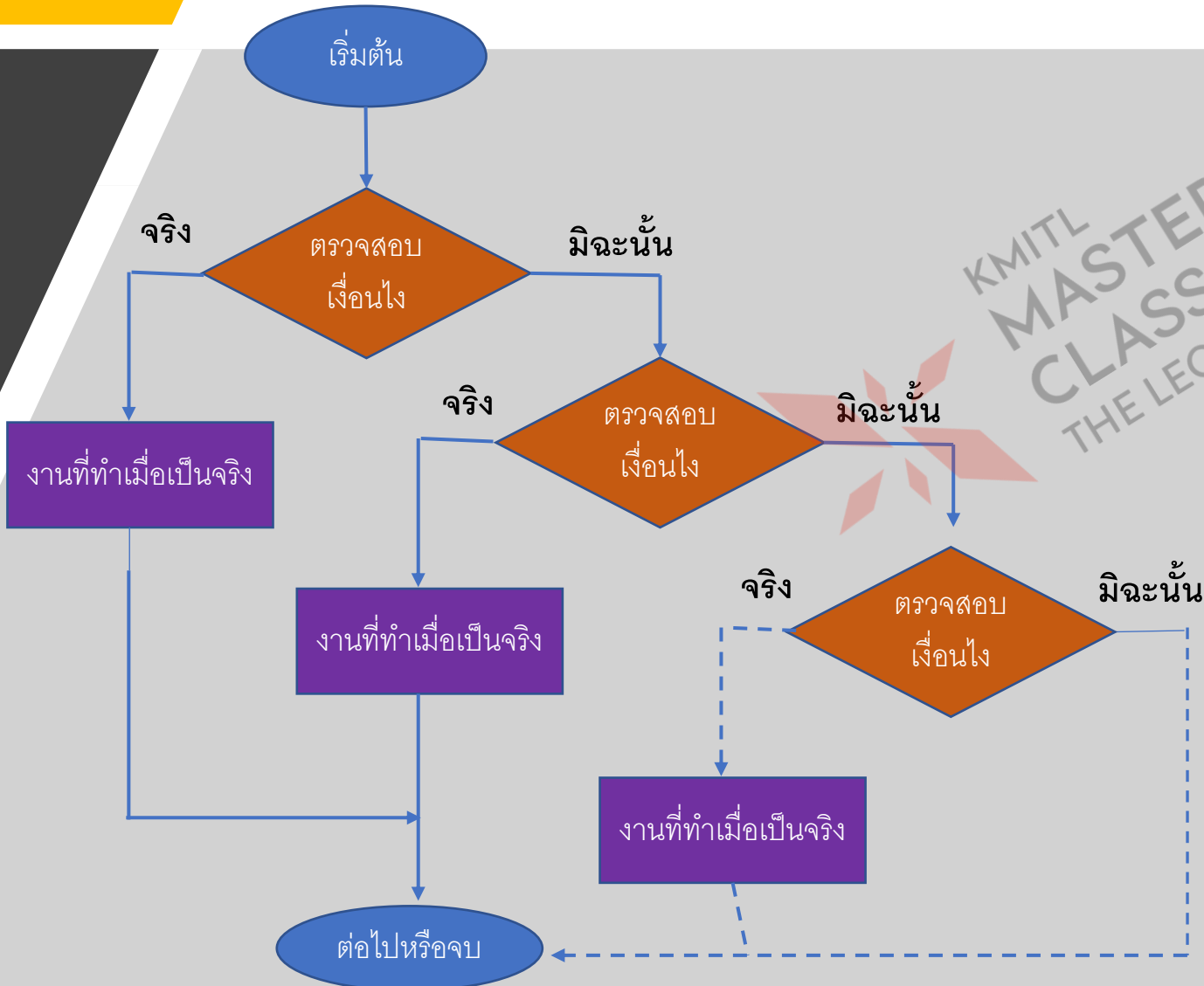


if (condition) :
statement()
else:
statement()

```
sc = int(input('กรุณาป้อนคะแนนของคุณ = '))  
  
if (sc >= 50) :  
    print('คุณสอบผ่าน')  
else:  
    print('คุณสอบไม่ผ่าน')  
    print('ให้กำลังใจนะครับ')
```

คำสั่งตัดสินใจ

เลือกทำแบบหลายทางเลือก



if (condition) :

statement()

elif (condition) :

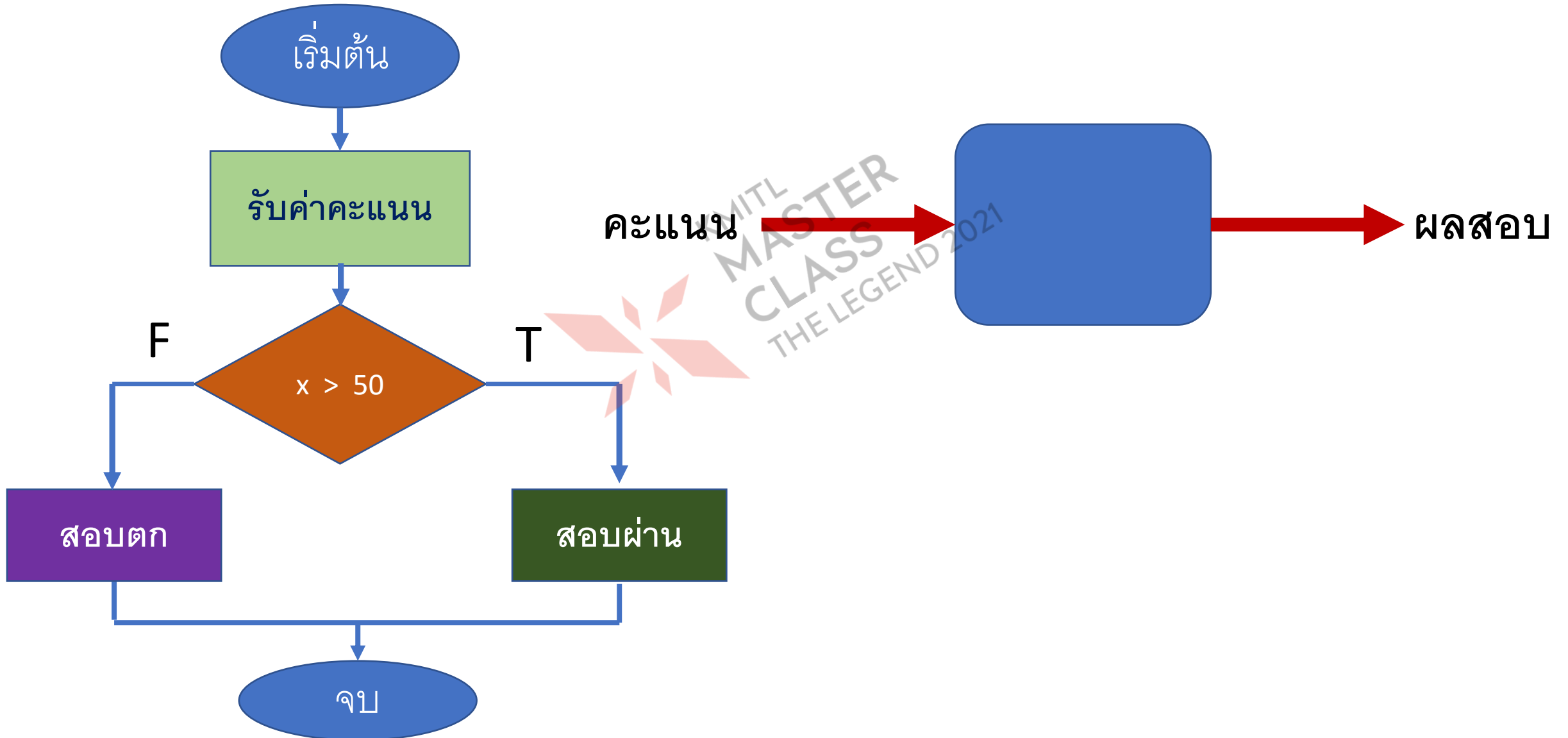
statement() :

elif (condition):

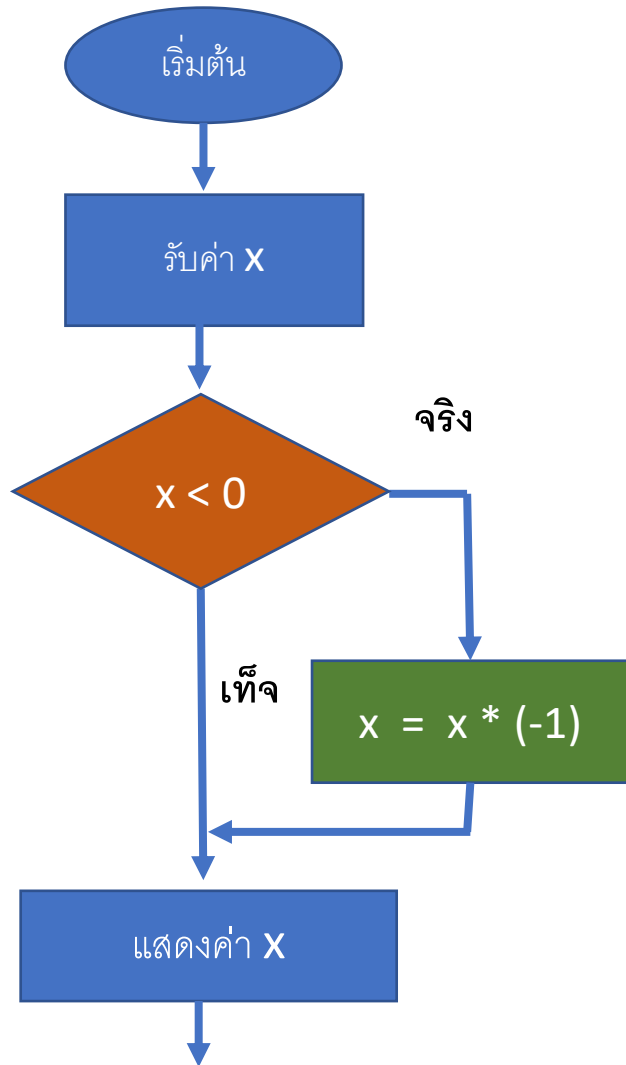
statement()

else:

การทำงานแบบมีทางเลือก

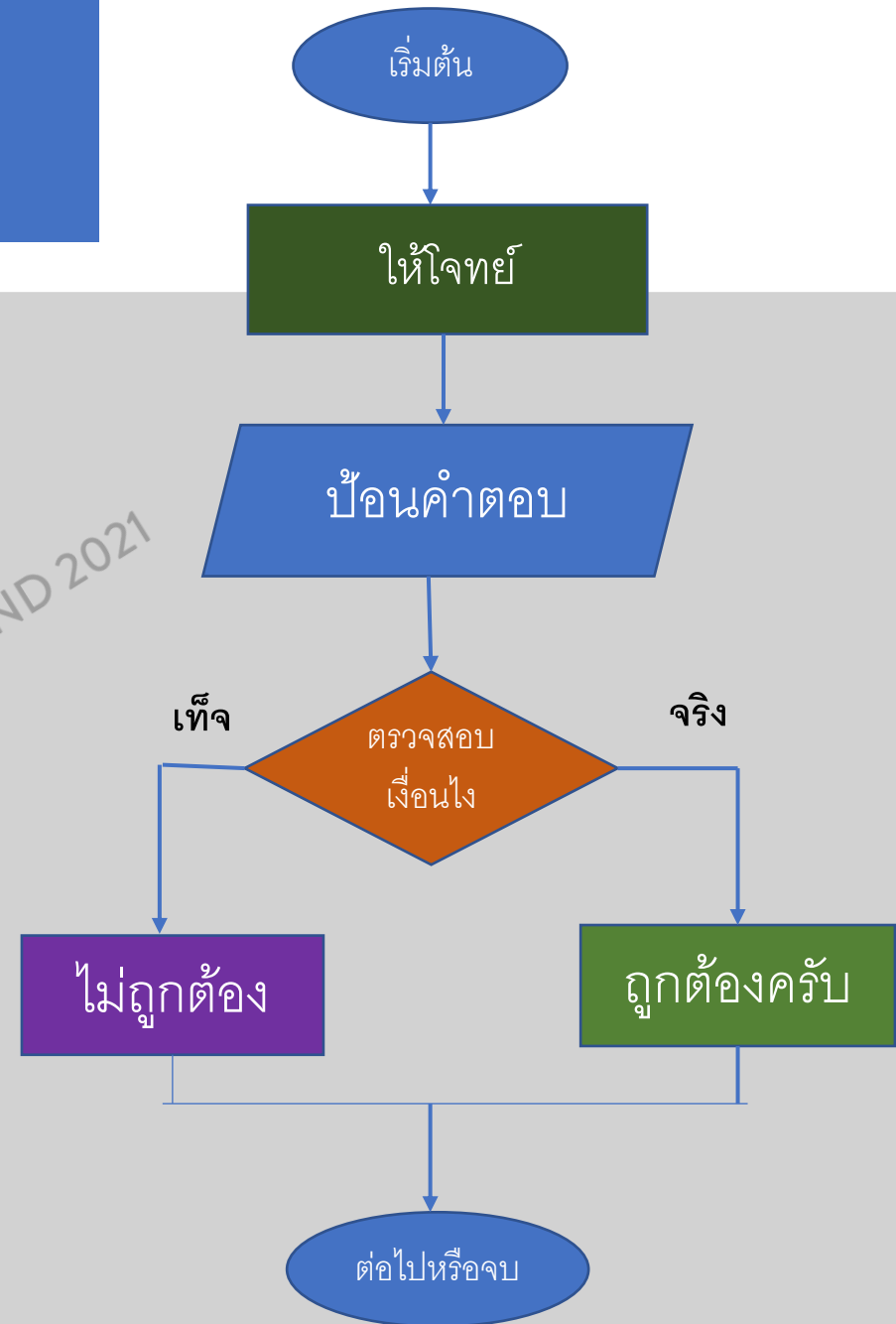
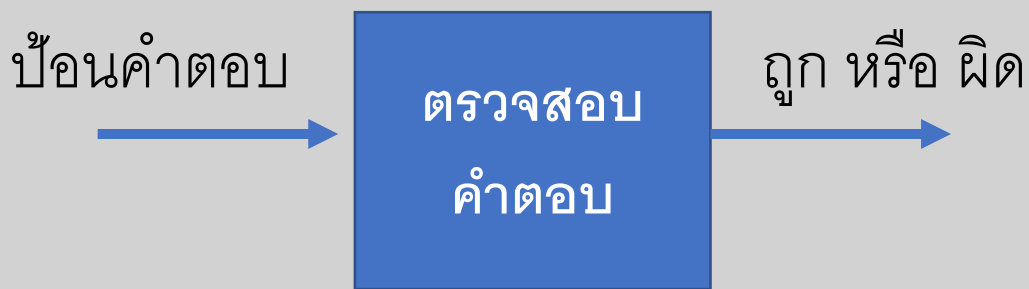


การหาค่าสัมบูรณ์ของตัวเลข



ตัวอย่างโปรแกรมบวกเลข

- ตัวอย่างนี้จะให้โจทย์ แล้วให้ผู้เล่นป้อนคำตอบ

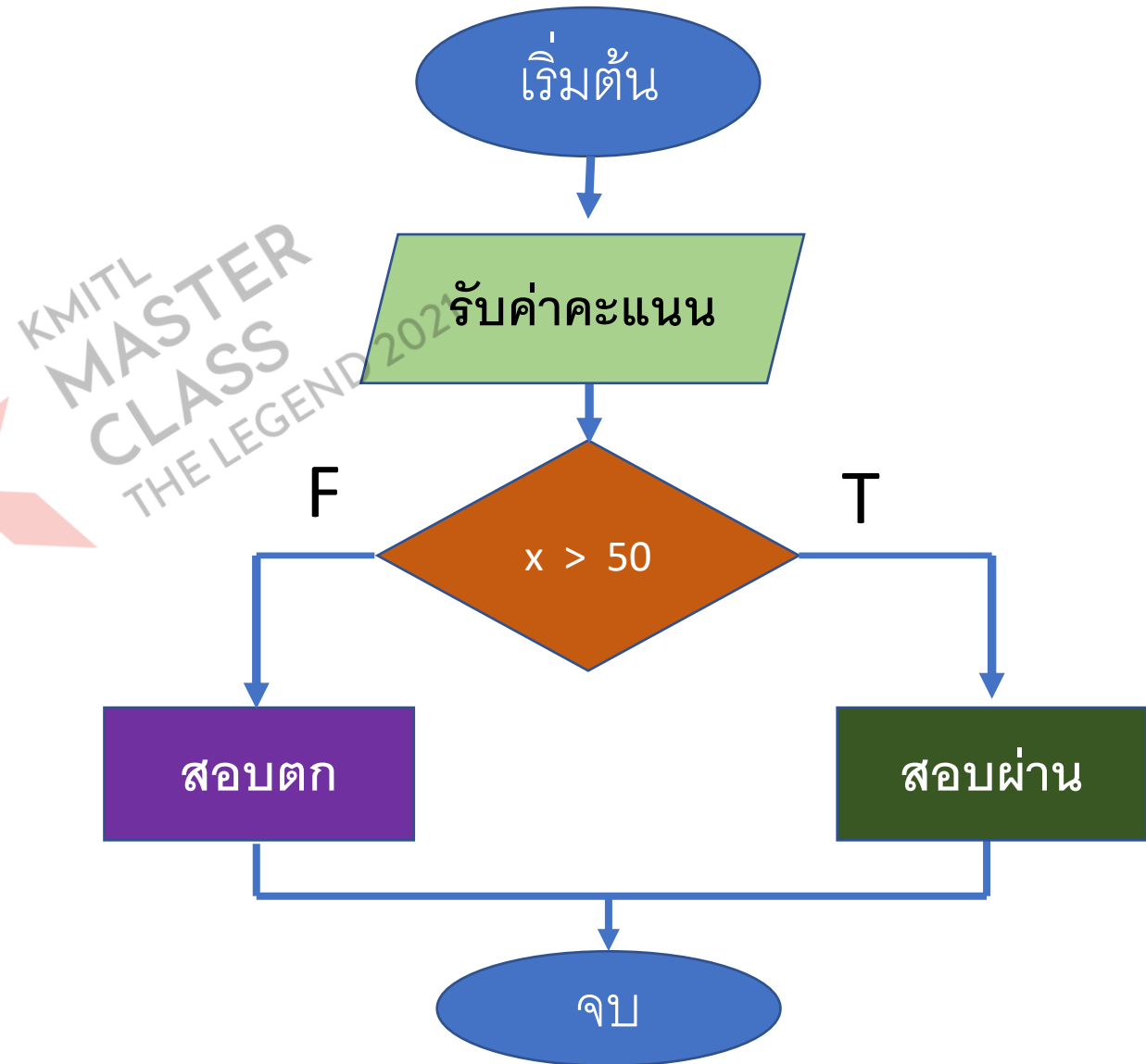


การทำงานแบบมีทางเลือก

เริ่มต้น

1. รับคะแนนสอบกลางภาค
2. รับคะแนนสอบปลายภาค
3. หาผลรวมของคะแนน
4. คะแนนรวมมากกว่า 50
 - 4.1 ถ้าจริง สอบผ่าน
 - 4.2 ถ้าเท็จ สอบไม่ผ่าน

