

# คลาสของไพ่

ข้างล่างนี้แสดงการเรียกใช้คลาส Card ซึ่งแทนไพ่ 1 ใบ ประกอบด้วย ค่าของไพ่ (value) ซึ่งเป็นสตริง ('A', '2', '3', ..., '10', 'J', 'Q', 'K') และ ดอกของไพ่ (suit) ซึ่งเป็นสตริงเช่นกัน ('club', 'diamond', 'heart', 'spade') โปรแกรมข้างล่างนี้รับไพ่เข้ามา แล้วไปมาสร้างเป็นลิสต์ของไพ่ (cards) และมีการเรียกใช้เมทอดต่าง ๆ ของคลาส Card ให้นิสิตเขียนเมทอดต่าง ๆ ของคลาส Card ให้สมบูรณ์

```
class Card:
    def __init__(self, value, suit):
        # ???

    def __str__(self):
        # ???

    def getScore(self):
        # ???

    def sum(self, other):
        # ???

    def __lt__(self, rhs):
        # ???

n = int(input())
cards = []
for i in range(n):
    value, suit = input().split()
    cards.append(Card(value, suit))
for i in range(n):
    print(cards[i].getScore())
print("-----")
for i in range(n-1):
    print(Card.sum(cards[i], cards[i+1]))
print("-----")
cards.sort()
for i in range(n):
    print(cards[i])
```

รายละเอียดต่าง ๆ ของคลาส Card และเมทอดของคลาส Card

- เมทอด `getScore` จะคืนค่าคะแนนของไพ่เป็นจำนวนเต็ม ตามกฎดังนี้
  - ไพ่ที่มีค่า A จะมีคะแนน 1 คะแนน
  - ไพ่ที่มีค่า 2 ถึง 10 จะมีคะแนนเท่ากับค่าของไพ่ คือ 2 ถึง 10 คะแนน ตามลำดับ
  - ไพ่ที่มีค่า J, Q, K จะมีคะแนน 10 คะแนน
- เมทอด `sum` จะคืนค่ารวมคะแนนของไพ่สองใบและ mod ด้วย 10 เช่น
  - `Card.sum(Card('7', 'club'), Card('2', 'heart'))` ได้ผลลัพธ์เป็น 9
  - `Card.sum(Card('J', 'spade'), Card('5', 'diamond'))` ได้ผลลัพธ์เป็น 5
- การเรียงลำดับของไพ่เป็นดังนี้
  - ค่าของไพ่เรียงตามลำดับดังนี้  $3 < 4 < 5 < \dots < 10 < J < Q < K < A < 2$
  - ดอกของไพ่เรียงตามลำดับดังนี้ `club < diamond < heart < spade`
  - ถ้าไพ่สองใบมีค่าไม่เท่ากัน ไพ่ที่มีค่ามากกว่าจะเป็นไพ่มากกว่า
  - ถ้าไพ่สองใบมีค่าเท่ากัน ไพ่ที่มีดอกสูงกว่าจะเป็นไพ่มากกว่า

#### ► ข้อมูลนำเข้า

บรรทัดแรกมีจำนวนเต็ม  $n$  แทนจำนวนไพ่ที่จะรับเข้ามา

$n$  บรรทัดต่อมา แต่ละบรรทัดมีค่าและดอกของไพ่แต่ละใบ คั่นด้วยช่องว่าง

#### ► ข้อมูลส่งออก

มี  $3n+1$  บรรทัด

$n$  บรรทัดแรก แสดงคะแนนของไพ่แต่ละใบ ตามด้วยขีดคั่น 1 บรรทัด

$n-1$  บรรทัดถัดมา แสดงคะแนนรวมของไพ่ 2 ใบที่ติดกันในลำดับ ตามด้วยขีดคั่น 1 บรรทัด

$n$  บรรทัดสุดท้าย แสดงไพ่เรียงตามลำดับ

#### ► ตัวอย่าง

Input (จากแป้นพิมพ์)	Output (ทางจอภาพ)
5	1
A spade	10
K heart	10
K club	7
7 diamond	2
2 spade	-----
	1
	0
	7
	9
	-----
	(7 diamond)
	(K club)
	(K heart)
	(A spade)
	(2 spade)

# กระปุกออมสิน

จงเขียนคลาส piggybank เพื่อผลิตกระปุกออมสิน ที่สามารถหยอดเหรียญมูลค่าอะไรก็ได้ โดยจำกัดจำนวนเหรียญรวมทุกแบบในกระปุกแล้วห้ามเกิน 100 เหรียญ (ถ้าหยอดแล้วเกินไม่รับเพิ่ม) ตามโครงของคลาสและตัวอย่างการใช้งานข้างล่างนี้

โครงของคลาส piggybank	ตัวอย่างการใช้งาน piggybank
<pre>class piggybank:     def __init__(self):         # มีตัวแปร self.coins เก็บ dict เริ่มต้นให้ว่าง ๆ         # มี key เป็นมูลค่าเหรียญ และ value เป็นจำนวนเหรียญ      def add(self, v, n):         # ถ้าเพิ่มจำนวนเหรียญในกระปุกอีก n เหรียญแล้วเกิน 100         # จะไม่ให้เพิ่ม ให้คืน False แทนว่า เพิ่มไม่สำเร็จ         # แปลง v เป็น float ก่อน (เพิ่ม 5 กับ 5.0 จะได้เหมือนกัน)         # ถ้ากระปุกไม่เคยมีเหรียญ v ทำ self.coins[v] = 0         # ทำคำสั่ง self.coins[v] += n         # คืน True แทนว่าเพิ่มสำเร็จ      def __float__(self):         # นำค่าของเหรียญคูณกับจำนวนเหรียญ ของเหรียญทุกแบบ         # ต้องคืนจำนวนแบบ float เท่านั้น อยากรู้คืนศูนย์ ก็ต้อง 0.0      def __str__(self):         # คืนสตริงที่แสดงจำนวนเหรียญแต่ละแบบตามตัวอย่าง         # โดยให้เรียงเหรียญตามมูลค่าจากน้อยไปมาก</pre>	<pre>p1 = piggybank() print( float(p1) )    # 0.0 p1.add(0.25, 4)     # เพิ่มเหรียญ 25 สตางค์ 4 เหรียญ print( float(p1) )    # 1.0 p1.add(0.50, 1)     # เพิ่มเหรียญ 50 สตางค์ 1 เหรียญ print( float(p1) )    # 1.5 p1.add(10, 1)     # เพิ่มเหรียญ 10 บาท 1 เหรียญ print( float(p1) )    # 11.5 print( p1 )     # {0.25:4, 0.5:1, 10.0:1} print(p1.add(10, 1))     # True เพิ่มได้ print( float(p1) )    # 21.5 print(p1.add(1,94))     # False เพิ่มไม่ได้ เกิน 100 เหรียญ print( float(p1) )    # 21.5</pre>

