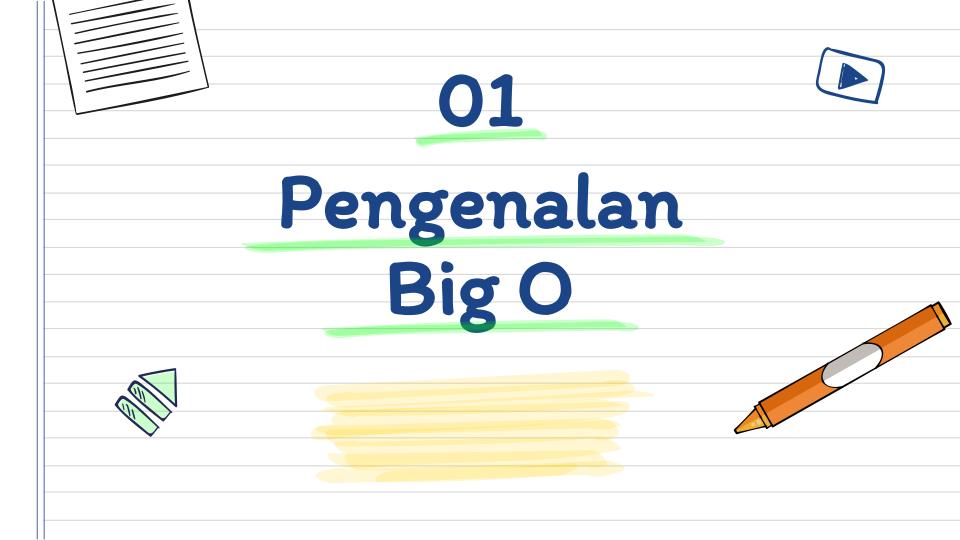
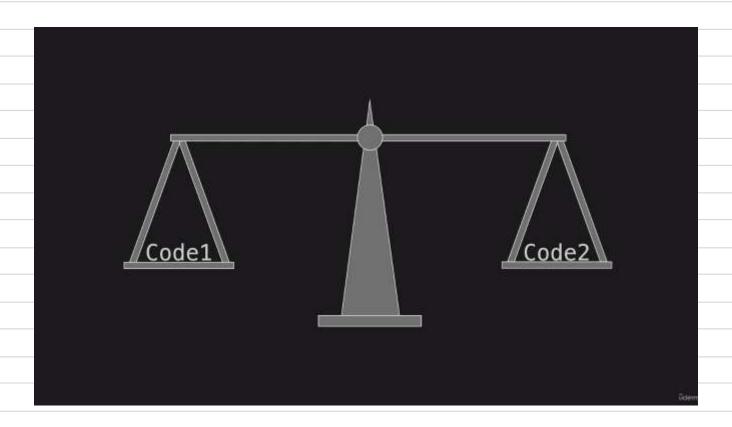
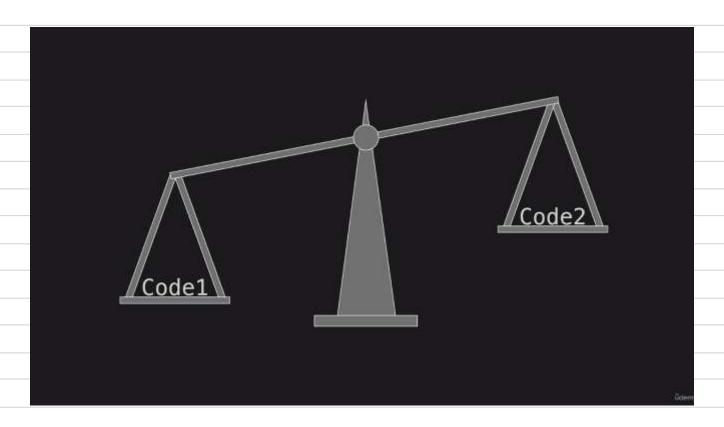
Struktur Data Meet 02