จบ



แบบฝึกหัด

- ทดลองรันโปรแกรมและหาผลลัพธ์จากการรันโปรแกรมต่อไปนี้

```
1 A = 5
2 B = int(input("INPUT NUMBER : "))
3 if (B > (5+3)):
4     print("สวัสดีครับ")
5 else:
6     print("สบายดีมั๊ยครับ")
7 print("ขอให้สุขภาพแข็งแรง")
8
```

```
1
2 A = int(input("INPUT NUMBER : "))
3 if ((A%2)==0):
4     print("ตัวเลขนี้")
5     print("เป็นเลขคู่")
6 else:
7     print("ตัวเลขนี้")
8     print("เป็นเลขคี่")
9 print("ขอให้สุขภาพแข็งแรง")
10
```

แบบฝึกหัด

- จงเขียนอัลกอริทึม สำหรับคำนวณผลการเรียน โดยคะแนนจะมาจาก การบ้านมีคะแนนเต็ม 20 คะแนน สอบกลางภาคมีคะแนนเต็ม 40 คะแนน สอบปลายภาคมีคะแนนเต็ม 40 คะแนน โดยเงื่อนไขผลการเรียนเป็นดังนี้
 - มากกว่า 80 คะแนน ดีมาก
 - มากกว่า 50 คะแนน ผ่าน
 - คะแนน 0 ถึง 50 ไม่ผ่าน
- อินพุต
- ประมวลผล
- เอาต์พุต



ประมวลผลทางคณิตศาสตร์

รับตัวเลขและการดำเนินการ

จริง
เลือก บวก
มิฉะนั้น

แสดงผลการบวก

จริง
เลือก ลบ
มิฉะนั้น

แสดงผลการลบ

จริง
เลือก คูณ
มิฉะนั้น

แสดงผลการคูณ

ต่อไปหรือจบ

$$a = 4$$

$$b = 3$$

add/sub/mul/div : mul

$$\text{Answer} = 12$$

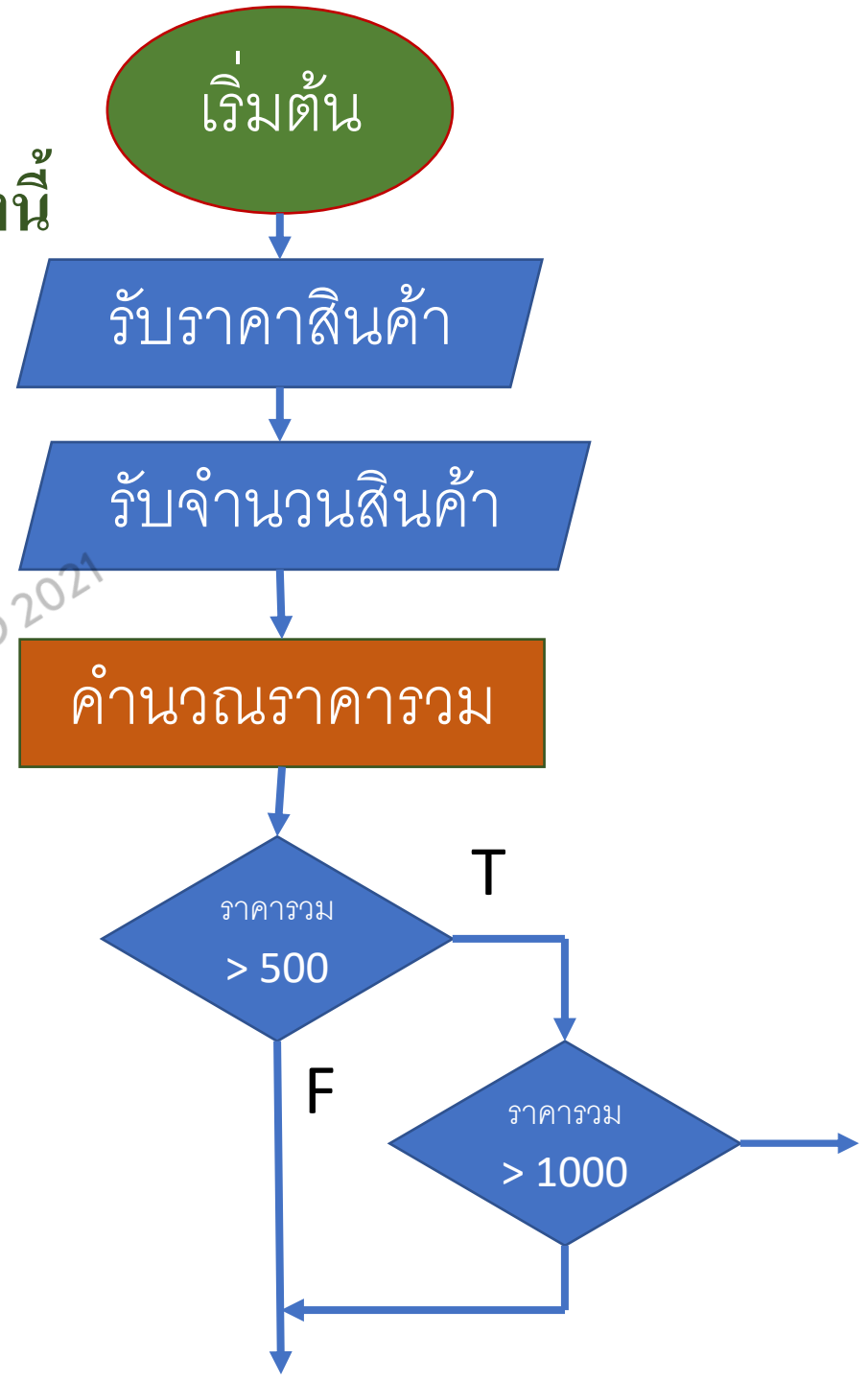
KMITL
MASTER
CLASS
THE LEGEND 2021

แบบฝึกหัด



การคำนวณวงจรราคาสินค้า

- ร้านค้าร้านหนึ่งขายของหลายชนิด และมีส่วนลดดังนี้
 ราคารวมตั้งแต่ 500 บาทขึ้นไป ลด 10%
 ราคารวมตั้งแต่ 1000 บาทขึ้นไป ลด 20%



KMITL
 MASTER
 CLASS
 THE LEGEND 2021

1. หาผลลัพธ์จากการทำงานต่อไปนี้

x = 3

y = 4

print(x+y)

print(x==y)

print((x+1)<y)

print((x+1)<=y)

x = x+y

y = x+1

print("x = ",x)

print("y = ",y)

คำตอบ

INPUT x =

INPUT y =

2. หาผลลัพธ์จากการทำงานต่อไปนี้

x = 3

y = 4

z = x+y

print("z ^ 2 = ",z**2)

print("z % 3 = ",z%3)

z = (x+2)%x

print("z = ",z)

print(z, ">", y, " ", z>y)

คำตอบ



KMITL
MASTER
CLASS
THE LEGEND 2021

3. หาผลลัพธ์จากการทำงานต่อไปนี้

x = 4

y = 5

z = x+y

if(z < (y*2)):

 print("สวัสดีครับ.")

else:

 print("เป็นอย่างไรบ้างครับ")

print("ขอให้สุขภาพแข็งแรงนะครับ")

คำตอบ



KMITL
MASTER
CLASS
THE LEGEND 2021

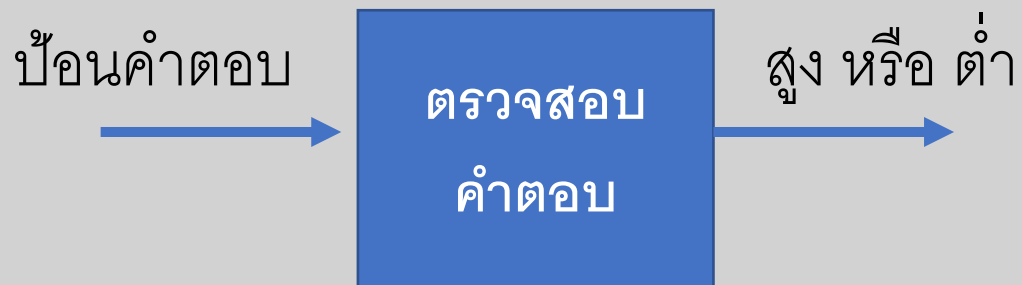
แบบฝึกหัด

- ร้านค้าร้านหนึ่งขายของหลายชนิด และมีส่วนลดดังนี้
 - ราคารวมทั้งตั้งแต่ 500 บาทขึ้นไป ลด 10%
 - ราคารวมทั้งตั้งแต่ 1000 บาทขึ้นไป ลด 20%
 - ให้แสดงราคาที่ต้องจ่ายรวมกับภาษีการค้า 7%
- เขียนผังงานพร้อมโปรแกรม

โจทย์ ทอยลูกเต๋า

- ลองเขียนโปรแกรม
ทอยลูกเต๋า **3** ลูก

KMITL
MASTER
CLASS
THE LEGEND 2021



โจทย์

- คะแนนการเรียนวิชาหนึ่งแบ่งเป็นดังนี้
 - การบ้านเต็ม **40**
 - สอบกลางภาคเต็ม **20**
 - สอบปลายภาคเต็ม **40**

เขียนโปรแกรมประมวลผล ถ้าคะแนนตั้งแต่ **50** ขึ้นไป ผ่าน

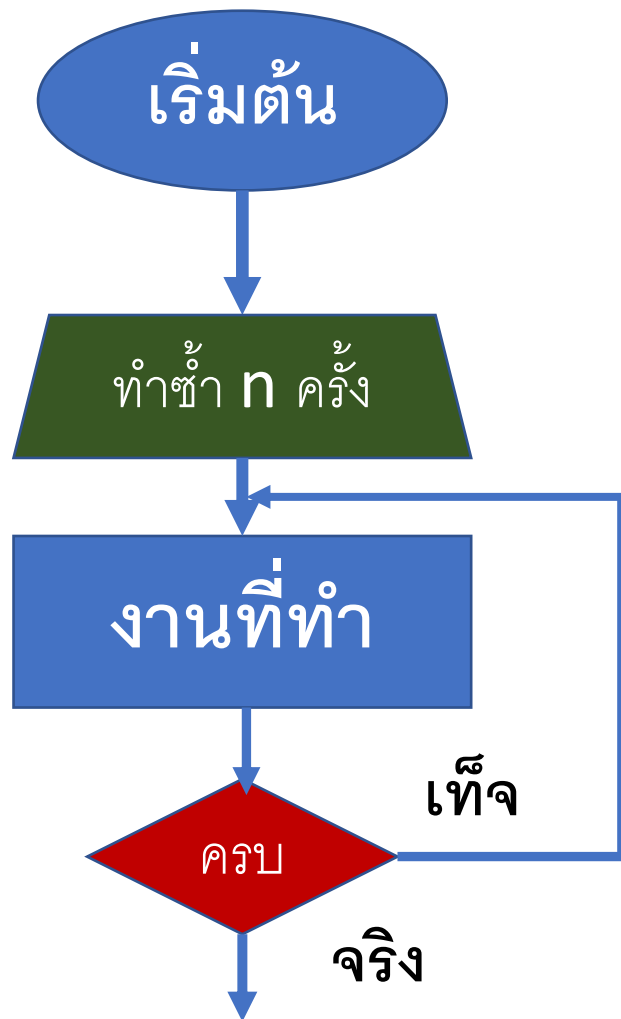
แบบฝึกหัด

เขียนโปรแกรมรับค่าคะแนน แล้วให้ตัดเกรด โดยมีเงื่อนไขต่อไปนี้

- คะแนนตั้งแต่ **80** ได้ **A**
- คะแนนตั้งแต่ **70** ถึง **79** ได้ **B**
- คะแนนตั้งแต่ **60** ถึง **69** ได้ **C**
- คะแนนตั้งแต่ **50** ถึง **59** ได้ **D**
- คะแนนน้อยกว่า **50** ได้ **F**

KMITL
MASTER
CLASS
THE LEGEND 2021

การทำซ้ำ



for x in range(10):
print(x)

0
1
2
3
4
5
6
7
8
9

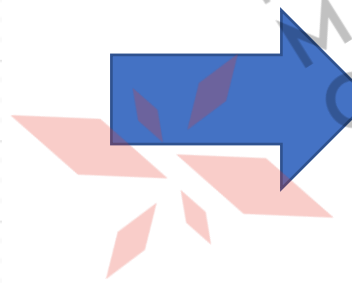
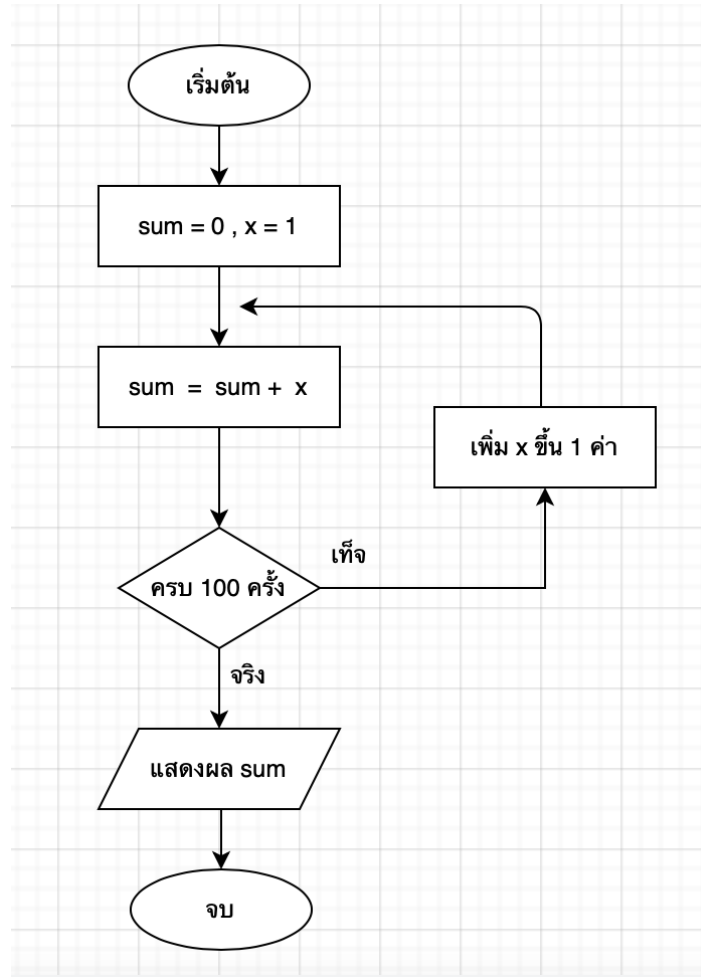
ตัวอย่าง

```
sub1 = ['Physics','Math','Chem']  
print('วิชาที่ผมชอบคือ :',sub1)
```

```
Data1 = [3, 8, 2, 9, 5]  
for i in range(5):  
    print(Data1[i])
```



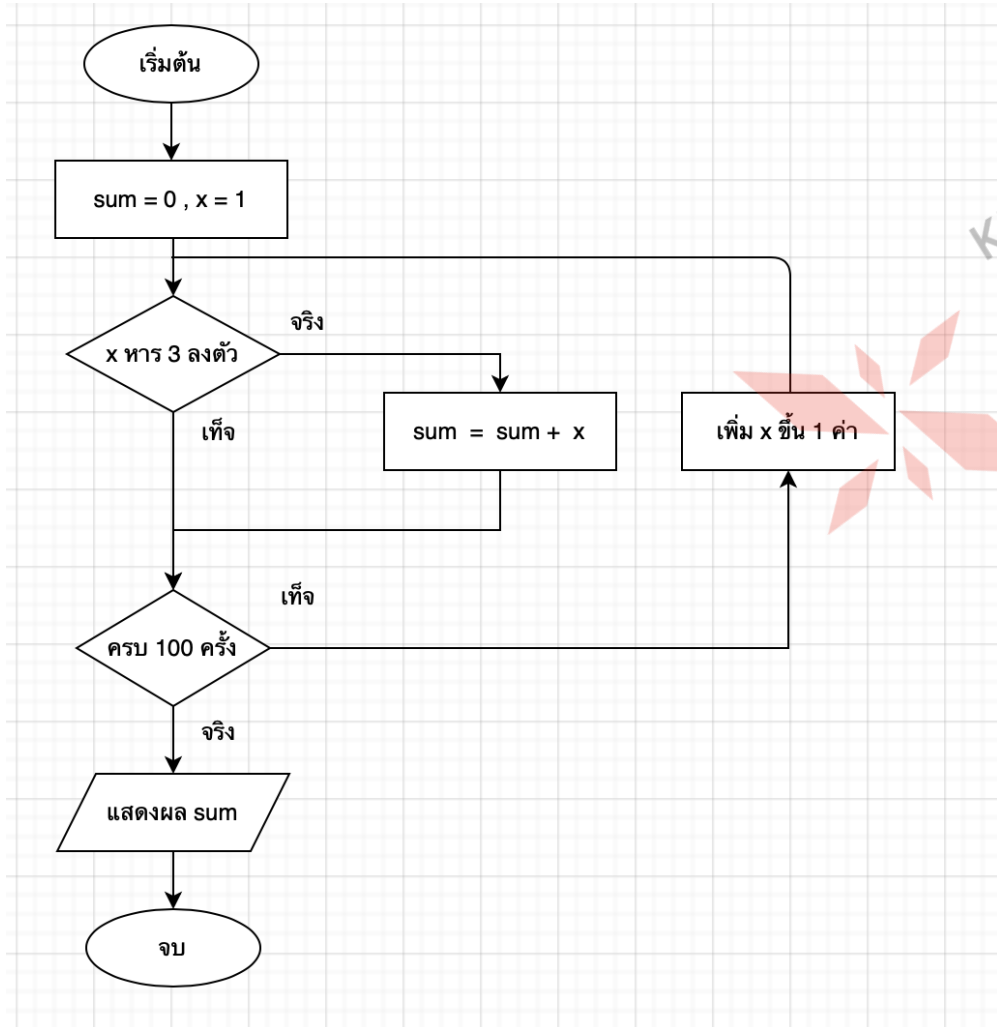
หาผลรวม



```
sum = 0
for x in range(1,101,1):
    sum = sum + x
print("sum = ",sum)
```

The text block shows a Python code snippet for calculating the sum of numbers from 1 to 100. A large blue arrow points from the flowchart to this code. A watermark 'KMITL MASTER CLASS THE LEGEND 2021' is visible in the background.