เมทอด `__float__` ถูกเรียกเมื่อ `float(p)` ทำงาน โดยที่ `p` เป็น `piggybank` ได้ผลลัพธ์เป็น `float` แทนค่าของ `p`  
เมทอด `__str__` ถูกเรียกเมื่อ `str(p)` ทำงาน โดยที่ `p` เป็น `piggybank` ได้ผลลัพธ์เป็นสตริงแทนค่าของ `p`

#### ► การส่งตรวจ

ให้นำโปรแกรมข้างล่างนี้ ต่อท้าย `class piggybank` ที่เขียนข้างบนนี้ แล้วจึงส่งให้ Grader ตรวจ

```
cmd1 = input().split(';')
cmd2 = input().split(';')
p1 = piggybank(); p2 = piggybank()
for c in cmd1: eval(c)
for c in cmd2: eval(c)
```

#### ► ข้อมูลนำเข้า

คำสั่งต่าง ๆ เพื่อการทดสอบคลาส

#### ► ข้อมูลส่งออก

ผลการทำงานของโปรแกรมข้างบนที่อาศัยคลาส `piggybank`

#### ► ตัวอย่าง

Input (จากแป้นพิมพ์)	Output (ทางจอภาพ)
<code>p1.add(1.11,2); print(float(p1), p1)</code> <code>print(float(p2), p2)</code>	2.22 {1.11:2} 0.0 {}
<code>p1.add(0.25,1);p1.add(5,1);p1.add(0.25,2);p1.add(5.0,1)</code> <code>print(float(p1), str(p1))</code>	10.75 {0.25:3, 5.0:2}
<code>p1.add(0.25,1); print(p1.add(0.25,100))</code> <code>print(p1.add(0.25,99)); print(float(p1))</code>	False True 25.0

CONFIDENTIAL



## ดร. สุกชัย สุตันทวิบูลย์

Intania 63

Principal Member of Technical Staff  
Advanced Micro Devices Inc. (AMD)  
California, USA

การใช้คอมพิวเตอร์ช่วยออกแบบ (Computer Aid Design : CAD) มีความสำคัญมากสำหรับการออกแบบ System on Chips ซึ่งประกอบด้วยทรานซิสเตอร์หลายพันล้านตัว วิศวกรออกแบบ chips ที่ประสบความสำเร็จส่วนใหญ่มีความรู้ความสามารถในการใช้โปรแกรม CAD และสามารถเขียนโปรแกรมเสริมเพื่อให้ตนเองทำงานออกแบบได้เร็วขึ้น การเขียนโปรแกรมให้ได้ดีจึงต้องเริ่มด้วยทักษะพื้นฐานด้าน Programming แล้วเสริมด้วยองค์ความรู้เรื่อง Algorithms และ Data Structures จึงจะทำให้เขียนโปรแกรมที่มีประสิทธิภาพ เช่น ทำงานได้เร็ว ใช้หน่วยความจำน้อย และใช้พลังงานต่ำ เป็นต้น

# 11 : Solutions to Exercises

## unth 1

### ► ข้อที่ 1

```
h = int(input())
m = int(input())
s = int(input())
total = 60*60*h + 60*m + s
print( total )
```

### ► ข้อที่ 2

```
import math
x = float(input())
y = 2 - x + 3/7*x**2 - 5/11*x**3 + \
    math.log(x,10)
print(y)
```

### ► ข้อที่ 3

```
a = float(input())
x = 1
x = (x + a/x)/2
x = (x + a/x)/2
x = (x + a/x)/2
x = (x + a/x)/2
print(x)
```

### ► ข้อที่ 4

```
v1,v2,v3 = [float(e) \
            for e in input().split()]
u1,u2,u3 = [float(e) \
            for e in input().split()]
dotp = v1*u1 + v2*u2 + v3*u3
print(dotp)
```

### ► ข้อที่ 5

```
import math
x1,y1,x2,y2 = \
    [float(e) for e in input().split()]
d = math.sqrt((x1-x2)**2+(y1-y2)**2)
print(d)
```

### ► ข้อที่ 6

```
import math
r,theta = [float(e) \
           for e in input().split()]
x = r*math.cos(theta)
y = r*math.sin(theta)
print(x, y)
```

### ► ข้อที่ 7

```
import math
x,y = [float(e) for e in input().split()]
r = math.sqrt(x**2+y**2)
theta = math.atan2(y,x)
print(r,theta)
```

### ► ข้อที่ 8

```
a,b,c,d,g = \
    [float(e) for e in input().split()]
avg = (a+b+c+d+g)/5
print(avg)
```

### ► ข้อที่ 9

```
a,b,c = [e for e in input().split()]
c = int(c)
out = a + b + str(c) + (a + b) * c
print(out)
```

## unth 2

### ► ข้อที่ 1

```
# ex. 3 4 1 --> median is 3
a,b,c = [int(e) for e in input().split()]
if b <= a <= c or c <= a <= b :
    print(a)
elif a <= b <= c or c <= b <= a :
    print(b)
else :
    print(c)
```

### ► ข้อที่ 2

```
x1,y1,r1 = [float(e) \
            for e in input().split()]
x2,y2,r2 = [float(e) \
            for e in input().split()]
d = (x1-x2)**2 + (y1-y2)**2
sumr2 = (r1+r2)**2
if d < sumr2 :
    print('overlap')
elif d == sumr2 :
    print('touch')
else :
    print('free')
```

### ► ข้อที่ 3

```
x,y = [float(e) \
      for e in input().split()]
if x == 0 and y == 0 :
    print('At the origin')
elif x == 0 :
    print('On y-axis')
elif y == 0 :
    print('On x-axis')
elif x > 0 and y > 0 :
    print('Quadrant I')
elif x < 0 and y > 0 :
    print('Quadrant II')
elif x < 0 and y < 0 :
    print('Quadrant III')
else :
    print('Quadrant IV')
```

### ► ข้อที่ 4

```
a,b,c,d,e = [int(e) \
             for e in input().split()]
if a <= b <= c <= d <= e :
    print('True')
else :
    print('False')

# print( a <= b <= c <= d <= e )
```

### ► ข้อที่ 5

```
a,b,c,d = [int(e) \
          for e in input().split()]
mx = a
if b > mx : mx = b
if c > mx : mx = c
if d > mx : mx = d
mn = a
if b < mn : mn = b
if c < mn : mn = c
if d < mn : mn = d
s = (a+b+c+d) - mx - mn
print(s)

# s = (a+b+c+d)-max(a,b,c,d)-min(a,b,c,d)
```

### ► ข้อที่ 6

```
a = int(input())
x = int(round(a**(1/3),0))
if x**3 == a :
    print(x)
else :
    print('Not Found')
```