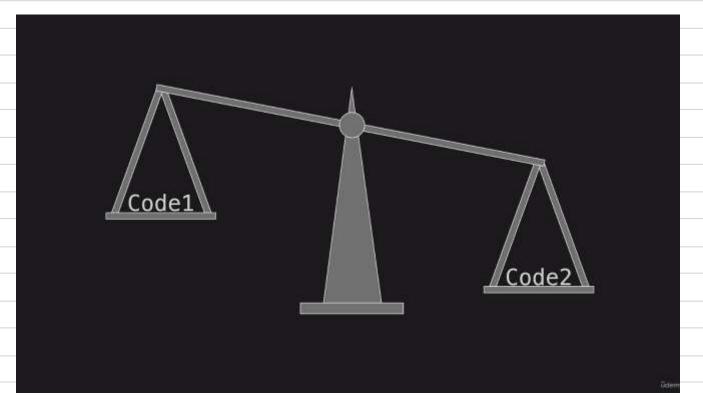




Code 1 lebih pendek



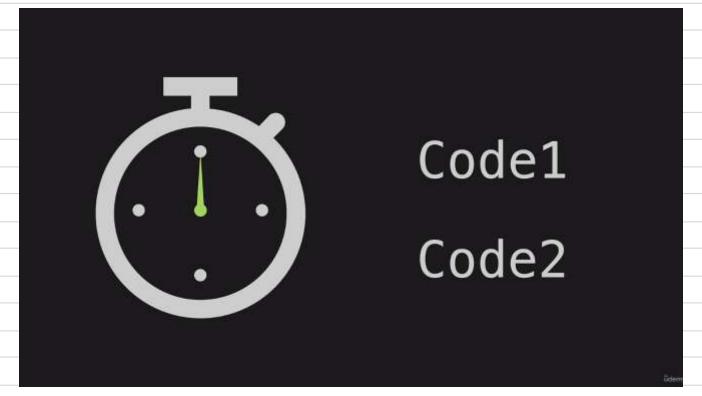
Code 2 lebih mudah dibaca



- Big O adalah cara untuk membandingkan 2 buah code
- Secara matematis
- Untuk menentukan mana code yang lebih efisien
- Efisiensi bisa dilihat dari 2 hal:
 - Efisien secara waktu
 - Efisien secara ruang

Time Complexity

- Code 1:15 detik
- Code 2:1 menit

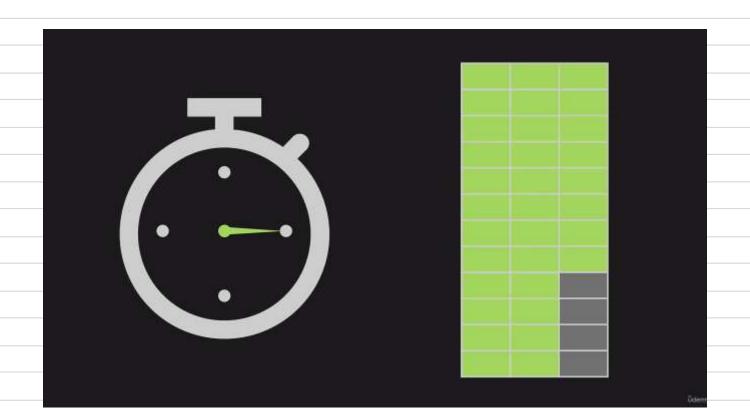


Time Complexity

- Dalam dunia pemrograman, time complexity tidak diukur berdasarkan waktu
- Karena waktu running juga ditentukan oleh kekuatan mesin
- Time complexity diukur dari banyaknya operasi yang dilakukan untuk menyelesaikan sesuatu