หาผลรวมของตัวเลขที่หารด้วย 3 ลงตัว



sum = 0

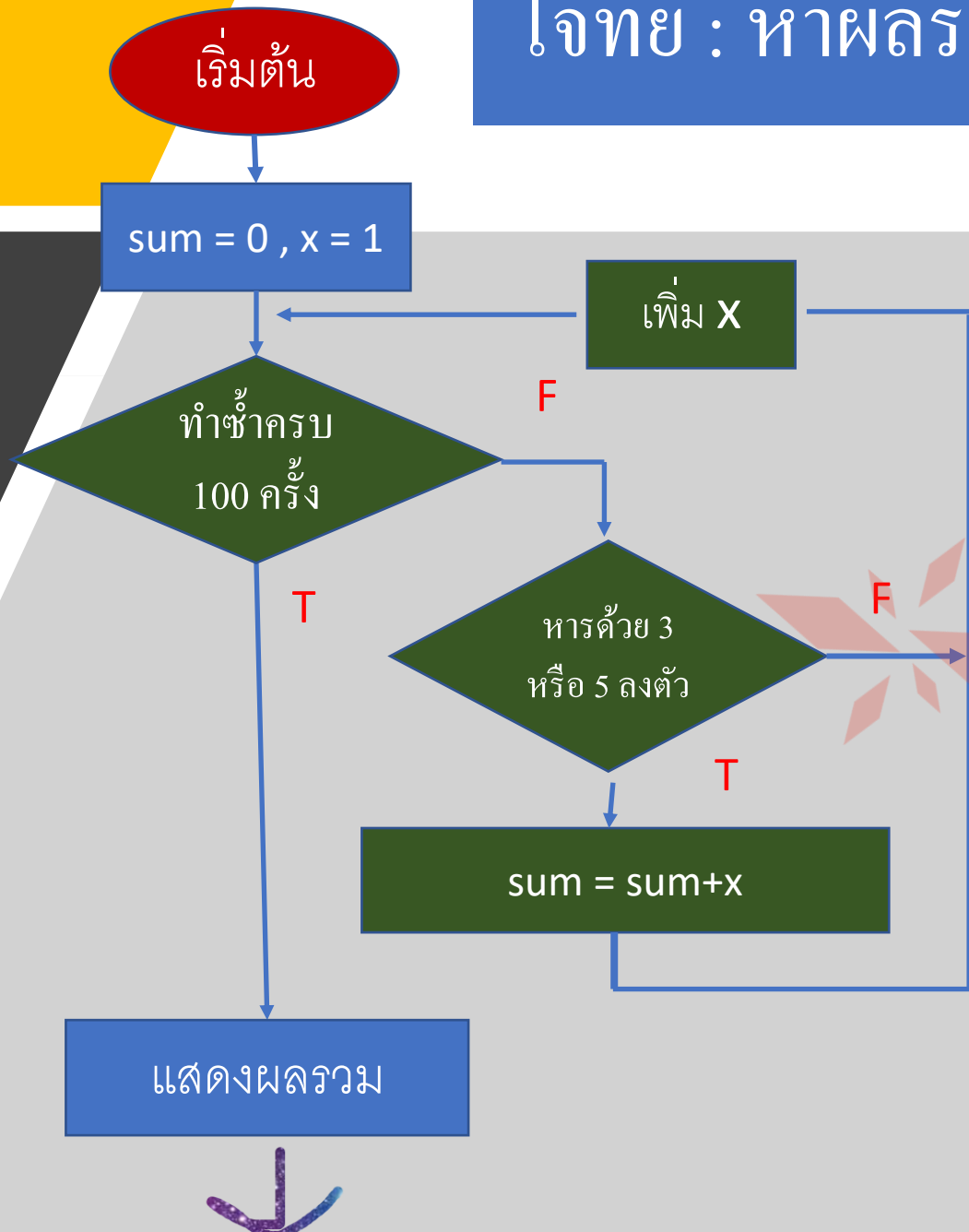
```
for x in range(1,101,1):
```

```
    if ((x%3)==0):
```

```
        sum = sum + x
```

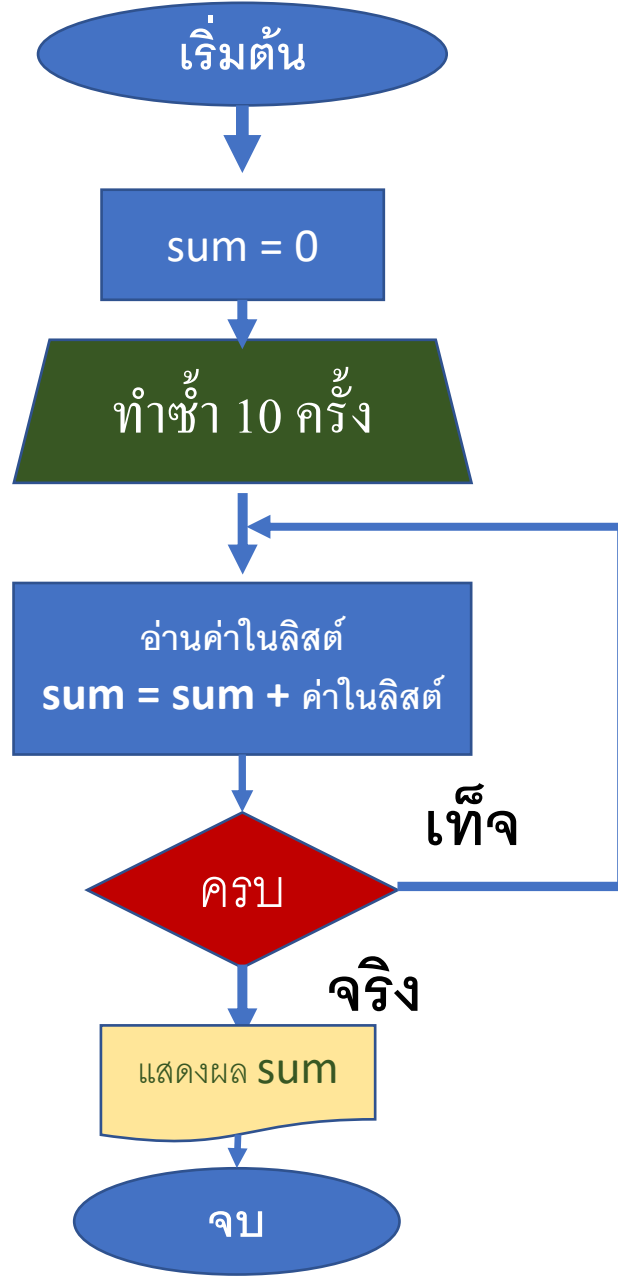
```
    print("sum = ",sum)
```

โจทย์ : หาผลรวมตัวเลขที่หารด้วย 3 หรือ 5 ลงตัว



```
sum = 0
for x in range(1,101,1):
    if ((x%3) == 0) or ((x%5) == 0):
        print(x)
        sum = sum+x
print('sum = ', sum)
```

หาผลรวม



```
import random
```

```
A = random.sample(range(1,12),10)
```

```
sum_i = 0
```

```
for x in A:
```

```
    sum_i = sum_i + x
```

```
print("A = :",A)
```

```
print("Sum = ",sum_i)
```

```
import random
```

```
A = random.sample(range(1,12),10)
```

```
m = sum(A)
```

```
print("A = :",A)
```

```
print("Sum = ",m)
```

A = : [4, 11, 3, 7, 1, 5, 2, 10, 6, 9]

Sum = 58

แบบฝึกหัด

- หาผลรวมของตัวเลข **a** ถึง **b** โดยเป็นเลขที่หารด้วย **3** หรือ **5** ลงตัว
ตัวอย่าง

a =

b =

sum =

ตัวอย่าง

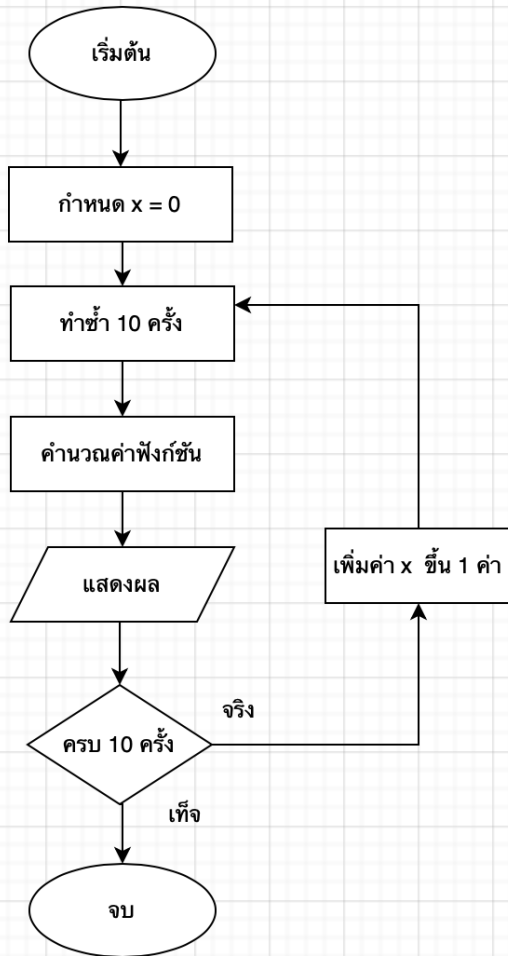
a = 1

b = 10

sum = 33



คำนวณค่าฟังก์ชัน



```
for x in range(10):  
    fx = x**2 + 1  
    print("f(",x,") = ",fx)
```



WITL
MASTER
CLASS
THE LEGEND 2021

Function และ OOP

ธีรวัฒน์ ประกอบผล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร
ลาดกระบัง



ฟังก์ชัน
กับ
เมธอด

- โปรแกรมภาษาโครงสร้าง (Structured Programming)

นิยมเรียกว่า “ฟังก์ชัน”

EX. `x = len(str)`

การเรียกใช้ฟังก์ชัน

- โปรแกรมภาษาเชิงวัตถุ (Object-Oriented Programming)

นิยมเรียกว่า “เมธอด”

EX. `str = str.upper()`

`function(a)`
`a.function()`

KMITL
MASTER
CLASS
THE LEGEND 2021

ฟังก์ชันของ Python

- ฟังก์ชันจากไลบรารีมาตรฐาน
- ฟังก์ชันที่สร้างขึ้นมาเอง



KMITL
MASTER
CLASS
THE LEGEND 2021

การสร้างฟังก์ชัน

```
def function_name([parameter list]):  
    statement  
    .....  
    [return expression]
```

```
def is_PrimeNumber(n):  
    prime = True  
    for i in range(2,n):  
        if ((n%i)==0):  
            prime = False  
    return prime  
return prime
```

```
x = int(input("ป้อนตัวเลขที่ต้องการตรวจสอบ : "))  
print("เลข ",x," เป็นจำนวนเฉพาะ ",is_PrimeNumber(x))
```

Function



*f*x

$$f(x) = x^2 + 1$$

ฟังก์ชันมีการคืนค่า

```
def f_x(x):  
    y = 3*x**2+1  
    return y  
  
for i in range(1,11,1):  
    print('f(',i,')=',f_x(i))
```

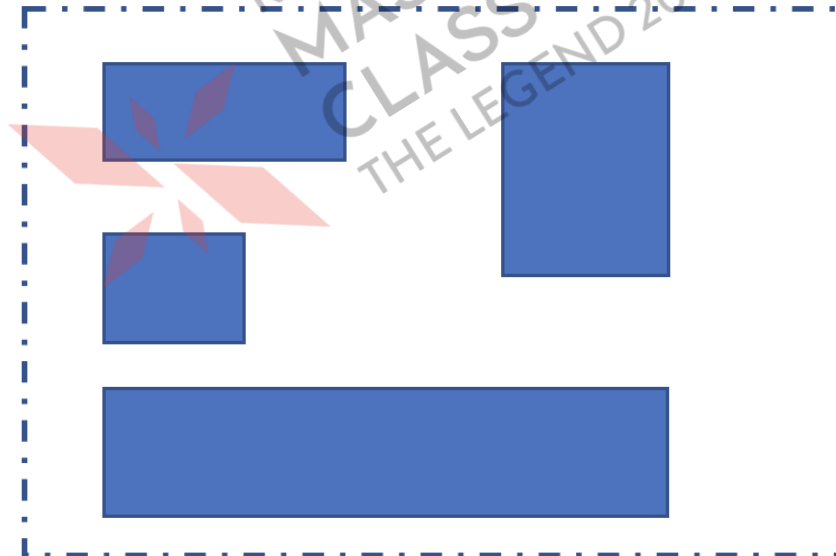
```
----- RESTART: -----  
f( 1 )= 4  
f( 2 )= 13  
f( 3 )= 28  
f( 4 )= 49  
f( 5 )= 76  
f( 6 )= 109  
f( 7 )= 148  
f( 8 )= 193  
f( 9 )= 244  
f( 10 )= 301  
>>>
```

การเขียนโปรแกรมเชิงวัตถุ

- นิยามคลาส

Ractangle

width
length
<u>setWidth</u>
<u>setLength</u>
<u>getArea</u>



KMITL
MASTER
CLASS
THE LEGEND 2021

```
class className :
```

```
    class suit
```

```
        .....
```

ตัวอย่าง

```
class Ractangle :
```

```
    def get_area(self, w, l)
```

```
        return w*l
```



```
class Box:  
    pass
```

```
a = Box()  
b = Box()  
print(a)  
print(b)
```

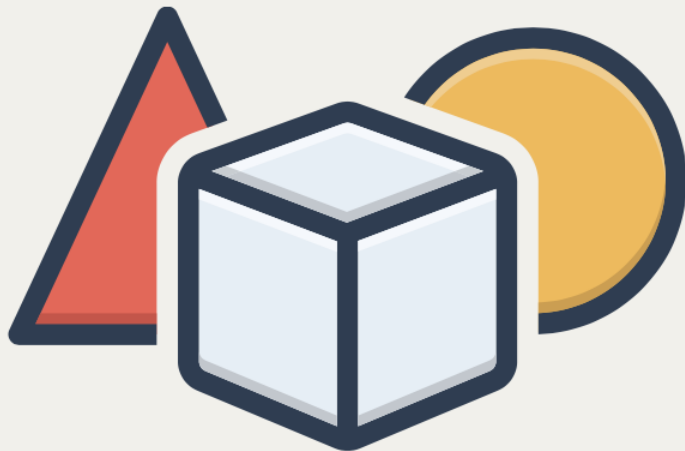
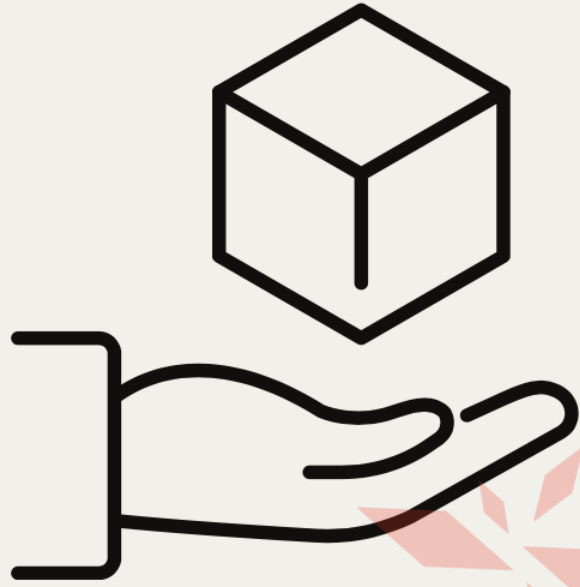
```
class Box:  
    def get_area(self,w,l):  
        return w*l  
    def get_volume(self,w,l,h):  
        return w*l*h  
  
box_A = Box()  
box_B = Box()  
  
print("box_A Area = ",box_A.get_area(2,4))  
print("box_A Volume = ",box_A.get_volume(2,4,6))  
print("box_B Area = ",box_B.get_area(4,6))  
print("box_B Volume = ",box_B.get_volume(4,6,8))
```

```
class Box :  
    def __init__(self, w, l):  
        self.w = w  
        self.l = l  
    def get_area(self):  
        return (self.w*self.l)
```

← คลาส Box



OOP PROGRAMMING



```
class Box:  
    def get_area(self,w,l):  
        return w*l  
    def get_volume(self,w,l,h):  
        return w*l*h
```

```
box_A = Box()  
box_B = Box()
```

```
print("box_A Area = ",box_A.get_area(2,4))  
print("box_A Volume = ",box_A.get_volume(2,4,6))  
print("box_B Area = ",box_B.get_area(4,6))  
print("box_B Volume = ",box_B.get_volume(4,6,8))
```

ผลการรัน



```
box_A Area = 8  
box_A Volume = 48  
box_B Area = 24  
box_B Volume = 192  
>>>
```

```
class Box :
    def __init__(self, name,w, l):
        self.name = name
        self.w = w
        self.l = l
    def get_area(self):
        return (self.w*self.l)
    def __str__(self):
        return(str(self.name)+" มีพื้นที่ = "+str(self.get_area()))
```



KMITL
MASTER
CLASS
THE LEGEND 2021

```
class Student :
    def __init__(self, name):
        self.name = name
        self.H = 0
        self.M = 0
        self.F = 0
        self.gread = ''
    def get_sum(self):
        return (self.H+self.M+self.F)
    def get_grade(self):
        sum = self.get_sum()
        if (sum < 50):
            self.grade = 'F'
        elif ((sum>=50)and(sum<60)):
            self.grade = 'D'
        elif ((sum >= 60) and (sum < 70)):
            self.grade = 'C'
        elif ((sum >= 70) and (sum < 80)):
            self.grade = 'B'
        else:
            self.grade = 'A'
        return self.grade

    def __str__(self):
        return(str(self.name)+" มีคะแนนรวม = "+str(self.get_sum()))
```

```
st1 = Student("Tee")
st1.H = 18
st1.M = 37
st1.F = 30
st2 = Student("Pui")
st2.H = 15
st2.M = 34
st2.F = 25
print(st1)
print(st1.get_grade())
print(st2)
print(st2.get_grade())
```

KMITL
MASTER
CLASS
THE LEGEND 2021

แบบฝึกหัด

- เขียนโปรแกรมบ็อนคะแนนการบ้าน คะแนนสอบกลางภาค คะแนนสอบปลายภาคของนักเรียน **10** คน จากนั้นแสดงผลข้อมูลของนักเรียน พร้อมเกรดแสดงผลออกมาอย่างน่าสนใจ



KMITL
MASTER
CLASS
THE LEGEND 2021

การประมวลผลทาง คณิตศาสตร์



KMITL
MASTER
CLASS
THE LEGEND 2021

ธีรวัฒน์ ประกอบผล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

การใช้งานโมดูล (Module)

ตัวอย่างโมดูล

- โมดูลคณิตศาสตร์ (math module)
- โมดูลแสดงปฏิทิน (calendar module)
- โมดูลแสดงเวลา (time module)
- โมดูลพัฒนา GUI (tkinter module)

รูปแบบการเรียกใช้โมดูลและฟังก์ชัน

- ใช้คำสั่ง `import`
เรียกใช้ฟังก์ชันในโมดูล
- ใช้คำสั่ง `from .. import`
เรียกใช้เฉพาะฟังก์ชันภายในโมดูล
- ใช้คำสั่ง `from .. import *`
โหลดฟังก์ชันที่มีอยู่ทั้งหมดเข้ามา
- ใช้คำสั่ง `import .. as`
โหลดเข้ามาแล้วเปลี่ยนชื่อ

```
import math  
print(math.pow(5,2))
```

```
from math import pow  
print(pow(5,2))
```

```
import matplotlib.pyplot as pl1
```

การประยุกต์

- NumPy/SciPy ฟังก์ชันกรรมวิธีเชิงตัวเลข และวิทยาศาสตร์
- ALGLIB การวิเคราะห์ทางพีชคณิต
- Pandas เครื่องมือในการวิเคราะห์ข้อมูลและโครงสร้างข้อมูล
- PyGSL การเชื่อมต่อกับ GNU Scientific Library
- ScientificPython โมดูลการคำนวณทางวิทยาศาสตร์

