

100 → เกิดผลคูณของเลข สองตัว

ex.

$$\frac{100}{2} = 50$$
$$100 = 50 \times 2$$

✓ จะสามารถเกิดจากผลคูณ  
ของเลขสองตัว

วิธีอื่น

$$100 = 10 \times 10 \rightarrow \text{เลขตัวเดียวคูณตัวเอง หรือ } \sqrt{x} \text{ หนึ่งตัว}$$

$$2, 4, 5 < \sqrt{100}$$

$$10, 20, 25, 50 \geq \sqrt{100}$$

→ ตัวเกิดจากสองตัวนี้คูณกัน

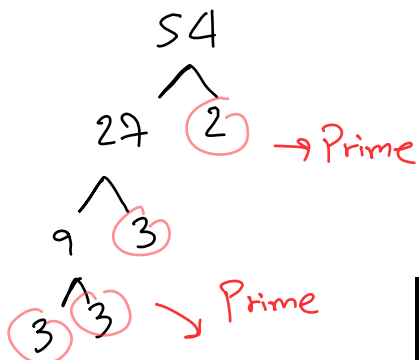
→ เลขที่เช็คแค่  $< \sqrt{x}$  เท่านั้น  
ถ้าใน  $\sqrt{x}$  ไม่เป็นเลขประกอบ  
→ "Prime Number" หนึ่ง

$$\text{range}(3, \text{int}(\sqrt{x} + 1), 2)$$

↓ เช็คทุก 2 ท

→ ถ้า 2 ไม่เป็นเลขประกอบ  
จะเช็คเลขต่อไปเลย

\* จำนวนเฉพาะเป็น Factor ที่ย่อยได้เล็กสุด



# 54 แรกด้วย Prime Num.  
ก่อนบวกให้มันลบแล้วไม่เหลืออะไร  
→ Prime Number  
ถ้าไม่เหลืออะไร  
→ ไม่ Prime Number

```
7 if x in all_primes:
8     return True
9 #If x is divisible by any prime in our list then it's not prime
10 for p in all_primes:
11     if x%p==0:
12         return False
13 #Otherwise, look at potential divisors beyond our primes list
14 upper_limit = int(math.sqrt(x)+1)
15 divisor = all_primes[len(all_primes)-1]+2
16 while divisor < upper_limit:
17     if isPrime(divisor):
18         all_primes.append(divisor)
19     if x%divisor==0:
20         return False
```

→ ถ้าพบค่าสุดท้ายใน list แล้ว  
ลบ 1 แล้ว +2  
(+2 เพราะมันแค่เลขตัว)

[2,3,5]

7

ใช้จำนวนเฉพาะ

all\_primes.py > is\_prime

```
1  """All Primes"""
2  def is_prime(n):
3      """check if the number is prime"""
4      all_prime = [2, 3] #initial
5      if n == 1:
6          return False
7      if n in all_prime:
8          return True
9      for i in all_prime:
10         if not n % i:
11             return False #หารลงตัว แสดงว่าไม่ใช่
12         max_range = int(n ** 0.5) + 1
13         divisor = all_prime[-1] + 2 #+2 เพราะตัวมากสุดใน prime หารแล้วยังไม่ลงตัว เลยเอาเลขตัวต่อไป
14         while divisor ≤ max_range:
15             if is_prime(divisor): #เลขตัวนั้นเป็น Prime (ความจริงโปรแกรมเราแค่ cnt ไม่จำเป็นต้องทำก็ได้)
16                 all_prime.append(divisor)
17             if not n % divisor: #ถ้าเจอตัวไหนที่หารแล้วลงตัวก็ไป
18                 return False
19             divisor += 2
20         return True
21  def coutner(n):
22      """count prime number from 1 to n"""
23      cnt = 0
24      for number in range(1, n+1):
25          if is_prime(number):
26              cnt += 1
27      print(cnt)
28  coutner(int(input()))
```