### ► ข้อที่ 7

```
c = int(input())
if c < 37 :
    s = 'XS'
elif c < 41 :
    s = 'S'
elif c < 43 :
    s = 'M'
elif c < 46 :
    s = 'L'
else :
    s = 'XL'
print(s)
```

## unt 3

### ► ข้อที่ 1

```
k = 1
while 1/k*k == 1 :
    k += 1
print(k)
```

### ► ข้อที่ 2

```
k = 1
p = 1.0
while (1-p) < 0.5 :
    p *= (365-k)/365
    k += 1
print(k)
```

### ► ข้อที่ 3

```
p = 0.0
for k in range(1,400000,4):
    p += 1/k
    p -= 1/(k+2)
print(4*p)
```

### ► ข้อที่ 4

```
a,b = [int(e) for e in input().split()]
s = 0
for i in range(a,b):
    t = 0
    for j in range(i+1,b+1):
        t += (i+j)
    s += (-1)**i * t
print(s)
```

### ► ข้อที่ 5

```
a,b = [int(e) for e in input().split()]
s = 0
for i in range(a,b):
    for j in range(i+1,b+1):
        s += (-1)**i * (i+j)
print(s)
```

#### ► ข้อที่ 6

```
n = int(input())
a = int(input())
mn = mx = a
c = 0
if a < 0 : c = 1
for k in range(n-1):
    a = int(input())
    if a > mx : mx = a
    if a < mn : mn = a
    if a < 0 : c = 1
print( (mx, mn), c )
```

#### ► ข้อที่ 7

```
n = int(input())
for x in range(1,n+1):
    for y in range(x,n+1):
        for z in range(y,n+1):
            t = x**2+y**2+z**2
            w = int(round(t**(1/3),0))
            if w**3 == t:
                print(w,x,y,z)
```

## unที่ 4

#### ► ข้อที่ 1

```
s = input().strip()
t = ''
for e in s:
    t += e*2
print(t)
```

#### ► ข้อที่ 2

```
s = input().strip()
t = ''
s = ' ' + s + ' '
for i in range(1,len(s)-1) :
    t += s[i]
    if s[i-1] != s[i] != s[i+1] :
        t += s[i]
print(t)
```

#### ► ข้อที่ 3

```
s = input().strip()
if s == s[::-1] :
    print('Y')
else:
    print('N')
```

#### ► ข้อที่ 4

```
d,n = [int(e) for e in input().split()]
t = "0"*n + str(d)
t = t[-max(n,len(str(d))):]
print(t)
```

#### ► ข้อที่ 5

```
h = input().strip()
d = '0123456789ABCDEF'.find(h)
if d >= 0 :
    print(d)
else :
    print('invalid hex digit')
```

#### ► ข้อที่ 6

```
t = input().strip()
c = 0
for e in t :
    if e in '13579' :
        c += 1
print(c)
```

#### ► ข้อที่ 7

```
t = input().strip()
c = 0
for k in range(len(t)-1) :
    if t[k] in 'aeiou' and \
        t[k+1] in 'aeiou' :
        c += 1
print(c)
```

#### ► ข้อที่ 8

```
b = input().strip()
b = b[::-1]
d = 0
for i in range(len(b)) :
    d += int(b[i])*2**i
print(d)
```

## unที่ 5

#### ► ข้อที่ 1

```
file1 = open( input().strip() )
s = ''
for line in file1 :
    if line[-1] != '\n' : # กรณีบรรทัดสุดท้ายไม่มี \n
        line = line + '\n'
    s = line + s # นำบรรทัดใหม่มาต่อทางซ้าย
file1.close()
print(s[::-1]) # ลบ \n ที่บรรทัดสุดท้าย
```



### ► ข้อที่ 2

```
file1 = open( input().strip() )
s = ''
for line in file1 :
    if len(line.strip()) > 0 :
        if line[-1] != '\n' :
            line = line + '\n'
        s = line + s
file1.close()
out = open('reverse txt', 'w')
out.write(s[::-1])
out.close()
```

### ► ข้อที่ 3

```
file1 = open( input().strip() )
for line in file1 :
    a = line.find("<headline>")
    if a >= 0 :
        j = a+len("<headline>")
        b = line.find("</headline>",j)
        print( line[j:b] )
file1.close()
```

### ► ข้อที่ 4

```
file1 = open( input().strip() )
file2 = open( input().strip() )
for line1 in file1 :
    line2 = file2.readline()
    if line1 != line2 :
        print( False )
        break
else :
    print( len(file2.readline()) == 0 )
file1.close()
file2.close()
```

หรือแบบสั้น ๆ ใช้ readlines

```
file1 = open( input().strip() )
file2 = open( input().strip() )
print( file1.readlines() == \
        file2.readlines() )
```

## บทที่ 6.1

### ► ข้อที่ 1

```
v1 = [float(e) for e in input().split()]
v2 = [float(e) for e in input().split()]
if len(v1) != len(v2) :
    print( 'Error' )
else :
    dotp = 0
    for k in range(len(v1)) :
        dotp += v1[k]*v2[k]
    print(dotp)
```

### ► ข้อที่ 2

```
n = int(input())
d = []
for k in range(n):
    d.append( int(input()) )
d.sort()
t = []
for e in d :
    t.append(str(e))
print(','.join(t))
```

### ► ข้อที่ 3

```
file1 = open( input().strip() )
d = []
for line in file1:
    d.append( int(line) )
file1.close()
c = []
for e in d:
    c.append( d.count(e) )
maxc = max(c)
out = []
for k in range(len(d)):
    if c[k]==maxc and d[k] not in out :
        out.append(d[k])
for e in out :
    print(e)
```

### ► ข้อที่ 4

```
file1 = open( input().strip() )
h = []
for line in file1 :
    a = line.find("<headline>")
    if a >= 0 :
        j = a+len("<headline>")
        b = line.find("</headline>",j)
        h.append( line[j:b] )
file1.close()
h.sort()
for e in h :
    print(e)
```

### ► ข้อที่ 5

```
file1 = open( input().strip() )
d = []
for line in file1 :
    d.append( line.strip() )
file1.close()

for k in range(len(d)-1) :
    for i in range(len(d)-1) :
        if len(d[i]) > len(d[i+1]) or \
           len(d[i]) == len(d[i+1]) and \
           d[i] > d[i+1] :
            d[i],d[i+1] = d[i+1],d[i]

for e in d :
    print(e)
```

## ตอนที่ 6.2

### ► ข้อที่ 1

```
r,c = [int(e) for e in \
      input().split()]
m = []
for k in range(r) :
    m.append( [int(e) for e in \
              input().split()] )
    if len(m[k]) != c :
        m = [[]]
        break
print( m )
```

### ► ข้อที่ 2

```
infile = open(input().strip())
f = []
for line in infile :
    usernames = line.split()
    f.append( [usernames[0], \
              usernames[1:]] )
infile.close()
print( f )
```