NumPy

- มาจากคำว่า Numeric Python หรือ Numerical Python

- เป็น โมดูลขยาย บรรจุฟังก์ชันสำหรับการคำนวณเชิงตัวเลข

- ออกแบบลักษณะของโครงสร้างข้อมูลใหม่

- เมื่อแสดงข้อมูลออกมาจะไม่มีเครื่องหมายคอมมา ,

```
>>> import numpy as np
```

```
>>> data = [1,2,3,4,5]
```

```
>>> narray = np.array(data)
```

```
>>> print(data)
```

```
[1, 2, 3, 4, 5]
```

```
>>> print(narray)
```

```
[1 2 3 4 5]
```

```
>>>
```

คณิตศาสตร์

ตัวอย่างฟังก์ชันอื่น ๆ

```
import math
print(math.e ** 2)
```

```
import math
x = math.e ** 2
print('คำตอบคือ = ', x)
```

```
Python 3.7.3 Shell
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 16:39:00)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /Users/teerawatprakobphon/Documents/test_math.py =====
7.3890560989306495
>>> |
```

```
sqrt(x)
factorial(x)
pow(x,y)
sin(x)
```

```
>>math.sqrt(100)
```

คณิตศาสตร์

ใช้วงเล็บในการจัดกลุ่มประมวลผล

➤ $(6 + 5) * 3$

➤ $6 + (5 * 3)$

เรียกใช้โมดูลที่ต้องการ เช่น การสุ่มตัวเลข

```
>> from random import randint
```

```
>> randint(1,9)
```

```
>>x = randint(1,9)
```

```
>>y = randint(1,9)
```

```
>>z = x+y
```

ฟังก์ชันแบบ Lambda (แลมบ์ดา)

- ใช้คำว่า lambda มีรูปแบบดังนี้

var = lambda arg1[,arg2]... : expression

f_x = lambda x: x**2+1

a = f_x(4)

print('f_x = ',a)

ave = lambda x,y,z: (x+y+z)/3

a = ave(4,5,6)

print('f_x = ',a)

โจทย์

1. $y = x^2 + 2x + 1$

2. $y = (x+1)^3 + x$

3. $y = \sqrt{(x + 1)}$

4. $y = 2x+1$

โจทย์

$$f(x) = \sqrt{(x^2 + 3)}$$

เมื่อ $x = 2, 4, 6, \dots, 20$



PRIME NUMBERS



KMIT
MASTERS
CLASSES
THE LEGENDS 2021

จำนวนเฉพาะ

เริ่มต้น

1. กำหนดตัวแปร prime เป็น 0
2. รับค่าตัวเลข n
3. กำหนดค่า x เป็น 2 แล้วทำซ้ำจนถึง n - 1
 - 3.1 ตรวจสอบว่า n หารด้วย x ลงตัวหรือไม่
 - 3.1.1 ถ้าหารลงตัว ให้ prime เป็น 1
4. ถ้า prime เป็น 0
 - 4.1 เป็นจำนวนเฉพาะ
มีฉะนั้น
 - 4.2 ไม่เป็นจำนวนเฉพาะ

จบ

```
prime = 0
```

```
n = int(input("ป้อนค่าตัวเลขจำนวนเต็ม : "))
```

```
for x in range(2,n):
```

```
    if ((n%x)==0):
```

```
        prime = 1
```

```
if prime == 0 :
```

```
    print("เป็นเลขจำนวนเฉพาะ")
```

```
else:
```

```
    print("ไม่เป็นเลขจำนวนเฉพาะ")
```

จำนวนเฉพาะ..โปรแกรม

```
prime = True
n = int(input("ป้อนค่าตัวเลขจำนวนเต็ม : "))
for x in range(2,n):
    if ((n%x)==0):
        prime = False
if prime :
    print("เป็นเลขจำนวนเฉพาะ")
else:
    print("ไม่เป็นเลขจำนวนเฉพาะ")
```



การเขียนโปรแกรม
ประยุกต์
ด้านคณิตศาสตร์
วิทยาศาสตร์

With Teacher Casey



ด้านตรงข้ามมุมฉาก

```
import math
A = 3
B = 4
C = math.sqrt((A**2)+(B**2))
print("C = ",C)
```

```
import math
A = 3
B = 4
C = math.hypot(A,B)
print("C = ",C)
```

เขียนโปรแกรม ภาษา Python

KMITL
MASTER
CLASS
THE LEGEND 2021

B = 4



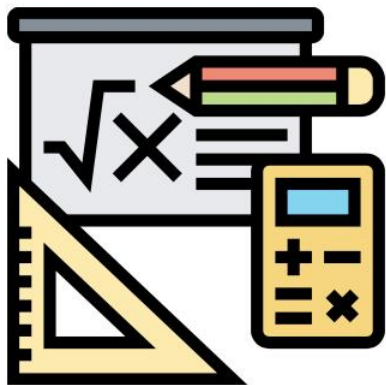
C = ?

A = 3

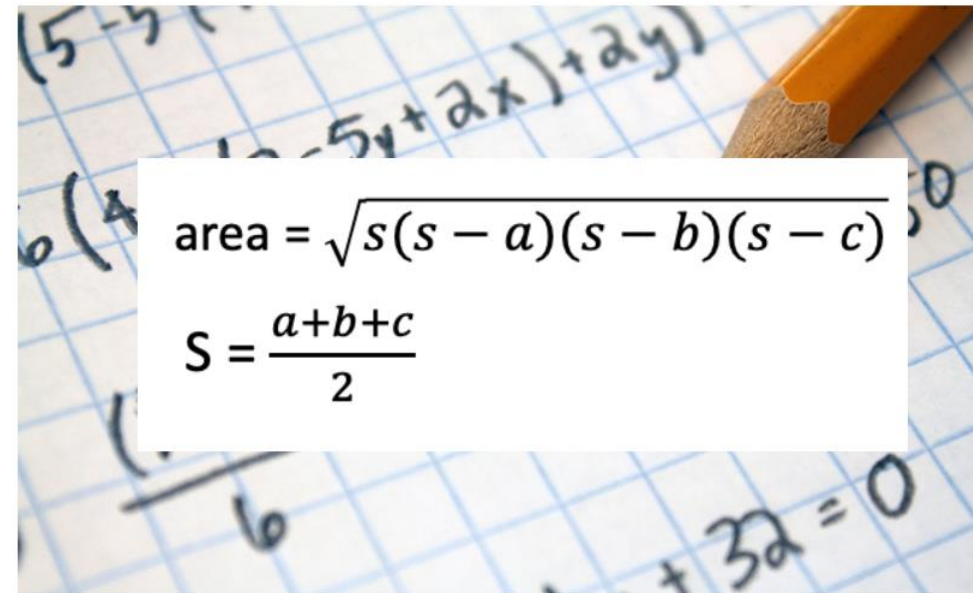
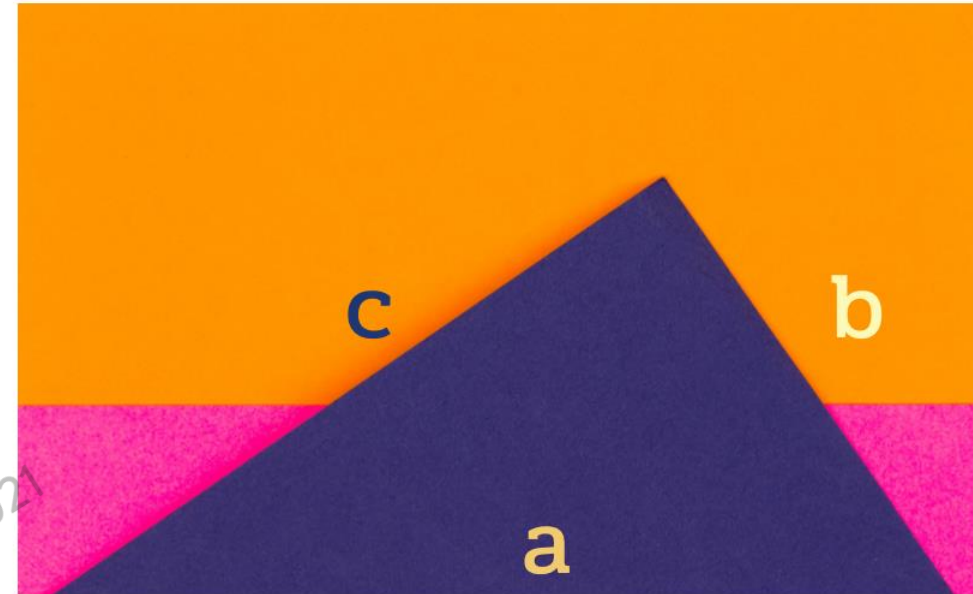


คำนวณพื้นที่สามเหลี่ยม

```
import math
a = int(input("a = : "))
b = int(input("b = : "))
c = int(input("c = : "))
s = (a+b+c)/2
area = math.sqrt(s*(s-a)*(s-b)*(s-c))
print("Area = %6.2f"%area)
```



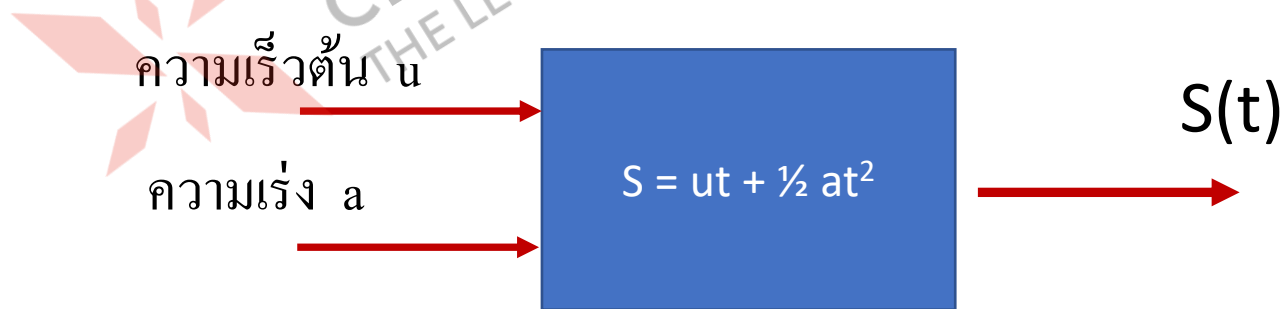
```
a = : 5
b = : 4
c = : 3
Area = 6.00
```



การเคลื่อนที่ ด้วยความเร่ง



รถยนต์คันหนึ่ง ผ่านจุด A ด้วยความเร็ว 2 m/s จากนั้นคนขับได้ใช้ความเร่ง 3 m/s^2 อยากทราบระยะทางห่างจากจุด A เมื่อเวลาผ่านไปทุก ๆ 2 วินาที จนถึงวินาทีที่ 20 รถมาถึงจุด B



สมการการเคลื่อนที่

KMITL
MASTER
CLASS
THE LEGEND 2021



$$S = ut + (1/2)at^2$$


```
u = 2
```

```
a = 3
```

```
def motion(t):
```

```
    s = u*t+0.5*a*(t**2)
```

```
    return s
```

```
for x in range(2,21,2):
```

```
    print("t = ",x, " s = ",motion(x))
```

```
t = 2 s = 10.0
```

```
t = 4 s = 32.0
```

```
t = 6 s = 66.0
```

```
t = 8 s = 112.0
```

```
t = 10 s = 170.0
```

```
t = 12 s = 240.0
```

```
t = 14 s = 322.0
```

```
t = 16 s = 416.0
```

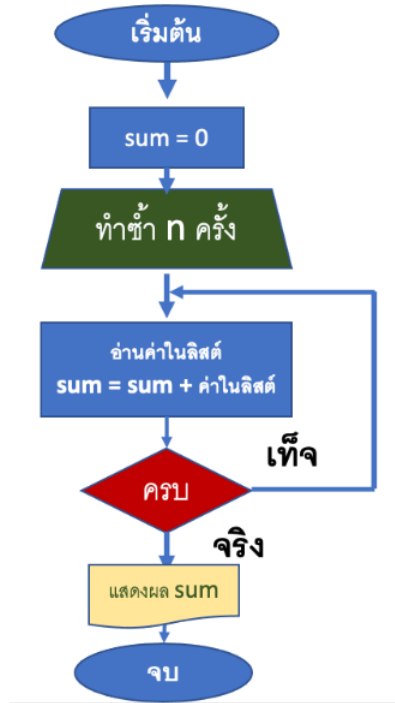
```
t = 18 s = 522.0
```

```
t = 20 s = 640.0
```

```
>>>
```

KMITL
MASTER
CLASS
THE LEGEND 2021

หาผลรวมของตัวเลข 10 ค่า



คิดเอง

```
import random
A = random.sample(range(1,12),10)
sum_i = 0
for x in A:
    sum_i = sum_i + x
print("A = :",A)
print("Sum = ",sum_i)
```

ใช้ฟังก์ชัน

```
import random
A = random.sample(range(1,12),10)
m = sum(A)
print("A = :",A)
print("Sum = ",m)
```

A = : [4, 11, 3, 7, 1, 5, 2, 10, 6, 9]
Sum = 58

แก้สมการ

- $Ax^2 + Bx + C = 0$

A =

B =

C =



```
import math
```

```
A = int(input("A = : "))
```

```
B = int(input("B = : "))
```

```
C = int(input("C = : "))
```

```
x1 = (-B+math.sqrt((B**2)-4*A*C))/(2*A)
```

```
x2 = (-B-math.sqrt((B**2)-4*A*C))/(2*A)
```

```
print("The roots are : ",x1, " and ",x2)
```

ตัวอย่างการรัน

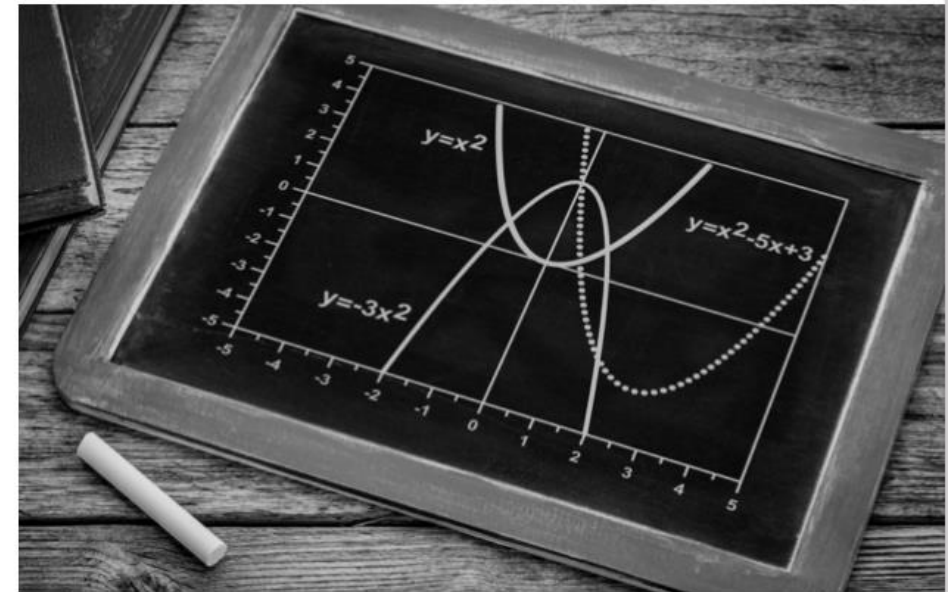
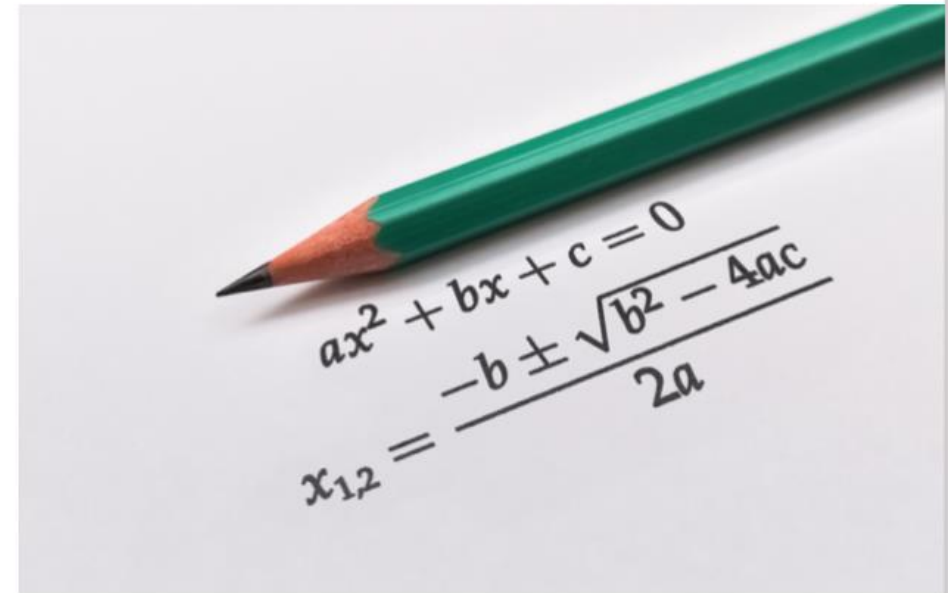
```
A = : 1
```

```
B = : 3
```

```
C = : 1
```

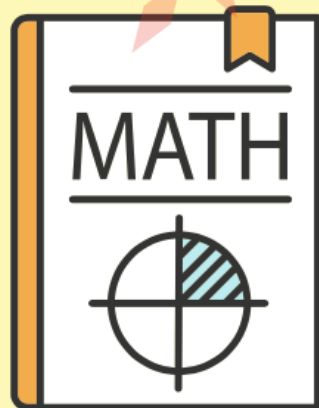
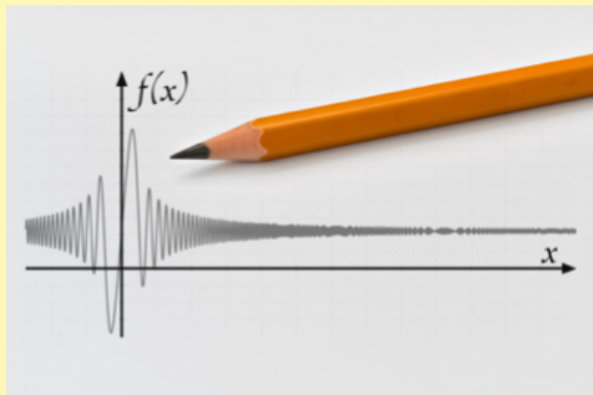
```
The roots are : -0.3819660112501051 and -2.618033988749895
```

ลองคิดต่อ



f_x

ส่งค่าเข้าไป
คืนค่าออกมา



```
def f_x(x):
    m = x**2 + 1
    return m
```

```
print(" f(x) = x^2 + 1 ")
print("=====")
for i in range(10):
    print("f(",i,")=",f_x(i))
```

ผลลัพธ์



```
f(x) = x^2 + 1
=====
f(0)= 1
f(1)= 2
f(2)= 5
f(3)= 10
f(4)= 17
f(5)= 26
f(6)= 37
f(7)= 50
f(8)= 65
f(9)= 82
>>>
```

```
import random
def random_dice():
    x = random.randint(1,6)
    y = random.randint(1,6)
    z = random.randint(1,6)
    r = [x, y, z]
    return r
```

```
print("ทอยลูกเต๋า")
for x in random_dice():
    print(x)
```