### ► ข้อที่ 3

```
# สมมติว่ามี f มาแล้ว
nofollowers = []
for [username,followers] in f :
    if len(followers) == 0 :
        nofollowers.append( username )
print( 'No followers :', \
      ', '.join(nofollowers) )
```

### ► ข้อที่ 4

```
n = int(input())
d = []
for k in range(n)
    s = input().strip()
    d.append( [len(s),s] )
d.sort()
for [c,s] in d :
    print(s)
```

## ตอนที่ 6.3

### ► ข้อที่ 1

```
d = [ e.count(c) for e in x ]
```

### ► ข้อที่ 2

แบบแรก

```
x = [e for e in x if e >= 0]
```

คือการสร้างลิสต์ใหม่ที่ไม่มีเลขลบ แล้วให้ x อ้างอิงลิสต์ใหม่นี้ ถ้าเขียนอีกแบบ

```
x[:] = [e for e in x if e >= 0]
```

คือการสร้างลิสต์ใหม่ที่ไม่มีเลขลบ แล้วนำค่าในลิสต์ใหม่นี้ไปใส่ในที่เก็บเดิมของลิสต์ x สองวิธีนี้ให้ผลคล้ายกัน ต่างกันตรงที่ ถ้าก่อนหน้านี้มีการทำคำสั่ง  $y = x$  คือให้ y อ้างอิงลิสต์เดียวกับ x การทำแบบแรกจะทำให้ y อ้างอิงลิสต์เดิม และ x อ้างอิงลิสต์ใหม่ ในขณะที่แบบหลัง y ยังคงอ้างอิงลิสต์เดียวกับ x

### ► ข้อที่ 3

```
# t = [ [1,2,3], [33], [3,3,3,4] ]
t = sum( [ sum(e) for e in x ] )
```

### ► ข้อที่ 4

```
c = sum( [1 for e in input().split() \
          if int(e) < 0] )
```

### ► ข้อที่ 5

```
t = ''.join([e for e in input() \
            if 'a' <= e.lower() <= 'z'])
```

### ► ข้อที่ 6

```
x = [float(e) for e in input().split()]
y = [float(e) for e in input().split()]
z = [x[i]+y[i] for i in range(len(x))]
```

### ► ข้อที่ 7

```
f = [e for row in m for e in row]
```

### ► ข้อที่ 8

```
x = [int(e) for e in input().split()]
x.sort()
t = [x[i] for i in range(len(x)-1) \
     if x[i] != x[i+1] ]
t.append(x[-1])
```

เขียนแบบนี้ก็ได้ (แต่ช้ากว่าเยอะ)

```
x = [int(e) for e in input().split()]
t = []
for e in x :
    if e not in t :
        t.append(e)
```

### ► ข้อที่ 9

```
n = int(input())
x = [j for i in range(2, n//2) \
      for j in range(2*i, n, i)]
x.sort()
c = [x[i] for i in range(len(x)-1) \
      if x[i] != x[i+1] ]
c.append(x[-1])
```

### ► ข้อที่ 10

```
n = int(input())
x = [j for i in range(2, n//2) \
      for j in range(2*i, n, i)]
x.sort()
c = [x[i] for i in range(len(x)-1) \
      if x[i] != x[i+1] ]
c.append(x[-1])
p = [e for e in range(2,n) if e not in c]
```

## บทที่ 7

### ► ข้อที่ 1

- 1) dict มี key คือ studentID,  
value คือ grade
- 2) set ของ studentID
- 3) dict มี key คือ dept,  
value คือ set ของ studentID
- 4) list ของ phone
- 5) dict มี key คือ เลขท้าย TelNo,  
value คือ count

### ► ข้อที่ 2

```
x = int(input())
t = ()
for i in range(0,x,2):
    # คำสั่งข้างล่างนี้ไม่ได้เปลี่ยน tuple ของ t แต่สร้าง
    # tuple ใหม่ แล้วแทนที่ t ตัวเก่า
    t += (i,) # เหมือน t = t + (i,)
print(t)
หรือ
t = tuple([e for e in range(0,x,2)])
หรือ
t = tuple(range(0,x,2))
```

### ► ข้อที่ 3

```
x = int(input())
t = ()
while x > 0:
    t = (x%10,) + t
    x //= 10
print(t)
หรือ เปลี่ยนมารับสตริง, นำแต่ละหลักมาเปลี่ยนเป็น int,
เก็บใส่ list, แล้วส่งไปสร้างเป็น tuple
x = input().strip()
print(tuple([int(e) for e in x]))
```

### ► ข้อที่ 4

```
x = input().strip()
d = {}
for e in x:
    if e not in d: d[e] = 1
    else: d[e] += 1
print(d)
```

### ► ข้อที่ 5

```
x = input().strip()
y = input().strip()
set_x = set(x)
set_y = set(y)
print(set_x.intersection(set_y))
```

## บทที่ 8

### ► ข้อที่ 1

```
def f1(a,b):
    for i in range(b): print(a)
```

### ► ข้อที่ 2

```
def f2(a,b):
    return [a]*b
```

### ► ข้อที่ 3

```
def g(m,b,n,c):
    if m==n and b!=c: return 1
    if m==n and b==c: return 2
    x = (c-b)/(m-n)
    y = m*x + b
    return (x,y)
```

### ► ข้อที่ 4

```
def h(x):
    return [e for e in x if e%2==0]
```

### ► ข้อที่ 5

```
def a(n):
    if n==0: return 1
    if n==1: return -2
    return a(n-2)*n
```

### ข้อที่ 6

```
def k(n):
    if n==0: return 1
    if n==1: return 2
    if n%2==0:
        x = k(n//2)
        return x + x%10
    return k(n//2-1)*(n//2)
```

### ข้อที่ 7

```
def s(i,k):
    if i>=k: return 0
    return k + s(i+1,k)
def t(j,k):
    if j>=k: return 0
    return j + s(j,k-1)
```

### ข้อที่ 8

```
def is_palindrome(s):
    if len(s) <= 1: return True
    if s[0]!=s[-1]: return False
    return is_palindrome(s[1:-1])
```

## บทที่ 9

### ข้อที่ 1

```
k = int(input())
M[:, :k, :k] = 0
```

### ข้อที่ 2

แบบช้า

```
k = int(input())
for i in range(M.shape[0]):
    for j in range(M.shape[1]):
        if i%k==0 and j%k==0:
            M[i][j]*=2
```

แบบนี้ก็ช้าอยู่

```
k = int(input())
M = np.array([[2*M[i][j] if i%k==0 \
               and j%k==0 else M[i][j] \
               for j in range(M.shape[1])] \
               for i in range(M.shape[0])])
```

แบบนี้เร็ว

```
k = int(input())
N = np.zeros_like(M)
N[:, :k, :k] = 1
M += M*N
```

แบบนี้เร็วและสั้น

```
k = int(input())
M[:, :k, :k] *= 2
```

### ข้อที่ 3

```
MAX = np.max(M,axis=0)
MIN = np.min(M,axis=0)
A = MAX-MIN
```

### ข้อที่ 4

```
Y = (X[:,0]**2+X[:,1]**2)**0.5
```

### ข้อที่ 5

```
k = int(input())
C = np.zeros((k,k),int)
C[:,2, 1::2] = C[1::2, ::2] = 1
```

### ข้อที่ 6

```
k = int(input())
C = np.zeros((k,k),int)
C[:,2, ::2] = C[1::2, 1::2] = 1
C = (C*np.arange(1,k+1)).T
# ใช้ C*(np.arange(1,k+1).T) ไม่ได้(เพราะอะไร?)
```

## บทที่ 10

### ข้อที่ 1

```
def __init__(self, name, year, faculty):
    self.name = name
    self.year = year
    self.faculty = faculty
```

```
def __str__(self):
    return self.name + \
           ' (year ' + str(self.year) + ') ' + \
           self.faculty
```

```
def __lt__(self, rhs):
    if self.faculty != rhs.faculty:
        return self.faculty < rhs.faculty
    if self.year != rhs.year:
        return self.year < rhs.year
    return self.name < rhs.name
```

► ข้อที่ 2

```
def __init__(self, license, brand, color):
    self.license = license
    self.brand = brand
    self.color = color
    self.report = []

def __str__(self):
    return self.license + ' - ' + \
        + self.color + ' ' + self.brand

def __lt__(self, rhs):
    return self.license < rhs.license

def add_report(self, new_report):
    self.report.append(new_report)

def total_payment(self):
    return sum([r[2] for r in self.report])

def max_payment(self):
    if self.report == []: return []
    max_p = max([r[2] for r in self.report])
    return [r for r in self.report \
            if r[2] == max_p]
```

► ข้อที่ 3

```
def add_book(self, book, n):
    for b in self.books:
        if b[0] == book:
            b[1] += n
            break
    else:
        self.books.append([book,n])

def delete_book(self, book):
    self.books = [[b,n] for [b,n] \
        in self.books if b != book]

หรือ
    for k in range(len(self.books)):
        if self.books[k][0] == book:
            self.books.pop(k)
            break

def get_total(self):
    return sum([b.price*n for \
        [b,n] in self.books])

def __lt__(self, rhs):
    return self.get_total() < \
        rhs.get_total()
```

► ข้อที่ 4

```
def add_value(self, x):
    self.value += x

def enter(self, station):
    if self.station == '':
        self.station = station
        return True
    else:
        return False

def leave(self, station):
    if self.station == '':
        return (self.value, -2)
    price = Station.get_price \
        (self.station, station)
    if price > self.value:
        return (self.value, -1)
    else:
        self.value -= price
        self.station = ''
        return (self.value, 0)

def __lt__(self, rhs):
    return self.value < rhs.value
```

## เว็บไซต์ที่เป็นประโยชน์

- [youtu.be/U2l\\_xgpVsu0?list=PL0ROnCzUGB4ieaQndKybT9xyoq2n9NGq](https://youtu.be/U2l_xgpVsu0?list=PL0ROnCzUGB4ieaQndKybT9xyoq2n9NGq) (ภาพยนตร์บรรยายเนื้อหาวิชา)
- [en.wikibooks.org/wiki/Python\\_Programming](https://en.wikibooks.org/wiki/Python_Programming) (เว็บไซต์อ้างอิงสำหรับการเขียนโปรแกรมภาษา Python)
- [www.python.org](https://www.python.org) (เว็บไซต์ทางการของภาษา Python)
- [docs.python.org/3/tutorial/index.html](https://docs.python.org/3/tutorial/index.html), [docs.python.org/3/library/index.html](https://docs.python.org/3/library/index.html), [docs.python.org/3/reference/index.html](https://docs.python.org/3/reference/index.html) (เว็บไซต์ทางการของภาษา Python ส่วนเอกสารอ้างอิง)
- [www.numpy.org](https://www.numpy.org) (เว็บไซต์ทางการของคลังคำสั่ง NumPy)
- [www.lfd.uci.edu/~gohlke/pythonlibs](https://www.lfd.uci.edu/~gohlke/pythonlibs) (เว็บไซต์สำหรับดาวน์โหลดคลังคำสั่งภาษา Python)
- [www.pythontutor.com](https://www.pythontutor.com) (เว็บไซต์สำหรับการ visualize โปรแกรมภาษา Python)
- [repl.it/languages/python3](https://repl.it/languages/python3) (เว็บไซต์สำหรับเขียนโปรแกรมภาษา Python ออนไลน์ สามารถใช้ NumPy ได้)
- [openbookproject.net/thinkcs/python/english3e](https://openbookproject.net/thinkcs/python/english3e) (เว็บไซต์อ้างอิงสำหรับการเขียนโปรแกรมภาษา Python)

## Error ที่สามารถพบได้

### แบบขึ้นเป็นกล่องข้อความ

ข้อความ error	สาเหตุ	แนวทางการแก้ไข
unexpected EOF while parsing	ใส่วงเล็บไม่ครบ เช่น <code>x = (1+(2**3)</code>	ตรวจสอบวงเล็บให้ครบ
EOL while scanning string literal	ใส่อักขระประกาศจบสตริงไม่ครบ เช่น <code>s = 'hello</code>	เติมอักขระประกาศปิดสตริงให้ครบ
invalid syntax	มีการเขียนโปรแกรมผิดกฎของภาษา เช่น ลืม : หลัง if หรือ while หรือเรียง else มาก่อน if หรือเปรียบเทียบด้วย = ตัวเดียว เป็น if x = 1 : หรืออาจจะมีอักขระพิเศษซ่อนอยู่ ซึ่งมาจากการ copy-paste คำสั่งจากไฟล์ pdf	ตรวจสอบการเขียนโปรแกรมให้เป็นไปตามกฎของภาษา
unexpected indent	มีการเยื้องที่ไม่ตรงกัน เช่น <code>if a&gt;0:     a+=1     a+=2</code>	จัดการเยื้องให้ตรงกัน

หมายเหตุ ควรตรวจสอบข้อผิดพลาดทั้งบรรทัดที่เกิด error และบรรทัดก่อนหน้าบรรทัดที่เกิด error