ตัวอย่างการรัน

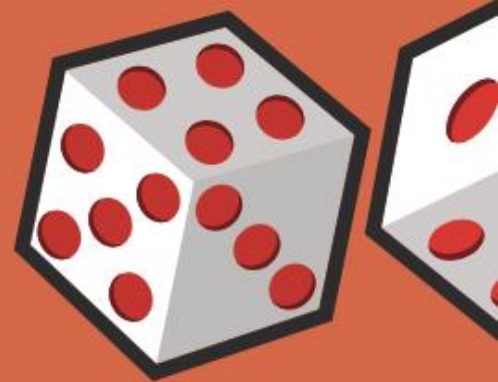
ทอยลูกเต๋า

1

4

4


>>>



Matrix

$$\begin{bmatrix} 3 & 4 & 2 \\ 1 & 3 & 1 \\ 2 & 1 & 4 \end{bmatrix} + \begin{bmatrix} 1 & 2 & 4 \\ 2 & 3 & 1 \\ 2 & 4 & 3 \end{bmatrix} = \begin{bmatrix} 4 & 6 & 6 \\ 3 & 6 & 2 \\ 4 & 5 & 7 \end{bmatrix}$$

A B C



การสร้าง Matrix

การเขียนโปรแกรม

- ควรเลือกใช้ข้อมูลชนิดใด
- ลองเขียนโปรแกรมประมวลผลรูปแบบต่าง ๆ



แสดงเมทริกซ์ A เป็นลิสต์

```
A = [[1, 2, 3],[4, 5, 6],[7, 8, 9]]  
for x in A:  
    print(x)
```

แสดงเมทริกซ์ A

```
A = [[1, 2, 3],[4, 5, 6],[7, 8, 9]]  
for y in A:  
    for x in y:  
        print(x, ' ',end=' ')  
    print()
```

ผลการรันโปรแกรม

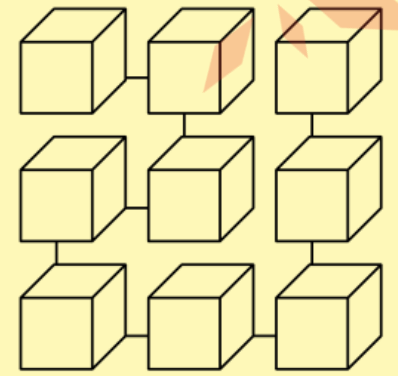
```
[1, 2, 3]  
[4, 5, 6]  
[7, 8, 9]  
>>>
```

ผลการรันโปรแกรม

```
1 2 3  
4 5 6  
7 8 9  
>>>
```




การบวก Matrix

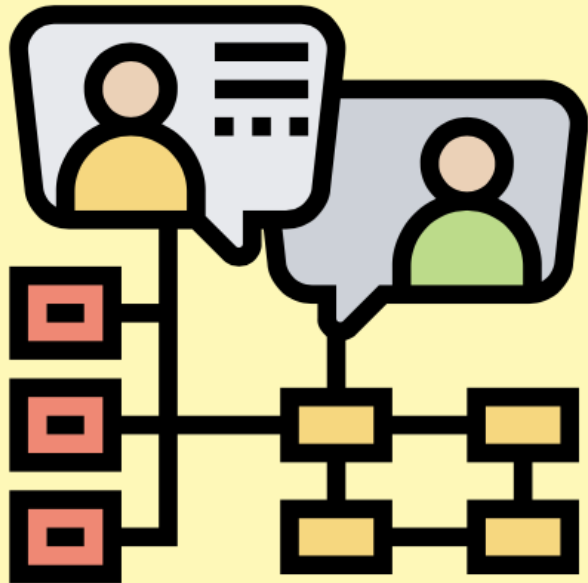


```
A = [[2, 2, 1],[3, 2, 1],[4, 1, 2]]  
B = [[3, 2, 1],[2, 1, 0],[1, 2, 3]]  
C = [[0, 0, 0],[0, 0, 0],[0, 0, 0]]
```

```
for y in range(3):  
    for x in range(3):  
        C[y][x] = A[y][x]+B[y][x]  
  
print(C)
```

```
[[5, 4, 2], [5, 3, 1], [5, 3, 5]]  
>>>
```

ส่งผ่าน Matrix

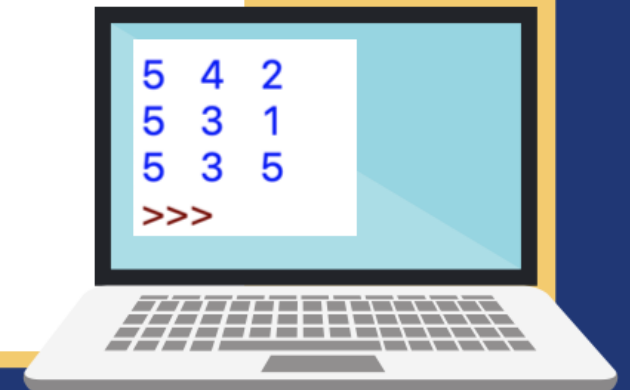


```
A = [[2, 2, 1],[3, 2, 1],[4, 1, 2]]  
B = [[3, 2, 1],[2, 1, 0],[1, 2, 3]]  
C = [[0, 0, 0],[0, 0, 0],[0, 0, 0]]
```

```
def add_matrix(m,n):  
    for y in range(3):  
        for x in range(3):  
            C[y][x] = m[y][x]+n[y][x]
```

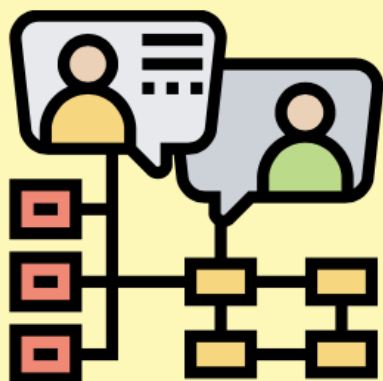
```
def show_matrix(M):  
    for y in M:  
        for x in y:  
            print(x, ' ',end=' ')  
        print()
```

```
add_matrix(A,B)  
show_matrix(C)
```

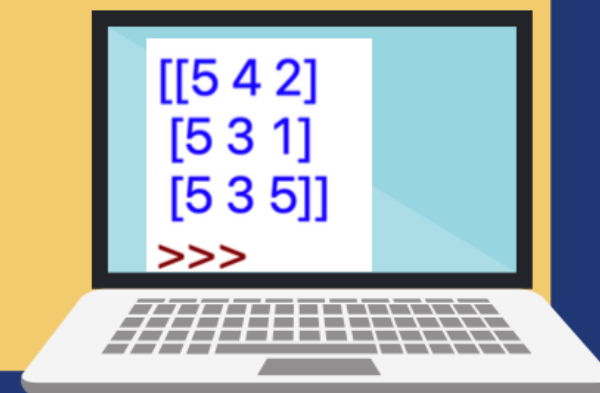


คำนวณ Matrix

ใช้แพ็คเกจ NumPy
สร้างเมทริกซ์ขึ้นมา



```
import numpy as np
A = np.array([[2,2,1],[3,2,1],[4,1,2]])
B = np.array([[3,2,1],[2,1,0],[1,2,3]])
C = A+B
print(C)
```



เขียนตรง ๆ

```
A = [[2, 2, 1],[3, 2, 1],[4, 1, 2]]
B = [[3, 2, 1],[2, 1, 0],[1, 2, 3]]
C = [[0, 0, 0],[0, 0, 0],[0, 0, 0]]
```

```
def add_multiply(m,n):
    for y in range(3):
        for x in range(3):
            for i in range(3):
                C[y][x] += m[y][i]*n[i][x]
    return C
```

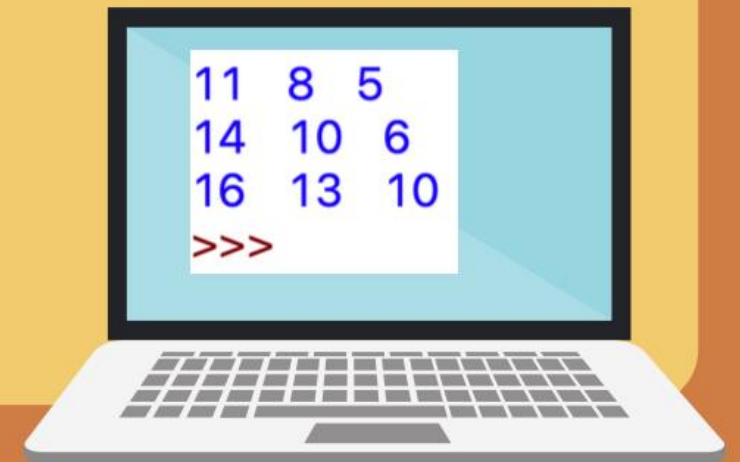
```
def show_matrix(M):
    for y in M:
        for x in y:
            print(x, ' ',end=' ')
        print()
```

```
show_matrix(add_multiply(A,B))
```

ใช้แพ็คเกจ

```
import numpy as np
A = np.array([[2,2,1],[3,2,1],[4,1,2]])
B = np.array([[3,2,1],[2,1,0],[1,2,3]])

A = np.matrix(A)
B = np.matrix(B)
C = A*B
print(C)
```



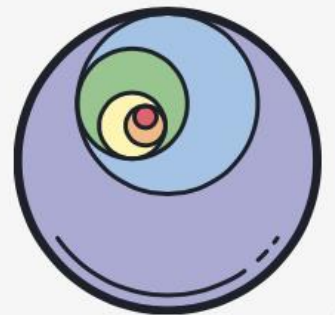
... Fibonacci number



```
def fib(num):  
    if((num==0) or (num==1)):  
        return num  
    return (fib(num - 1) + fib(num - 2))
```

```
x = int(input("Input Number : "))  
for i in range(x):  
    print(fib(i), " ", end=" ")
```

```
Input Number : 12  
0 1 1 2 3 5 8 13 21 34 55 89  
>>>
```



ตัวอย่างการเขียนเซตคณิตศาสตร์

$$S = \{x^2 \mid x \in N, 0 \leq x \leq 10\}$$

$$S = \{0, 1, 4, 9, 16, 25, 36, 49, 64, 81, 100\}$$

ตัวอย่าง

$$\bullet S = \{x^2 \mid x \in \mathbb{N}, 0 \leq x < 10\}$$

```
>>> S = []
```

```
>>> for x in range(10):  
    S += [x**2]
```

```
>>> S
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

```
>>> |
```

KMITL
MASTER
CLASS
THE LEGEND 2021

กราฟและ การประมวลผลข้อมูล

KMITL
MASTER
CLASS
THE LEGEND 2021




```
a = [1,12,5,17,21,8,25,30,16,9]
print("จำนวนสมาชิก คือ ",len(a))
print("สมาชิก คือ ",a)
print("ค่าสูงสุดคือ ",max(a))
a.sort()
print("สมาชิกเรียงลำดับ คือ ",a)
```

จำนวนสมาชิก คือ 10

สมาชิก คือ [1, 12, 5, 17, 21, 8, 25, 30, 16, 9]

ค่าสูงสุดคือ 30

สมาชิกเรียงลำดับ คือ [1, 5, 8, 9, 12, 16, 17, 21, 25, 30]

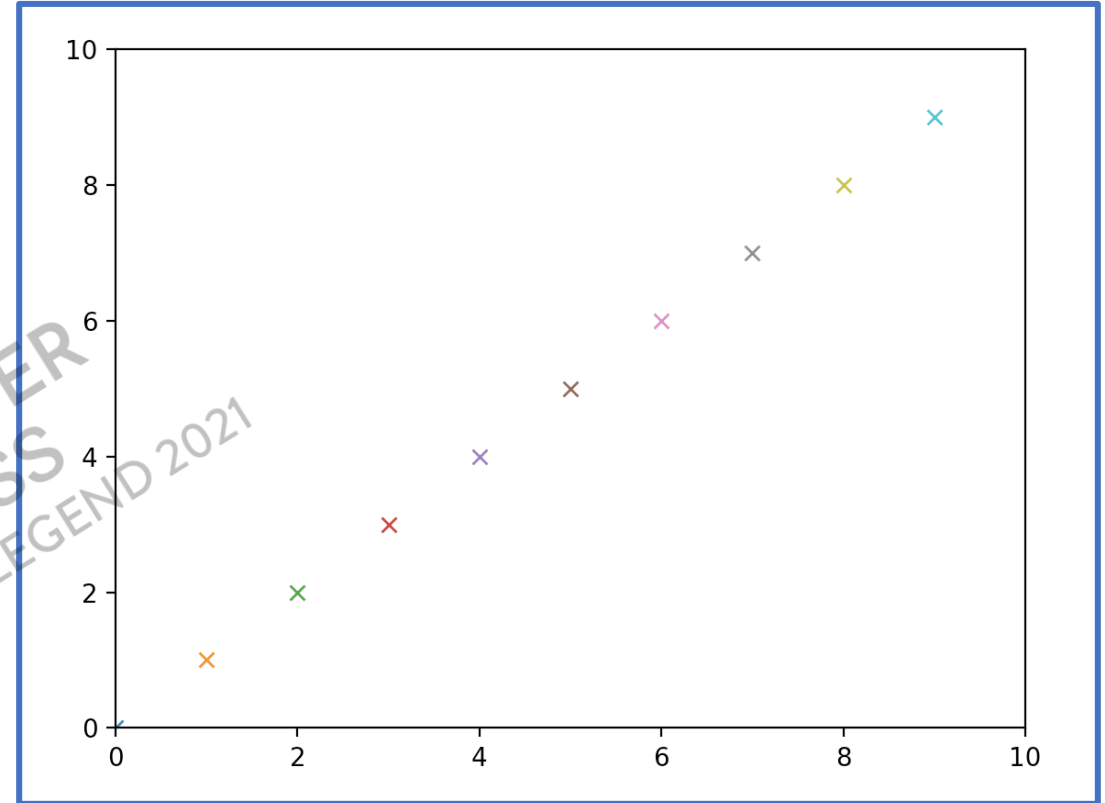
>>>

```
m = []  
for i in range(10):  
    x = int(input("ป้อนค่าตัวเลข : "))  
    m.append(x)  
print(m)
```



ป้อนค่าตัวเลข : 3
ป้อนค่าตัวเลข : 2
ป้อนค่าตัวเลข : 5
ป้อนค่าตัวเลข : 6
ป้อนค่าตัวเลข : 7
ป้อนค่าตัวเลข : 2
ป้อนค่าตัวเลข : 3
ป้อนค่าตัวเลข : 4
ป้อนค่าตัวเลข : 9
ป้อนค่าตัวเลข : 8
[3, 2, 5, 6, 7, 2, 3, 4, 9, 8]

การวาดกราฟ ด้วยภาษา Python



เรียกใช้โมดูล `matplotlib`

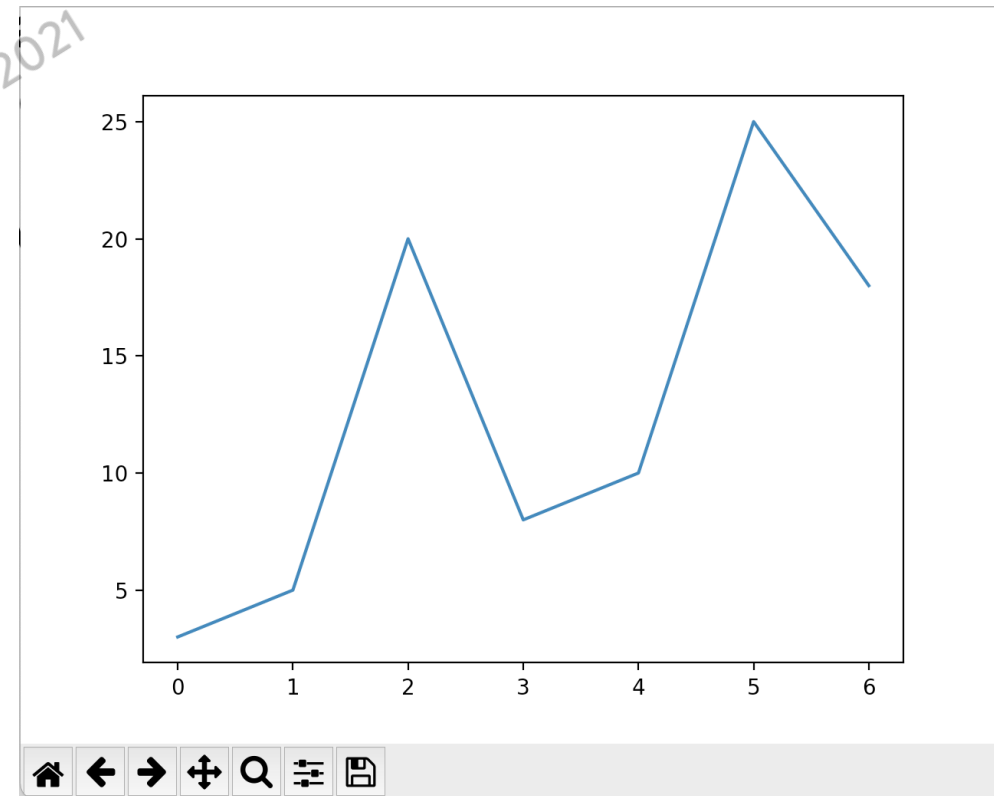
วาดกราฟได้หลายรูปแบบ เก็บไฟล์ภาพไว้ได้

ฟังก์ชันวาดกราฟอย่างง่าย

```
import matplotlib.pyplot as pl
```

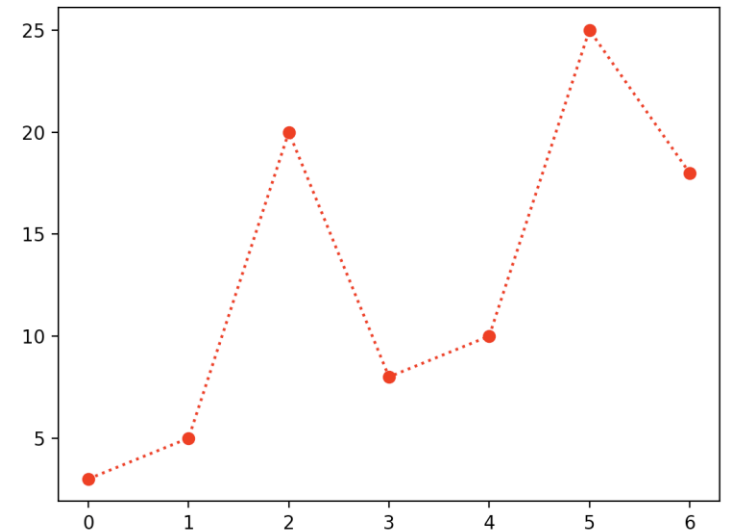
```
pl.plot([3,5,20,8,10,25,18])
```