## แบบขึ้นเป็นตัวอักษรสีแดงใน shell

ข้อความ error	สาเหตุ	แนวทางการแก้ไข
name 'x' is not defined	ไม่ได้กำหนดค่าตัวแปร x ก่อนใช้งาน เช่น <code>print(x+1)</code> หรือพิมพ์ชื่อฟังก์ชันผิด เช่น <code>printt('cat')</code> หรือลืม <code>import</code>	กำหนดค่าตัวแปรก่อนใช้งานให้เรียบร้อย ตรวจสอบการพิมพ์ชื่อฟังก์ชัน ทำการ <code>import</code> ให้เรียบร้อย
ZeroDivisionError: division by zero	มีการหารด้วย 0 เช่น <code>print(5/0)</code>	แก้ไขตัวหารให้ไม่เป็น 0
ImportError: No module named 'maht'	พิมพ์ชื่อ module ที่ต้องการ import ผิด เช่น <code>import maht</code>	แก้ไขชื่อ module ที่ต้องการ import ให้ถูกต้อง
AttributeError: module 'math' has no attribute 'arcsin'	module ที่เรียก ไม่มีฟังก์ชันที่ต้องการ เช่น <code>print(math.arcsin(0))</code>	แก้ไขชื่อฟังก์ชันให้ถูกต้อง เช่นในที่นี้ต้องใช้ <code>math.asin(0)</code>
ValueError: math domain error	มีการใส่ค่าที่ฟังก์ชันไม่รองรับ เช่น <code>print(math.asin(1000))</code> <code>print(math.log(10,0))</code>	ตรวจสอบค่าที่ฟังก์ชันนั้นรองรับ และใส่ค่าที่ถูกต้อง
invalid literal for int() with base 10	มีการใส่ค่าที่แปลงไม่ได้ลงไปใน <code>int()</code> เช่น <code>x = int('cat')</code>	ตรวจสอบค่าที่ต้องการแปลง ว่าเป็นจำนวนเต็มหรือไม่ และแก้ไขให้ถูกต้อง
could not convert string to float	มีการใส่ค่าที่แปลงไม่ได้ลงไปใน <code>float()</code> เช่น <code>x = float('cat')</code>	ตรวจสอบค่าที่ต้องการแปลง ว่าเป็นจำนวนทศนิยมหรือไม่ และแก้ไขให้ถูกต้อง
Can't convert 'int' object to str implicitly	มีการบวก <code>int</code> กับ <code>string</code> เช่น <code>print('cat'+3)</code>	แก้ไขให้ถูกต้อง อาจใช้ <code>comma</code> แทนเครื่องหมาย <code>+</code> หรือเพิ่มอัญประกาศเป็น <code>'3'</code>
<code>sqrt()</code> takes exactly one argument (3 given)	ฟังก์ชัน <code>sqrt()</code> ต้องใส่ข้อมูล 1 ค่า แต่ตอนนี้ใส่ไป 3 ค่า เช่น <code>x = sqrt(4,9,16)</code>	เปลี่ยนให้ใส่ข้อมูลตามจำนวนที่ฟังก์ชันกำหนด
ValueError: too many values to unpack	ค่าที่มีให้ มากกว่าจำนวนตัวแปร เช่น <code>a,b,c = 1,2,3,4,5</code>	แก้ไขจำนวนทั้งสองฝั่งให้เท่ากัน
ValueError: not enough values to unpack	ค่าที่มีให้ น้อยกว่าจำนวนตัวแปร เช่น <code>a,b,c = 1,2</code>	แก้ไขจำนวนทั้งสองฝั่งให้เท่ากัน
TypeError: unorderable types: int() < str()	มีการเปรียบเทียบจำนวนกับสตริง เช่น <code>if 9 &lt; '9':</code>	แก้ไขประเภทข้อมูลให้ตรงกัน
'float' object cannot be interpreted as an integer	มีการใส่ค่า <code>float</code> ในส่วนที่ควรจะเป็นค่า <code>int</code> เช่น <code>for i in range(2.5):</code>	แก้ไขค่าให้เป็นจำนวนเต็ม

ข้อความ error	สาเหตุ	แนวทางการแก้ไข
'int' object is not iterable	มีการใช้คำสั่ง in กับสิ่งที่ไล่ลำดับไม่ได้ (int หรือ float) เช่น for i in 5:	ใช้ in กับสิ่งที่ไล่ลำดับได้ เช่น range, string, list, tuple, set, dict
index out of range	มีการเรียก index เกินจากที่มี เช่น x = 'cat'; print(x[100]) y = [1,2,3]; print(y[100])	ตรวจสอบว่า index ที่ต้องการเรียก ไม่เกินความยาวที่มี
indices must be integers	มีการเรียก index ของสตริงหรือลิสต์ ด้วยค่าที่ไม่เป็นจำนวนเต็ม เช่น x = 'cat'; print(x[2/1]) y = [1,2,3]; print(y['a'])	ตรวจสอบว่า index ที่เรียก เป็นจำนวนเต็มหรือไม่ และแก้ไขให้ถูกต้อง
FileNotFoundError: No such file or directory	ต้องการเปิดไฟล์ แต่ไม่เจอไฟล์นั้น	ตรวจสอบว่ามีไฟล์อยู่จริง และพิมพ์ชื่อไฟล์ถูกต้อง
OSError: Invalid argument: 'D:\x0cile.txt'	พิมพ์ชื่อไฟล์ด้วยเครื่องหมาย \ เช่น f = open('D:\file.txt')	พิมพ์ชื่อไฟล์ด้วย \\ แทน เป็น f = open('D:\\file.txt')
KeyError	dict ไม่มี key ที่ต้องการ เช่น d = {}; print(d[1]) หรือ set ไม่มีสมาชิกที่ต้องการ เช่น s = {1,2,3}; s.remove(4)	ดูว่าใช้ key ถูกต้องหรือไม่ เพิ่ม code เพื่อตรวจสอบว่ามี key อยู่ใน dict หรือไม่
'set' object has no attribute 'delete' (อาจเกิด error กับ list, string, tuple และ dict ได้ด้วย)	เรียกใช้บริการที่ไม่มีกำหนดไว้ เช่น s = {1,2,3}; s.delete(1) ซึ่ง set ไม่มีบริการที่ชื่อว่า delete	แก้ไขการใช้คำสั่งให้ถูกต้อง
TypeError: unhashable type	มีการใช้ list, set, dict เป็น key ของ dict หรือเก็บ list, set, dict ใน set เช่น d = {}; d[{1}] = 'cat'	ตรวจสอบประเภทข้อมูลที่นำมาใช้ว่าไม่ใช่ list, set, dict
unsupported operand type(s) for +=: 'NoneType' and 'int'	มีการนำ None ไปประมวลผล เช่น def func(x): x += 1 print(func(3)+1) ซึ่งอาจเกิดจากการลืม return	return ค่าให้ถูกต้อง
'int' object is not callable	มีการเรียกฟังก์ชันด้วยตัวแปร เช่น a = 3 b = a(5) อาจเกิดจากการตั้งชื่อฟังก์ชันซ้ำกับตัวแปร	ตรวจสอบการตั้งชื่อฟังก์ชันและตัวแปร และแก้ไขให้ถูกต้อง
RecursionError: maximum recursion depth exceeded	มีการเรียกฟังก์ชันตัวเองซ้ำมากเกินไป อาจเกิดจากการลืมกรณีพื้นฐานของการ recursive เช่น def f(n): return n*f(n-1)	เติมกรณีพื้นฐานของการ recursive และเช็คเงื่อนไขการหยุดทำงานให้ถูกต้อง



# ศุภเสณย์ อุษัยศรี

Intania 86

Solutions Engineer, Facebook  
และ นายกสมาคมโปรแกรมเมอร์ไทย



ผมคิดว่าการเรียนเขียนโปรแกรมจะเป็นเหมือนการเรียนรู้ภาษาที่ 3 ในอนาคตอันใกล้ ยิ่งเทคโนโลยีแทรกซึมไปในทุกสิ่งเร็วเท่าไร ความต้องการคนที่เข้าใจและควบคุมเทคโนโลยีได้ก็จะมีมากขึ้นเท่านั้น นอกจากนั้นการเรียนเขียนโปรแกรมายังช่วยให้เรารู้จักคิดแก้ปัญหาอย่างเป็นระบบ และให้พลังในการสร้างสรรค์สิ่งต่าง ๆ ได้อย่างไม่รู้จัก

# ณัฐยา ลีละศุภกุล

Intania 95

ตลาดหลักทรัพย์แห่งประเทศไทย



ถ้าเทียบการสร้าง software กับการสร้างตึกตึกหนึ่ง **programming** ก็เหมือนเป็นเสาเข็ม ที่รองรับแรงของทั้งตึกให้ตั้งอยู่ได้ และไม่พังทลาย ตราบใดที่ไม่มีเสาเข็ม ก็ไม่สามารถสร้างตึกได้ ดังนั้น **programming** ก็เป็นเหมือนรากฐานที่สำคัญของ software ถ้าเรามีทักษะ **programming** ที่ดี เราก็จะสามารถพัฒนา software ที่มีคุณภาพได้เช่นกัน