```
pl.show()
```



ฟังก์ชันวาดกราฟอย่างง่าย

```
import matplotlib.pyplot as plt  
plt.plot([3,5,20,8,10,25,18], 'o:r')  
plt.show()
```



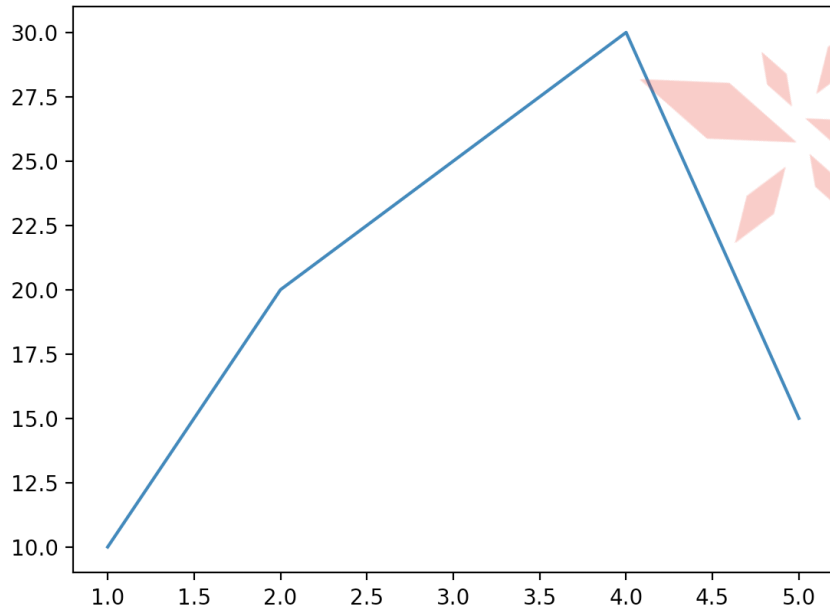
คำสั่ง plot



```
plt.plot(x_values, y_values, format_string[,x,y,format,])
```























การวาดกราฟเส้น



KMITL
MASTER
CLASS
THE LEGEND 2021

```
import matplotlib.pyplot as plt  
x = [1, 2, 3, 4, 5]  
y = [10, 20, 25, 30, 15]  
  
plt.plot(x, y)  
plt.show()
```

มาร์กเกอร์ ลักษณะเส้น สีของกราฟ

อักขระ	มาร์กเกอร์	อักขระ	มาร์กเกอร์	อักขระ	มาร์กเกอร์	อักขระ	มาร์กเกอร์
'o'		'v'		'h'		'1'	
's'		'^'		'H'		'2'	
'p'		'<'		'D'		'3'	
'*'		'>'		'd'		'4'	
'P'		'X'		'8'		'None'	ไม่มี
'+'	+	'x'	X	'.'			
'-'	-	' '		'.'	.		

กำหนดรูปแบบเส้นกราฟ

อักขระ	รูปแบบเส้น	อักขระ	รูปแบบเส้น
'-'	เส้นทึบ	'- !'	เส้นประจุด - สลับกับเป็นขีด
'- -'	เส้นประเป็นขีด	'!'	เส้นประเป็นจุด



KMITL
MASTER
CLASS
THE LEGEND 2021

กำหนดสีของเส้นกราฟ

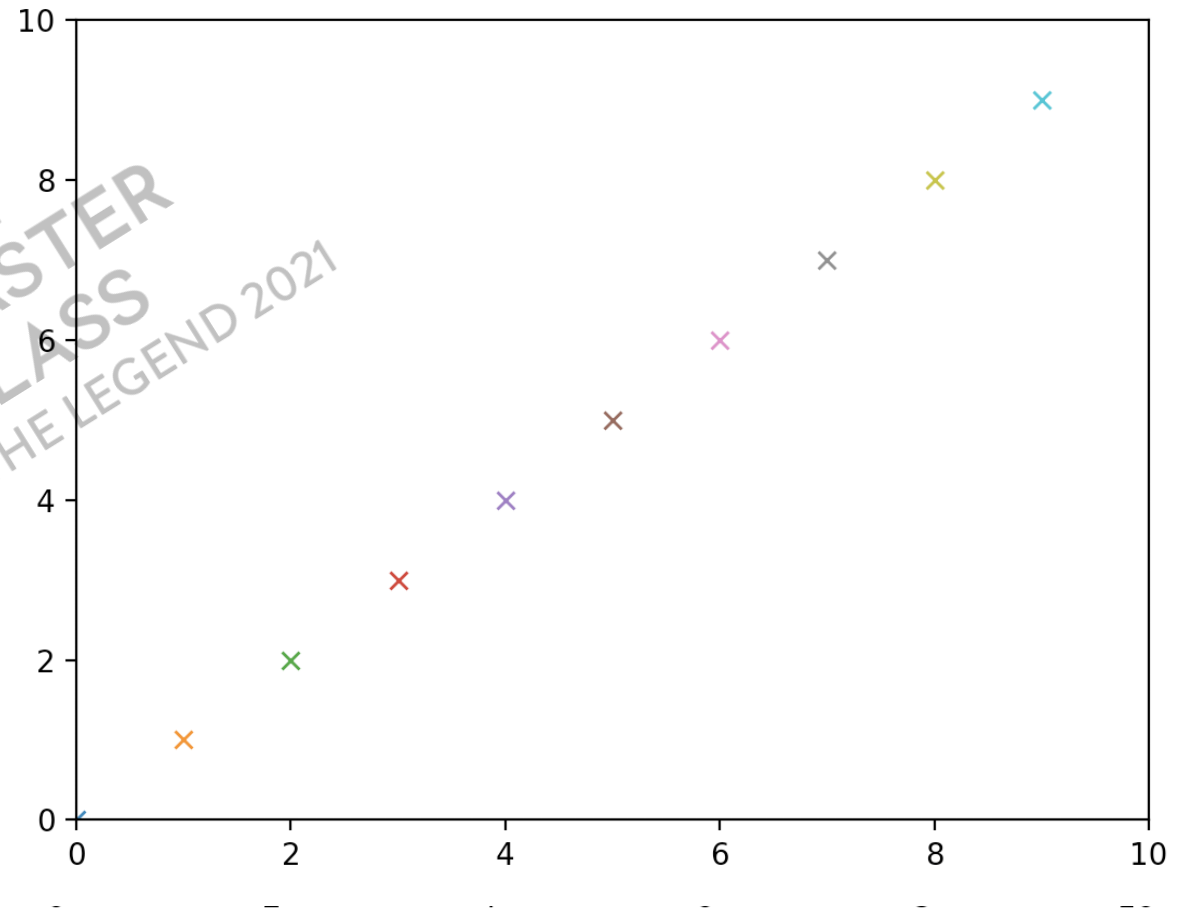
อักขระ	สี	อักขระ	สี	อักขระ	สี	อักขระ	สี
'b'	น้ำเงิน	'r'	แดง	'm'	มาเจนตา	'k'	ดำ
'g'	เขียว	'c'	ไซแอน	'y'	เหลือง	'w'	ขาว
'co'		'c1'		'c2'		'c→g'	

ฟังก์ชันวาดกราฟอย่างง่าย

```
import matplotlib.  
pl.axis([0,10,0,10])
```

```
for x in range(10):  
    y = x  
    pl.plot(x,y,'x')
```

```
pl.show()
```

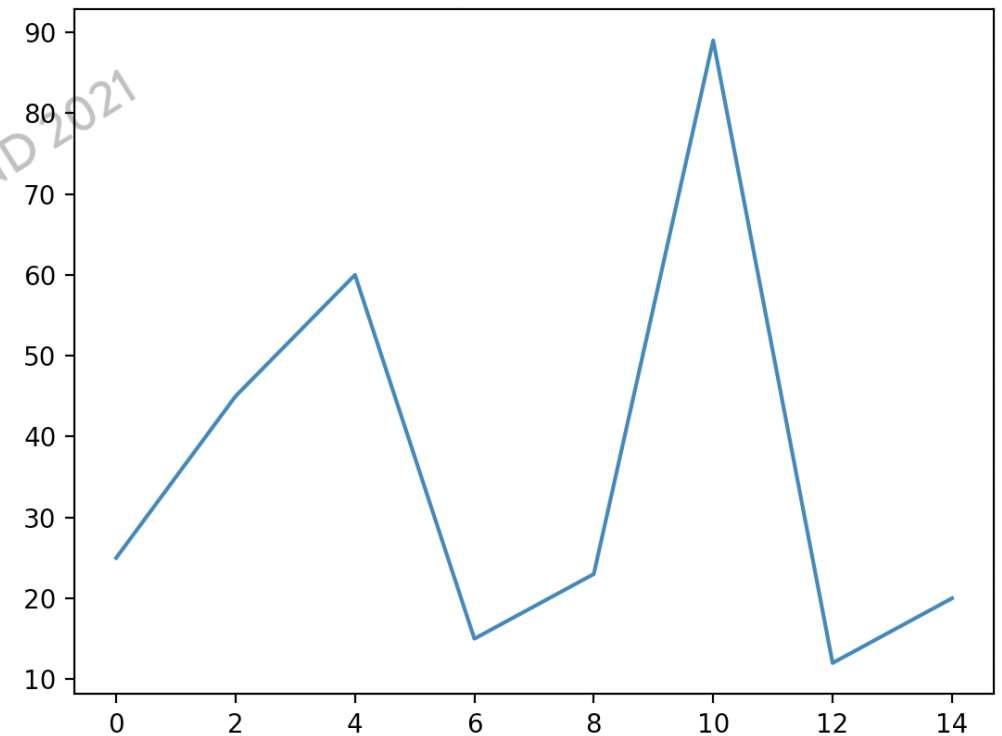


ฟังก์ชันวาดกราฟอย่างง่าย

```
import matplotlib.pyplot as pl
```

```
x = range(0, 16, 2)  
y = [25, 45, 60, 15, 23, 89, 12, 20]  
pl.plot(x,y)
```

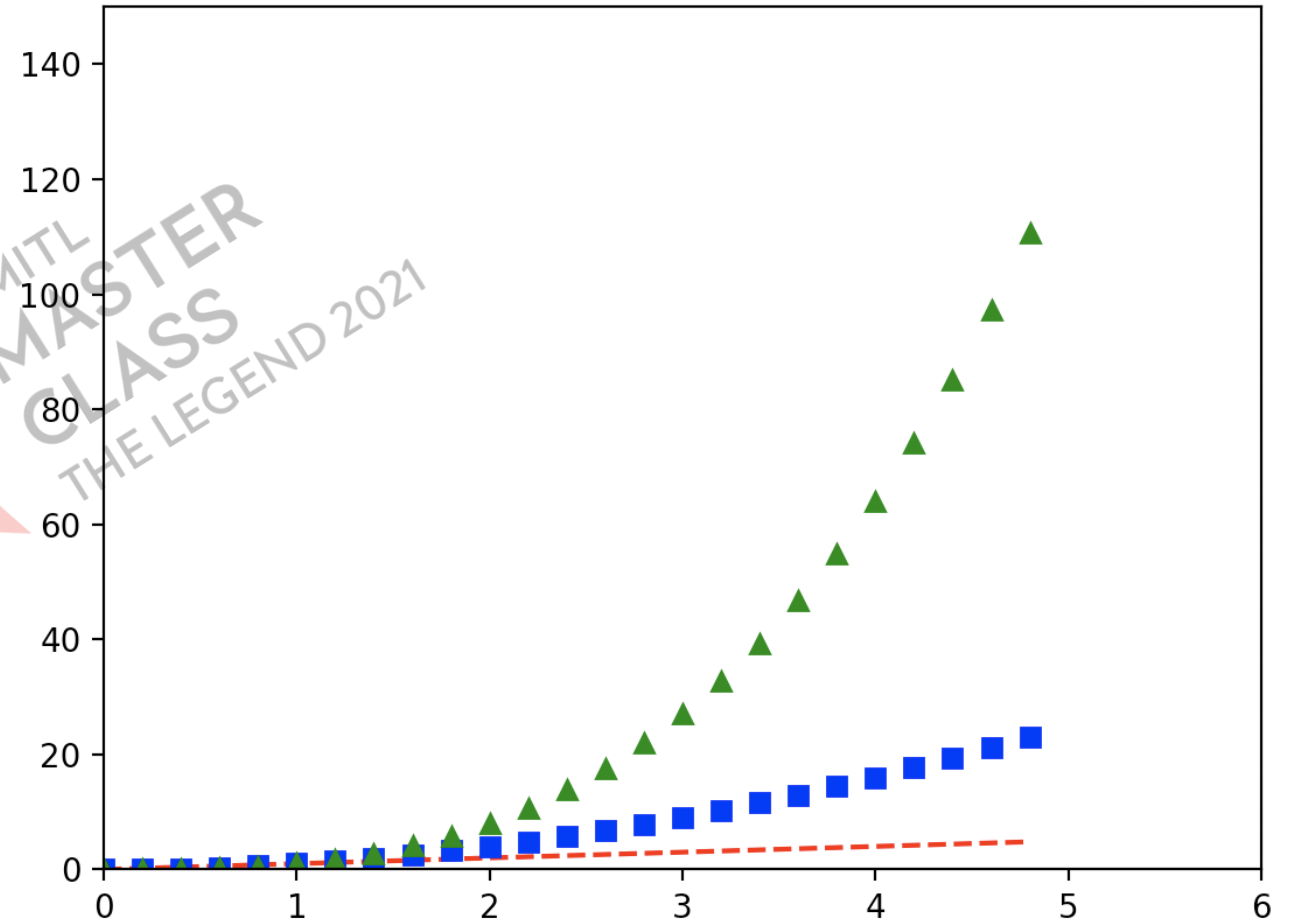
```
pl.show()
```



```
import matplotlib.pyplot as plt
import numpy as np
```

```
t = np.arange(0., 5., 0.2)
```

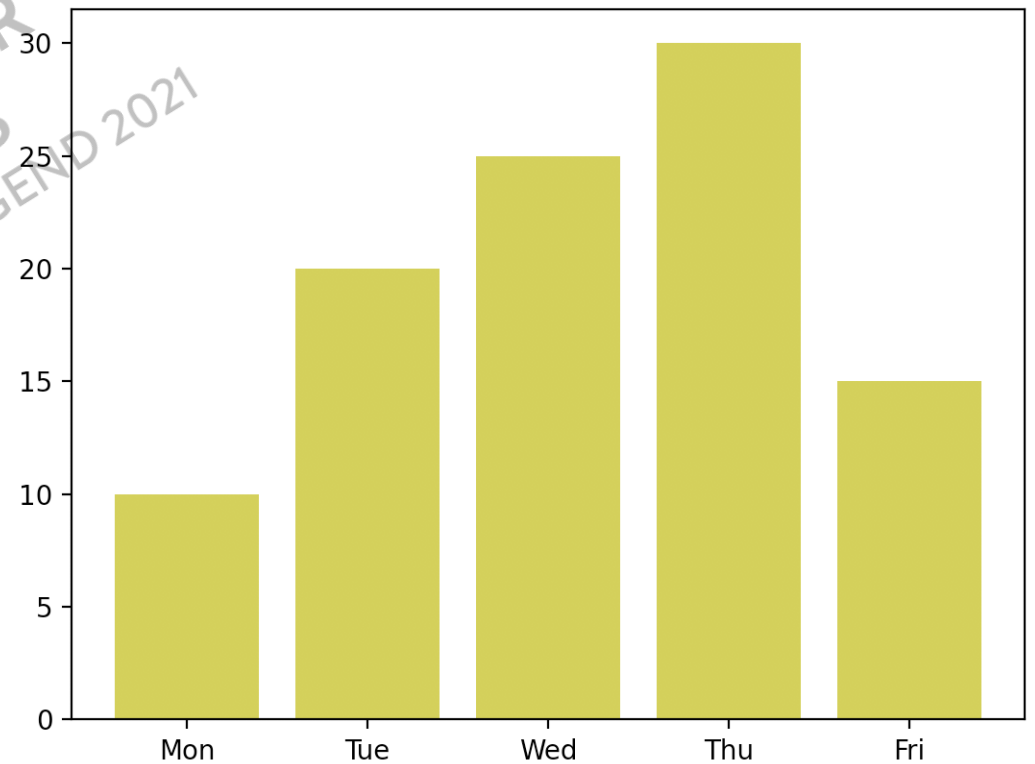
```
plt.plot(t,t,'r--',t,t**2,'bs',t,t**3,'g^')
plt.axis([0,6,0,150])
plt.show()
```



การวาดกราฟแท่ง

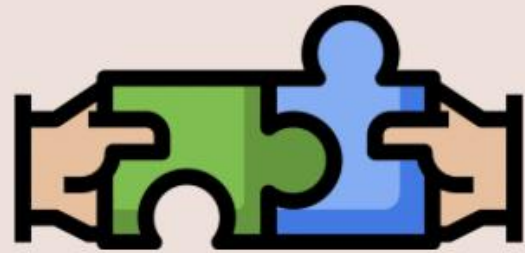
```
import matplotlib.pyplot as plt
x = ['Mon' , 'Tue' , 'Wed' , 'Thu' , 'Fri']
y = [10, 20, 25, 30, 15]

plt.bar(x,y,color = 'y' , alpha = 0.8)
plt.show()
```



โจทย์ปัญหา และ การประยุกต์

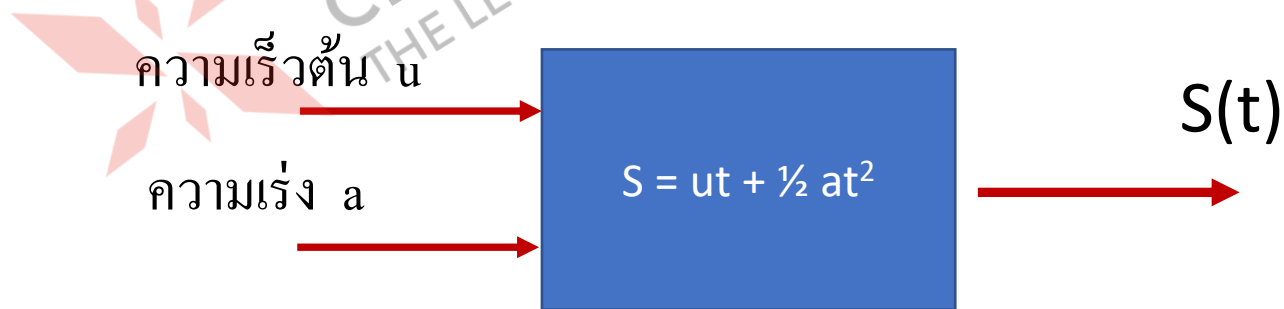
KMITL
MASTER
CLASS
THE LEGEND 2021



การเคลื่อนที่ ด้วยความเร่ง



รถยนต์คันหนึ่ง ผ่านจุด A ด้วยความเร็ว u m/s จากนั้นคนขับได้ใช้ความเร่ง a m/s² อยากทราบระยะทางห่างจากจุด A เมื่อเวลาผ่านไปทุก ๆ 2 วินาที จนถึงวินาทีที่ 20 รถมาถึงจุด B




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

```
x = [1,2,3,4,5,6,7,8,9,10]
```

```
y = [1,3,5,9,15,22,29,36,47,62]
```

```
plt.title('PROBLEMS')  
plt.xlabel('Time')  
plt.ylabel('Motion (m)')  
plt.plot(x,y)  
plt.show()
```

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

```
x = range(0,20,2)
```

```
y = []
```

```
def motion(u, a, t):
```

```
    S = u*t + (1/2)*a*t**2
```

```
    return S
```

```
u = int(input("ป้อนความเร็วต้นของรถ : "))
```

```
a = int(input("ป้อนความเร่งของรถ : "))
```

```
for t in range(0,20,2):
```

```
    S = motion(u, a, t);
```

```
    y.append(S)
```

```
    print('S(', t, ')= ', S)
```

```
plt.title("Motion")
```

```
plt.ylabel("S")
```

```
plt.xlabel("t")
```

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

```
plt.show()
```

ระยะเบรกของรถยนต์

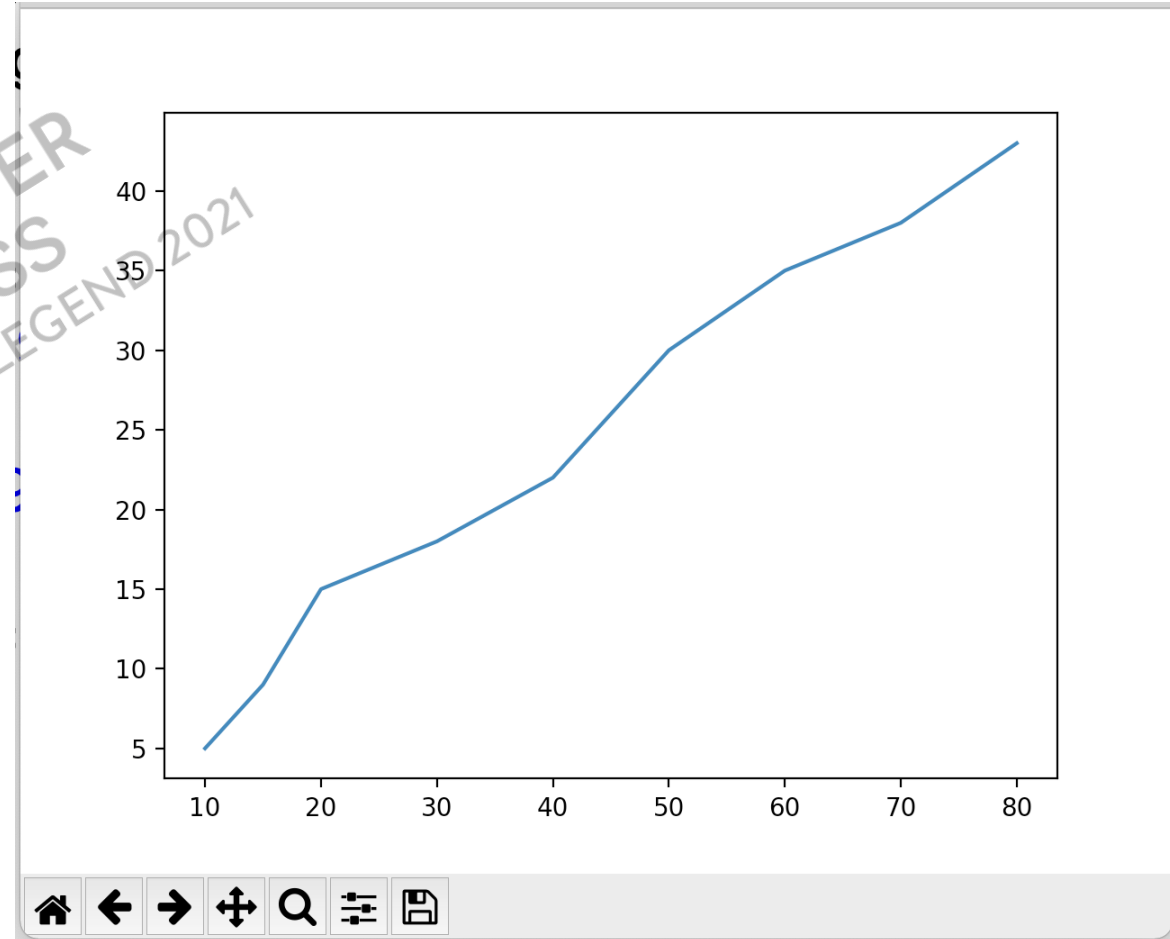
ความเร็ว km/h	10	15	20	30	40	50	60	70	80
ระยะหยุด m	5	9	15	18	22	30	35	38	43



MITL
MASTER
CLASS
THE LEGEND 2021

โปรแกรม

```
import matplotlib.pyplot as plt  
v = [10, 15, 20, 30, 40, 50, 60, 70, 80]  
s = [5, 9, 15, 18, 22, 30, 35, 38, 43]  
plt.plot(v,s)  
plt.show()
```



สมการ โพลีโนเมียล

- อันดับหนึ่ง

$$x + 7 = 0$$

```
>>> import numpy as np
>>> np.roots([1,7])
array([-7.])
```

- อันดับสอง

$$x^2 + 2x - 3 = 0$$

```
>>> import numpy as np
>>> np.roots([1,2,-3])
array([-3.,  1.])
```

สมการพหุนามเมื่อยลอันดับทั่วไป

$$f = x^3 - 4x^2 + x + 6 = 0$$

```
>>> import numpy as np
>>> np.roots([1, -4, 1, 6])
array([ 3.,  2., -1.]
```

KMITL
MASTER
CLASS
THE LEGEND 2021

MATRICES

#solving linear equation

```
>>> from numpy.matlib import
>>> A = matrix('1.0 2.0')
>>> A = matrix('1.0 2.0; 3.0 4.0')
>>> A
matrix([[1., 2.],
        [3., 4.]])
>>> A.T
matrix([[1., 3.],
        [2., 4.]])
>>> X = matrix('5.0 7.0')
>>> Y = X.T
>>> solve(A, Y)
matrix([[ -3.],
        [ 4.]])
>>> |
```

การแก้ระบบสมการเชิงเส้น

```
import numpy as np
```

```
A = np.array([[5, -4, 2],[3, 2, -1],[1, -2, 3]])
```

```
B = np.array([3, 4, 6])
```

```
X = np.linalg.solve(A,B)
```

```
print(X)
```

$$\begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} 3 \\ 4 \\ 6 \end{Bmatrix}$$

คำตอบ

```
[1. 2. 3.]
```

```
>>> |
```

$$[A]\{X\}=\{B\}$$

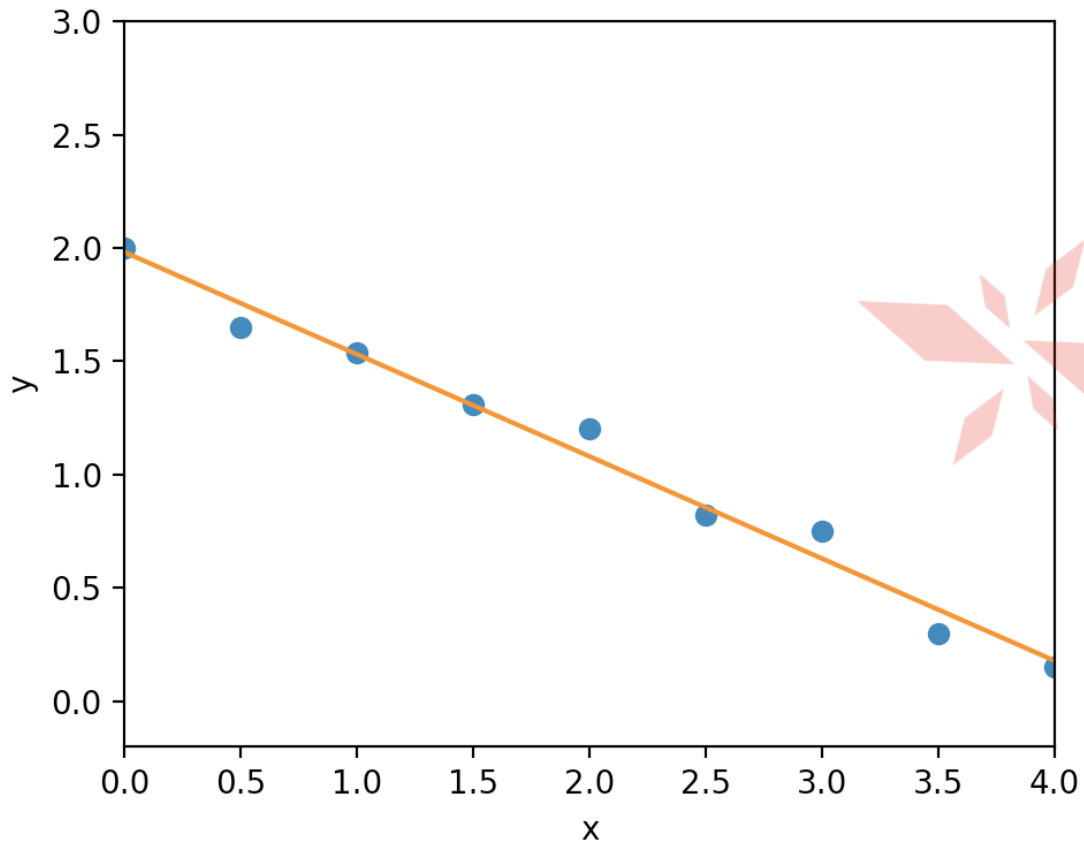


การพิตสมการ แบบเชิงเส้น

X	0	0.5	1	1.5	2	2.5	3	3.5	4
Y	2.0	1.65	1.54	1.31	1.2	0.82	0.75	0.3	0.15

$$y = -0.4507x + 1.981$$

KMITL
MASTER
CLASS
THE LEGEND 2021



KMITL
MASTER
CLASS
THE LEGEND 2021

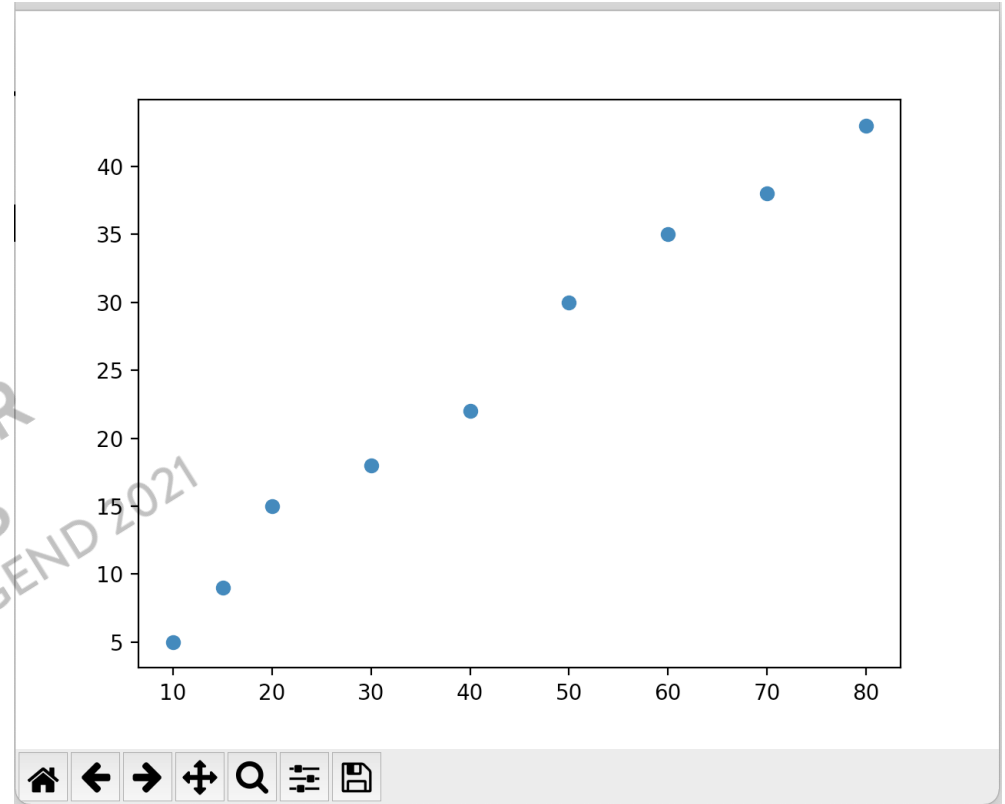
$[-0.45066667 \ 1.98133333]$

$-0.4507 x + 1.981$

\ggg

โปรแกรม

```
import numpy as np
from matplotlib.pyplot import *
v = [10, 15, 20, 30, 40, 50, 60, 70, 80]
s = [5, 9, 15, 18, 22, 30, 35, 38, 43]
coefficients = np.polyfit(v,s,1)
polynomial = np.poly1d(coefficients)
ys = polynomial(v)
print(coefficients)
print(polynomial)
ys = polynomial(v)
plot(v,s,'o')
show()
```



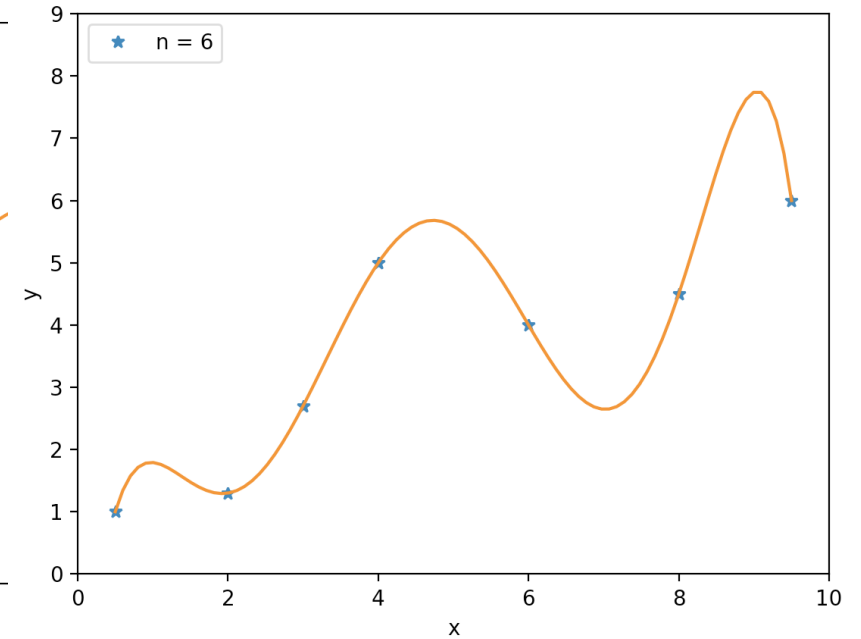
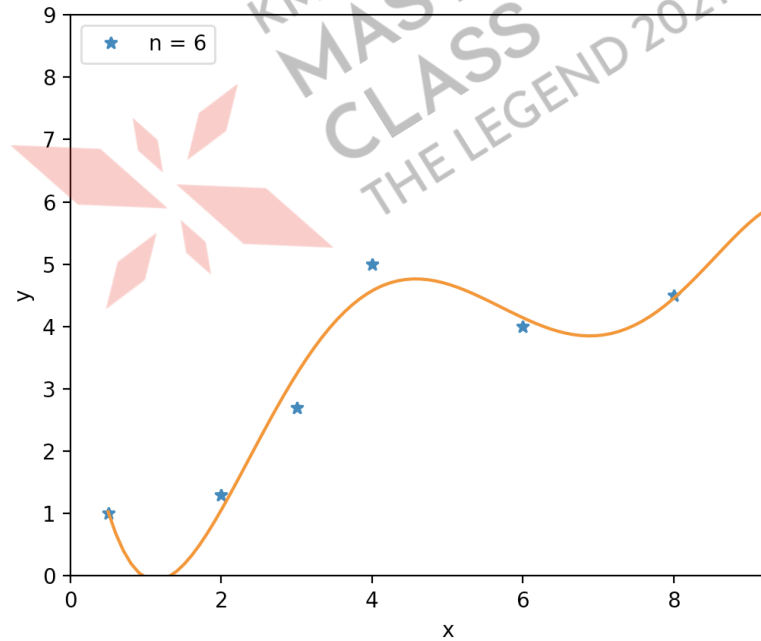
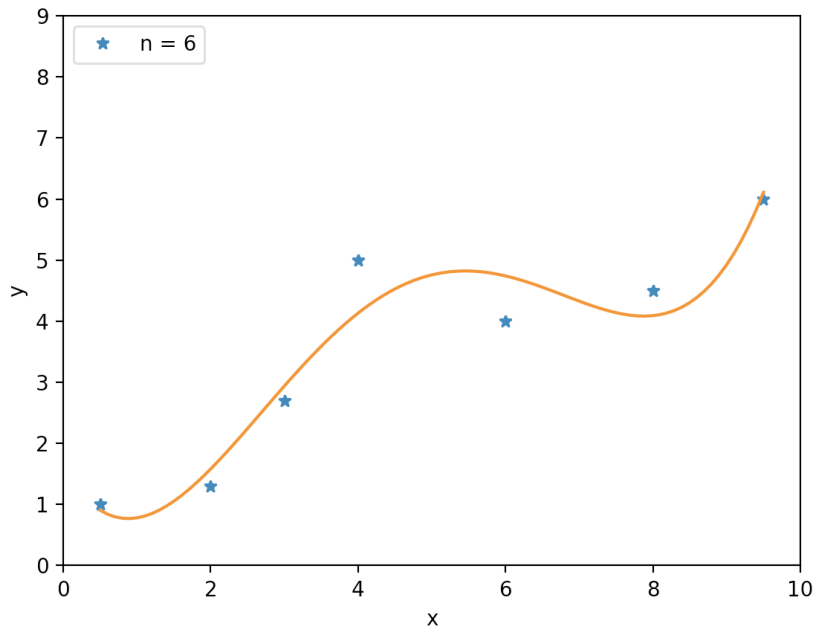
```
==== RESTART: /Volumes/Upd
[0.52933333 1.83333333]
```

```
0.5293 x + 1.833
```

```
>>> |
```

การฟิตสมการแบบเชิงเส้น

X	0.5	2	3	4	6	8	9.5
Y	1	1.3	2.7	5	4	4.5	6



เลือกใช้ฟังก์ชัน `polyfit`