คำสั่งพื้นฐาน	import math
<p><code>x = int(input())</code> รับข้อมูลจำนวนเต็ม 1 จำนวน</p> <p><code>x,y,z = [float(e) for e in input().split()]</code> รับข้อมูลจำนวนจริง 3 จำนวน แปรทศนิยมกัน คั่นด้วยช่องว่าง</p> <p><code>int(x), float(x), str(x):</code> คืนค่า x ที่ถูกเปลี่ยน ะเภทข้อมูล เป็นจำนวนเต็ม จำนวนจริง และสตริง</p> <p><code>abs(n):</code> คืนค่าสัมบูรณ์ของ n</p> <p><code>round(f):</code> คืนค่าจำนวนเต็มที่เกิดจากการปัดเศษจำนวนจริง f โดยถ้าเศษของ f มีค่าตั้งแต่ 0.5 จะปัดขึ้น ถ้าน้อยกว่า 0.5 จะปัดลง</p> <p><code>round(f,d):</code> คืนค่าจำนวนจริงที่เกิดจากการปัดเศษจำนวนจริง f โดยปัดให้มีจำนวนตัวเลขหลังจุดทศนิยม d หลัก</p> <p><code>range(start,stop [,step])</code> หรือ <code>range(stop):</code> คืนค่าเป็นลิสต์ของตัวเลขตามลำดับตั้งแต่ start ถึง stop-step และเพิ่มขึ้นทีละ step (ถ้าไม่ระบุ start จะมีค่า 0 และ step จะมีค่า 1)</p> <p><code>enumerate(L):</code> คืนลิสต์ของ tuple (index, element) ของแต่ละข้อมูลในลิสต์ L</p> <p><code>len(a):</code> คืนค่าเป็นจำนวนข้อมูลใน a ซึ่ง a อาจเป็นลิสต์ ดิกชันนารี เซต ทูเปิล สตริง หรือ numpy array ก็ได้</p> <p><code>max(a), min(a):</code> คืนค่าที่มาก/น้อยที่สุดของข้อมูลใน a ซึ่ง a อาจเป็นลิสต์ ดิกชันนารี เซต ทูเปิล หรือสตริงก็ได้ (numpy array ใช้ <code>np.max(a), np.min(a)</code>)</p> <p>ถ้า a เป็นดิกชันนารี จะคืนค่ามาก/น้อยที่สุดของ key ของดิกชันนารี</p> <p><code>type(a):</code> คืนค่าประเภทของ a เช่น <code>type([1,2])</code> ได้ <code>&lt;class 'list'&gt;</code></p> <p><code>list(), dict(), tuple(), set():</code> สร้างลิสต์ว่าง ดิกชันนารีว่าง ทูเปิลว่าง เซตว่าง</p>	<p><code>math.exp(x):</code> คืนค่า e ยกกำลัง x</p> <p><code>math.cos(x):</code> คืนค่า cosine ของ x เรเดียน</p> <p><code>math.sin(x):</code> คืนค่า sine ของ x เรเดียน</p> <p><code>math.sqrt(x):</code> คืนค่ารากที่สองของ x</p> <p><code>math.log(x,base):</code> คืนค่าลอการิทึมของ x ฐาน base</p> <p><code>math.degrees(x):</code> แปลงมุม x จากเรเดียนเป็นองศา</p> <p><code>math.radians(x):</code> แปลงมุม x จากองศาเป็นเรเดียน</p> <p><code>math.pi, math.e:</code> ค่าคงที่ pi และ e</p>
list L	string s
<p><code>L.append(e):</code> เพิ่ม e ไปที่ท้ายลิสต์ L</p> <p><code>L.insert(index,e):</code> เพิ่ม e ไปที่ตำแหน่ง index ในลิสต์ L</p> <p><code>L.pop(index):</code> ลบข้อมูลที่ตำแหน่ง index และคืนค่าข้อมูลที่ถูกลบ</p> <p><code>L.count(e):</code> คืนจำนวนครั้งที่ e ปรากฏในลิสต์ L</p>	<p><code>s.lower():</code> คืนสตริงใหม่ที่มีค่าเหมือน s แต่เป็นตัวพิมพ์เล็กทั้งหมด</p> <p><code>s.upper():</code> คืนสตริงใหม่ที่มีค่าเหมือน s แต่เป็นตัวพิมพ์ใหญ่ทั้งหมด</p> <p><code>s.find(sub):</code> คืน index แรกสุดที่พบ sub ใน s ถ้าไม่พบคืนค่า -1</p> <p><code>s.find(sub,i):</code> คืน index แรกสุดที่พบ sub ใน s โดยเริ่มค้นที่ index i</p> <p><code>s.count(sub):</code> คืนจำนวนครั้งที่ sub ปรากฏในสตริง s</p> <p><code>s.split(sep):</code> คืนลิสต์ของสตริงที่แยกด้วย sep (หรือ space ถ้าไม่ระบุ)</p> <p><code>s.strip():</code> คืนสตริงใหม่ที่มีค่าเหมือน s แต่ตัด spaces ท้ายออก</p> <p><code>s.join(L):</code> คืนสตริงที่สร้างจากการนำแต่ละ element ในลิสต์ L มาต่อกัน โดยมี s เป็นตัวคั่นระหว่างข้อมูลที่ต่อกัน (L ต้องเป็นลิสต์ของสตริง)</p>
dict D	import numpy as np
<p><code>D.items():</code> คืนลิสต์ของ tuple (key, value) ของดิกชันนารี D</p> <p><code>D.keys():</code> คืนลิสต์ของ key ทั้งหมดของดิกชันนารี D</p> <p><code>D.values():</code> คืนลิสต์ของ value ทั้งหมดของดิกชันนารี D</p> <p><code>D.pop(k):</code> ลบข้อมูลใน D ที่มี key เป็น k และคืนค่า value ของ key นั้น</p> <p><code>D.update(D1):</code> เพิ่มข้อมูลจากดิกชันนารี D1 เข้าไปใน D</p>	<p><code>np.array(L):</code> คืนค่า numpy array ที่สร้างจากลิสต์ L</p> <p><code>np.arange(start,stop,step):</code> คืนอาร์เรย์ 1 มิติของจำนวนที่มีค่าตาม start,stop,step</p> <p><code>np.ones(shape):</code> คืนอาร์เรย์ที่มีค่า 1 ทั้งหมด มีขนาดตาม tuple shape</p> <p><code>np.zeros(shape):</code> คืนอาร์เรย์ที่มีค่า 0 ทั้งหมด มีขนาดตาม tuple shape</p> <p><code>np.identity(size):</code> คืนอาร์เรย์ขนาด size x size ซึ่งมีข้อมูลในแนวเส้นทแยงมุมเป็น 1 และค่าในตำแหน่งอื่น ๆ เป็น 0</p> <p><code>np.empty_like(a):</code> คืนอาร์เรย์ใหม่ที่มีขนาดเหมือน a แต่ไม่มีการกำหนดค่าข้อมูลในอาร์เรย์ใหม่นี้</p> <p><code>np.add(a,b), np.subtract(a,b), np.multiply(a,b), np.divide(a,b):</code> คืนค่าอาร์เรย์ใหม่ที่เป็นผลบวกลบคูณหารแบบช่องต่อช่องของ a และ b</p> <p><code>np.dot(a,b):</code> คืนอาร์เรย์ที่เป็นผลคูณแบบเมทริกซ์ของ a และ b</p> <p><code>np.sin(a), np.cos(a), np.sqrt(a), np.abs(a):</code> คืนอาร์เรย์ที่มีค่าของข้อมูลในแต่ละตำแหน่งเป็นผลจากการเรียกฟังก์ชัน sine, cosine, sqrt, abs กับข้อมูลในอาร์เรย์ a ที่ตำแหน่งเดียวกัน</p> <p><code>np.max(a,axis), np.min(a,axis):</code> คืนอาร์เรย์ของค่ามาก/น้อยที่สุดใน a ตาม axis ที่กำหนด</p> <p><code>np.argmax(a,axis), np.argmin(a,axis):</code> คืนอาร์เรย์ของ index ที่มีค่ามาก/น้อยที่สุดใน a ตาม axis ที่กำหนด ตัวอย่างเช่น  <code>a = np.array([ [2, 4, 6], [8, 10, 12] ])</code>  <code>np.max(a)</code> คืนค่า 12, <code>np.argmax(a)</code> คืนค่า 5  <code>np.max(a,axis=0)</code> คืนค่า array([8,10,12])  <code>np.argmax(a,axis=0)</code> คืนค่า array([1,1,1])  <code>np.argmax(a,axis=1)</code> คืนค่า array([2,2])</p> <p><code>np.sum(), np.std(), np.mean():</code> มีการใช้งานเหมือน <code>np.max()</code></p> <p><code>np.ndenumerate(a):</code> คืนลิสต์ของ tuple (position,element) ของแต่ละข้อมูลใน a โดย position เป็น tuple ที่เก็บตำแหน่งของข้อมูล</p>
set S	
<p><code>S.add(e):</code> เพิ่ม e ในเซต S</p> <p><code>S.difference(T):</code> คืนเซตใหม่ที่เท่ากับ S-T</p> <p><code>S.discard(e):</code> ลบ e ออกจากเซต S ถ้าไม่มี e ใน S ก็ไม่ทำอะไร</p> <p><code>S.intersection(T):</code> คืนเซตใหม่ที่เท่ากับ S ∩ T</p> <p><code>S.union(T):</code> คืนเซตใหม่ที่เท่ากับ S ∪ T</p> <p><code>S.issubset(T):</code> ทดสอบว่า S ⊆ T หรือไม่</p> <p><code>S.issuperset(T):</code> ทดสอบว่า S ⊇ T หรือไม่</p> <p><code>S.pop():</code> ลบข้อมูลหนึ่งตัวออกจากเซต S และคืนข้อมูลที่ถูกลบ</p> <p><code>S.update(T):</code> ให้ S = S ∪ T</p>	

# เปลี่ยนความยากลำบากเห็น มาสู่งานอาชีพ

ในอนาคต ประเทศไทยจำเป็นต้องมีบุคลากรทางด้านวิศวกรรมและเทคโนโลยีเพิ่มขึ้นอีกมาก  
เพื่อให้ทันต่อความต้องการ ด้วยเหตุนี้ เอสโซ่ และ บริษัทในเครือเอ็กซอนโมบิลในประเทศไทย  
จึงให้ความสนับสนุนโครงการเรียนรู้ที่เกี่ยวข้องกับคณิตศาสตร์และวิทยาศาสตร์  
เพื่อเตรียมความพร้อมทั้งด้านศักยภาพและทักษะให้เยาวชนรุ่นใหม่ สำหรับงานในศตวรรษที่ 21  
เพราะการส่งเสริมสนับสนุนเยาวชนของเราวันนี้  
จะส่งผลถึงความก้าวหน้าของประเทศต่อไปในวันข้างหน้า

Energy lives here™

## ExxonMobil

Mobil  Mobil



Find job opportunities at [jobs.exxonmobil.com](https://jobs.exxonmobil.com)



CONFIDENTIAL

“  
ทุกอย่างใน **python** จะดูซอฟต์  
เมื่อเป็นสัปดาห์

”  
- โปรแกรมเมอร์นิรนาม -

สนับสนุนการพิมพ์โดย

**ExxonMobil**

ISBN 978-616-407-189-6



9 786164 071896