```
import numpy as np
import matplotlib.pyplot as plt
x = [0.5,2,3,4,6,8,9.5]
y = [1,1.3,2.7,5,4,4.5,6]
coefficients = np.polyfit(x, y, 6)
polynomial = np.poly1d(coefficients)
xs = np.linspace(0.5,9.5,90)
ys = polynomial(xs)
plt.plot(x,y, '*')
plt.plot(xs,ys)
plt.legend(['n = 6'],loc = 2)
plt.xlabel('x');plt.ylabel('y')
plt.xlim(0,10);plt.ylim(0,9)
plt.show()
```

`np.polyfit(x , y , n)`



KMUTT
MASTER
CLASS
THE LEGEND 2021

โครงการ Python



โมดูลเต๋าไพ ทอน



KMITL
MASTER
CLASS
THE LEGEND 2021

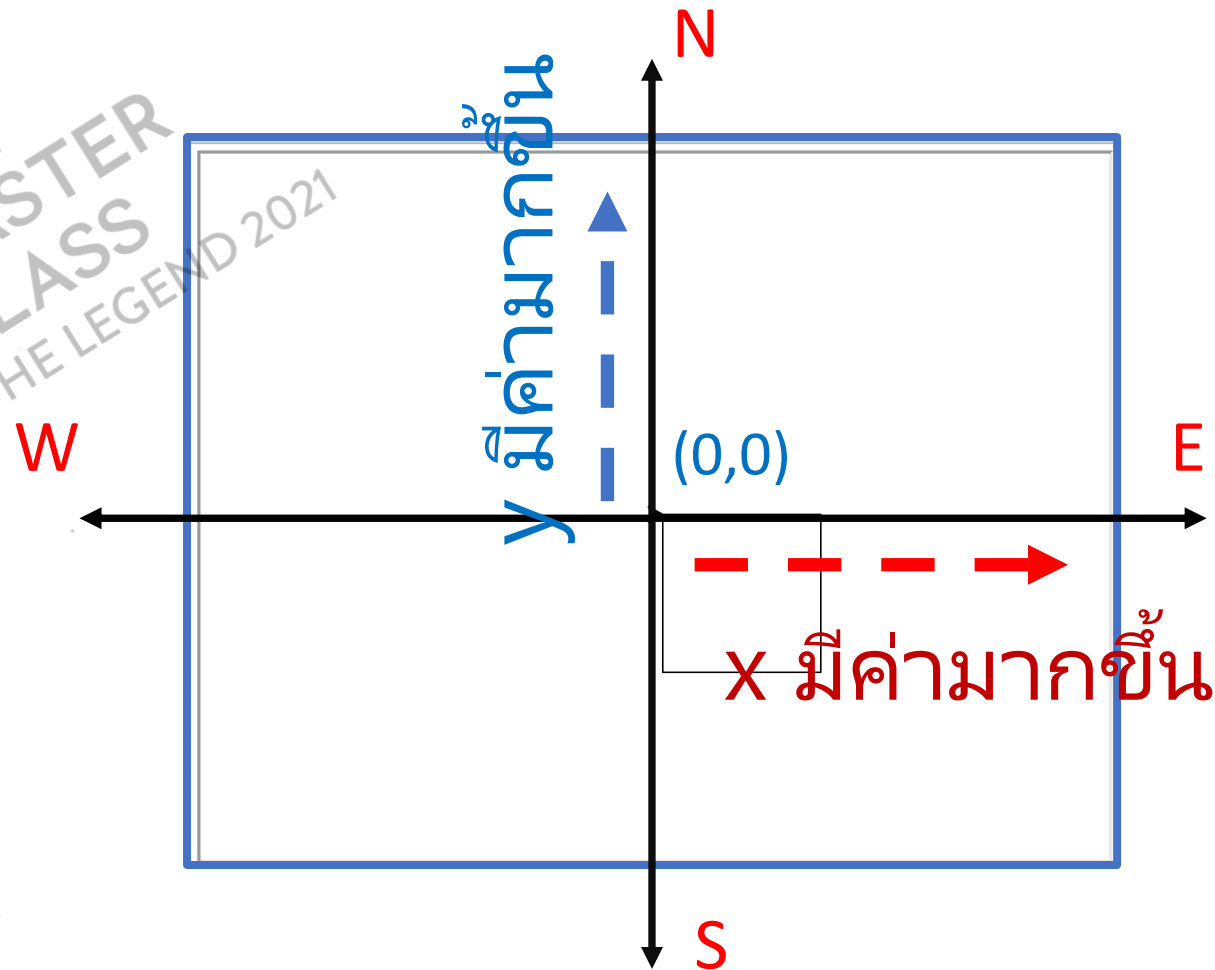
ธีรวัฒน์ ประกอบผล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



ลองสร้างกราฟิก ง่าย ๆ

```
from turtle import *  
for i in range(4):  
    forward(100)  
    right(90)
```



ส่วนประกอบของเต๋าไพทอน

1. พื้นที่วาดรูปหรือกระดาน (TurtleScreen)
2. อุปกรณ์ที่ใช้วาดรูป (RawTurtle)



เมธอดควบคุมการเคลื่อนที่ (Moving)

- ควบคุมการจรดปากกา แล้วลากเส้น

```
from turtle import *
forward(200)
sety(100)
setx(300)
```

เมธอด	การประมวลผล
-------	-------------

forward(ระยะ)	ไปข้างหน้าตามแนวลูกศร โดยมีระยะเป็นจำนวนจุด
---------------	---

backward(ระยะ)	ไปถอยหลังตามแนวลูกศร โดยมีระยะเป็นจำนวนจุด
----------------	--

```
from turtle import *
```

```
forward(200)
```

```
left(90)
```

```
forward(100)
```

```
right(90)
```

```
forward(100)
```

วาดสี่เหลี่ยมด้วยมุมที่กำหนด

วาดสี่เหลี่ยมด้วยมุมที่กำหนด

setx(x), y ใหม่

(0, 0)

(200, 100) (300, 100)

(200, 0)

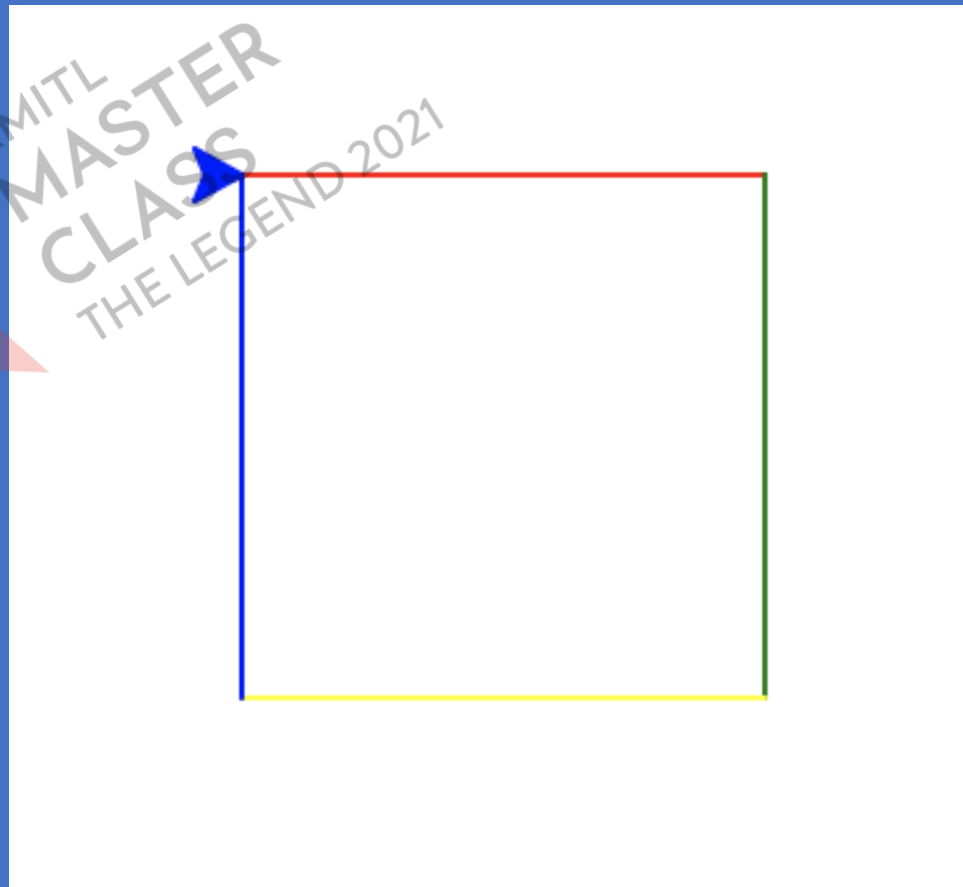
setx(x)	วาดเส้นไปยังตำแหน่ง x ใหม่ โดยค่า y เป็นค่าเดิม
---------	---

sety(y)	วาดเส้นไปยังตำแหน่ง y ใหม่ โดยค่า x เป็นค่าเดิม
---------	---

เมธอดควบคุม

เมธอด	การประมวลผล
<code>pensize(size)</code>	กำหนดขนาดของปากกาด้วย
<code>pencolor(สี)</code>	กำหนดสีของปากกา ใช้ได้ 3
<code>clear()</code>	ลบสิ่งที่วาดออกไปทั้งหมด
<code>penup()</code>	ยกปากกาขึ้นออกจากพื้นที่
<code>pendown()</code>	จรดปลายปากกาลงไปเพื่อวาด
<code>reset()</code>	ลบทั้งหมดแล้วกลับไปยังจุดเ
<code>speed(s)</code>	กำหนดความเร็วในการวาด
<code>bgcolor(สี)</code>	กำหนดสีพื้น

```
from turtle import *  
for c in ['red', 'green', 'yellow', 'blue']:  
    color(c)  
    forward(100)  
    right(90)
```



เมธอดวาดรูป



circle(r) วาดรูปวงกลมรัศมี r



fillcolor(สี) เทสีลงไปในรูปแบบ



end_fill() หยุดเทสี



write("ข้อความ", การเคลื่อนที่, ตำแหน่ง, รูปแบบตัวอักษร) ใช้เขียนข้อความ



onclick(ฟังก์ชัน) ตรวจสอบการคลิกเมาส์



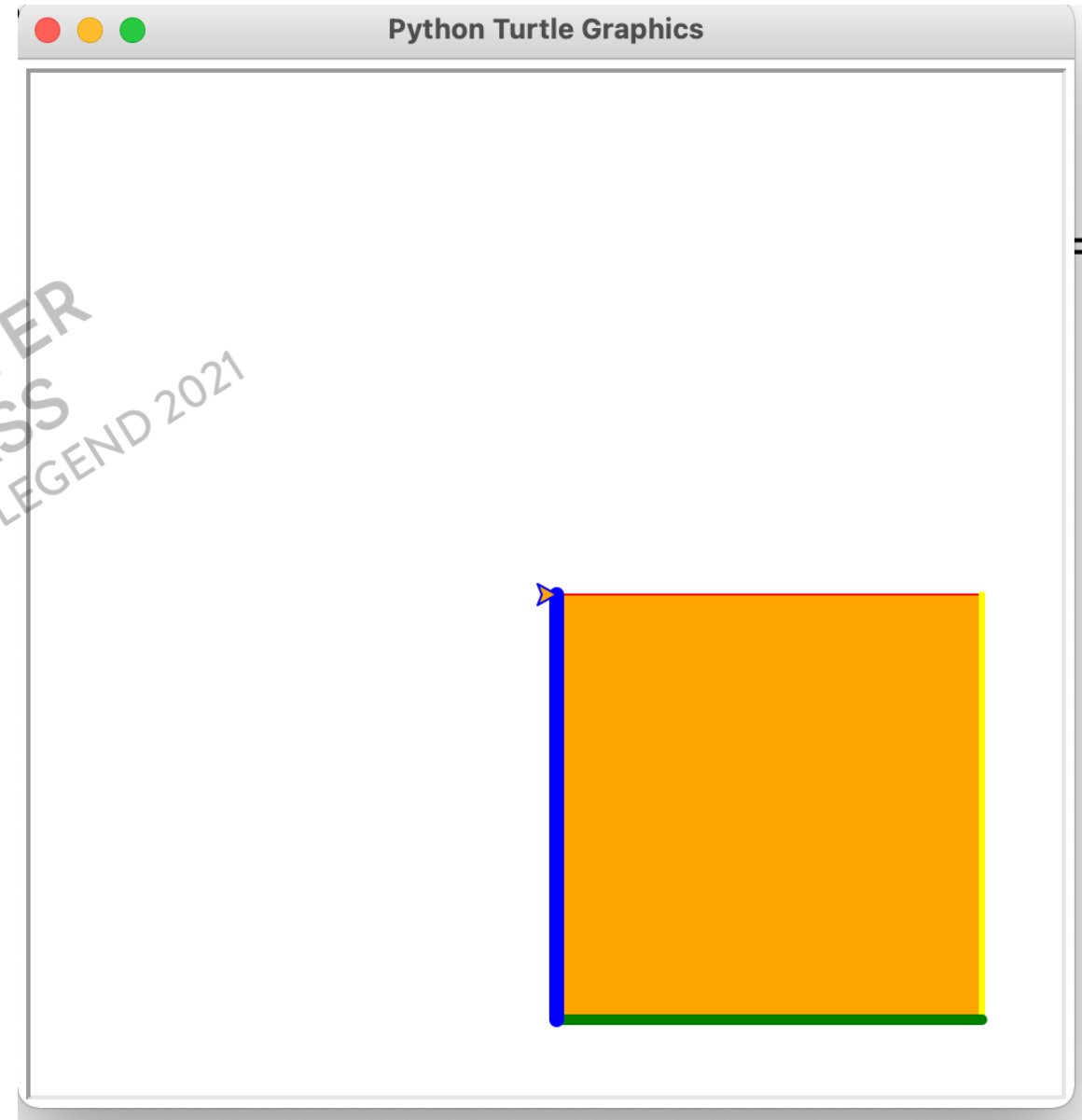
ondrag(ฟังก์ชัน) ตรวจสอบการลากเมาส์

```
import turtle as pen
s_color = ['red', 'yellow', 'green', 'blue']
```

```
for i in range(4):
    pen.pencolor(s_color[i])
    pen.pensize(i*2+1)
    pen.forward(200)
    pen.right(90)
```

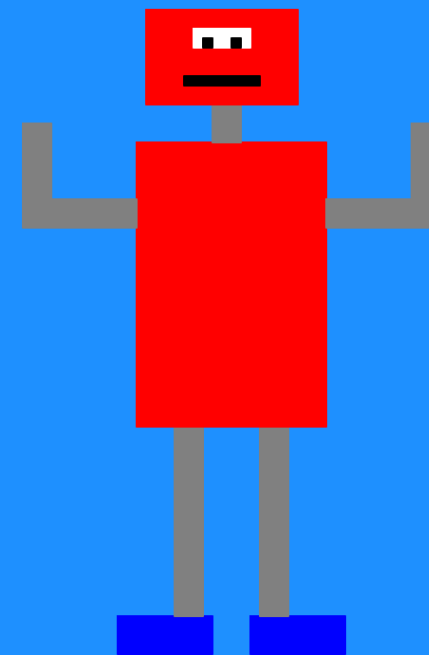


KMITL
MASTER
CLASS
THE LEGEND 2021

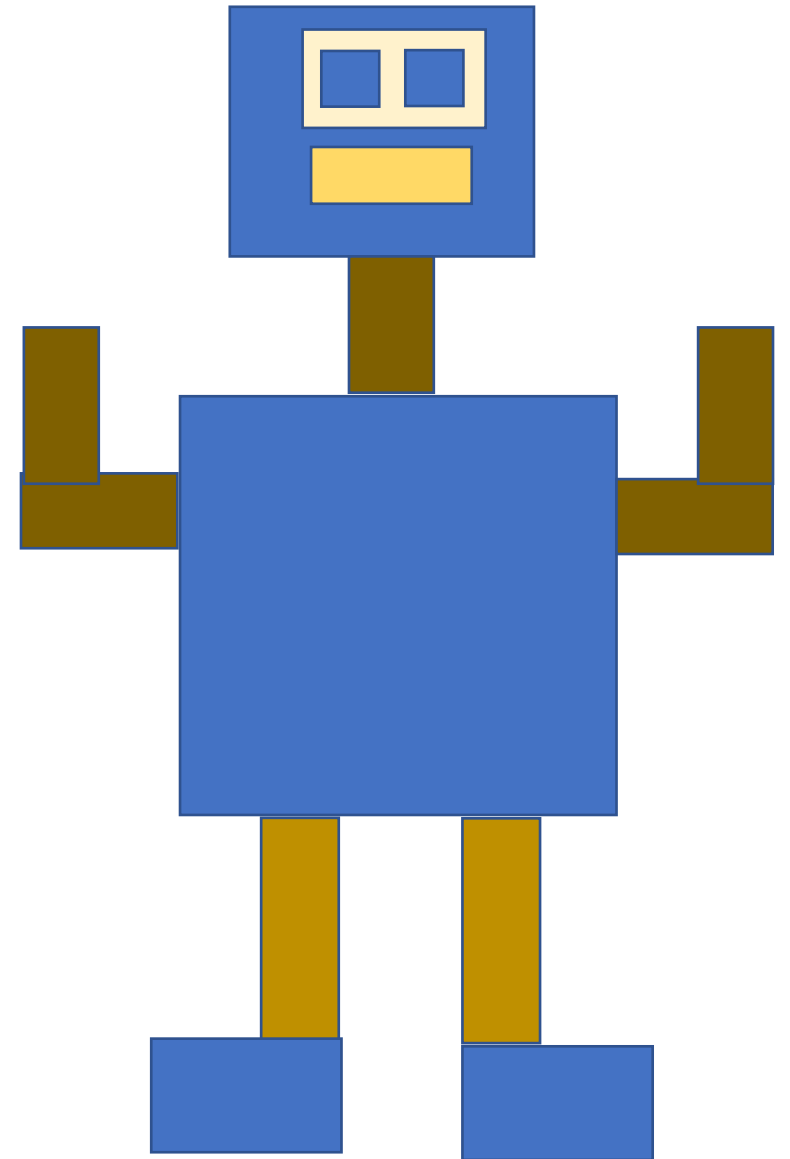
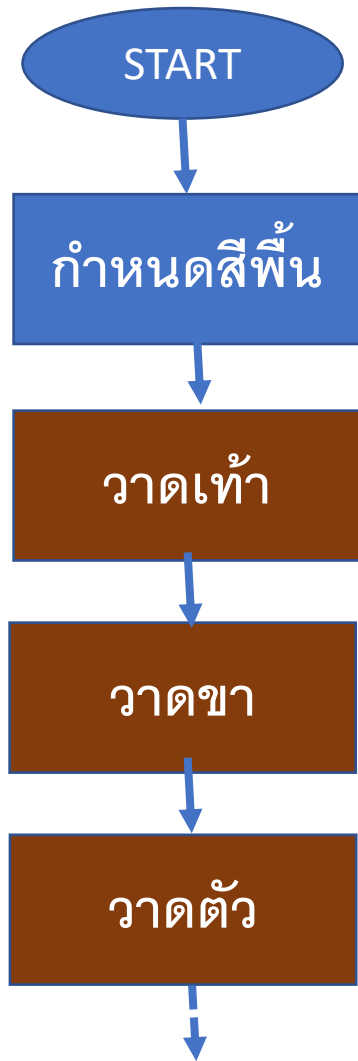


ลองวาดรูป
ดังต่อไปนี้

KMITL
MASTER
CLASS
THE LEGEND 2021

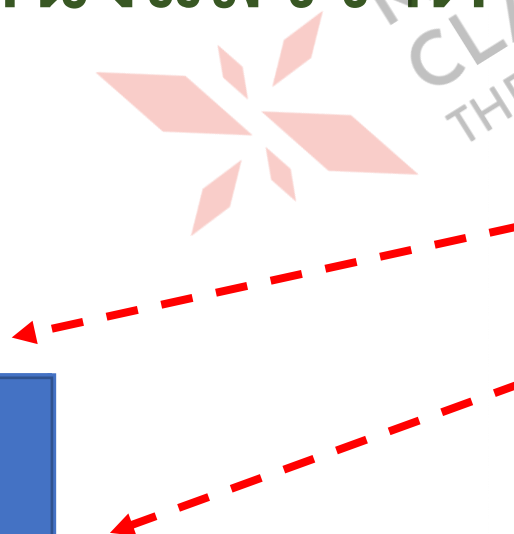


Robot Builder



วิธีการพัฒนาโปรแกรม

- สร้างฟังก์ชันวาดสี่เหลี่ยม
- กำหนดตำแหน่งแล้ววาด



```
import turtle as t
def rectangle(horizontal, vertical, color):
    t.pendown()
    t.pensize()
    t.color(color)
    t.begin_fill()
    for counter in range(1,3):
        t.forward(horizontal)
        t.right(90)
        t.forward(vertical)
        t.right(90)
    t.end_fill()
    t.penup()
```



**KMITL
MASTER
CLASS
THE LEGEND 2021**