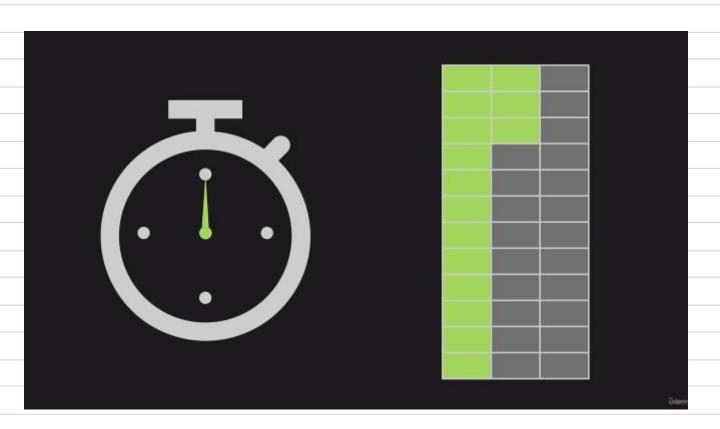
Space Complexity

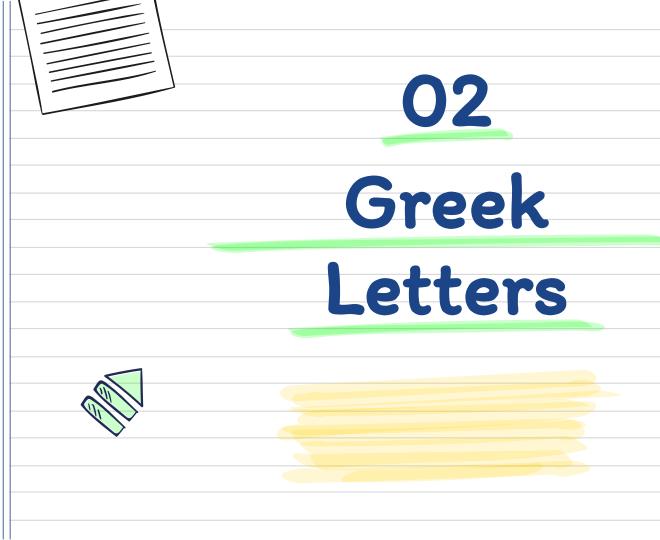
• Code 1 : 15 detik



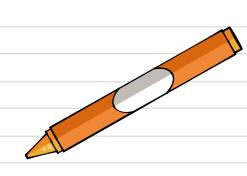
Space Complexity

• Code 2:1 Menit



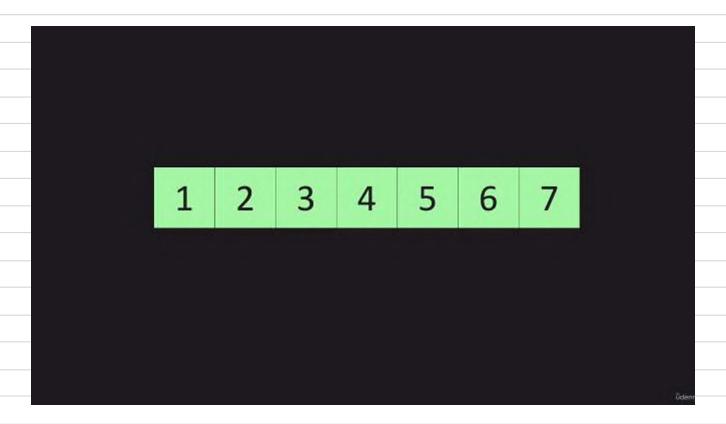






 Carilah angka "1" dari list di bawah ini

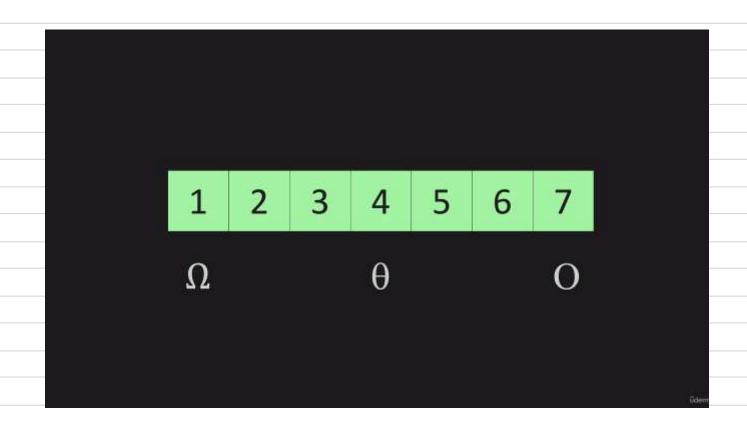
Greek Letters

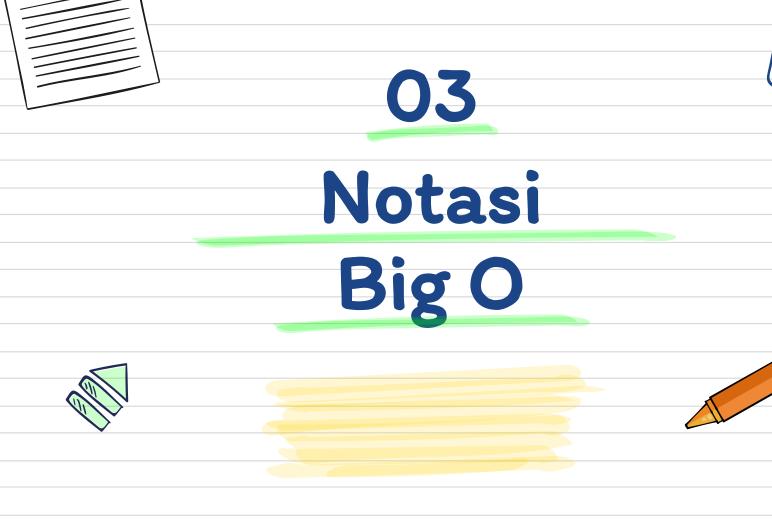


Greek Letters

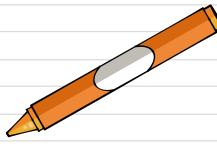
- Mencari angka "1" = Ω = **best scenario**
- Mencari angka "4" = θ = average scenario
- Mencari angka "7" = O = worst scenario

Greek Letters

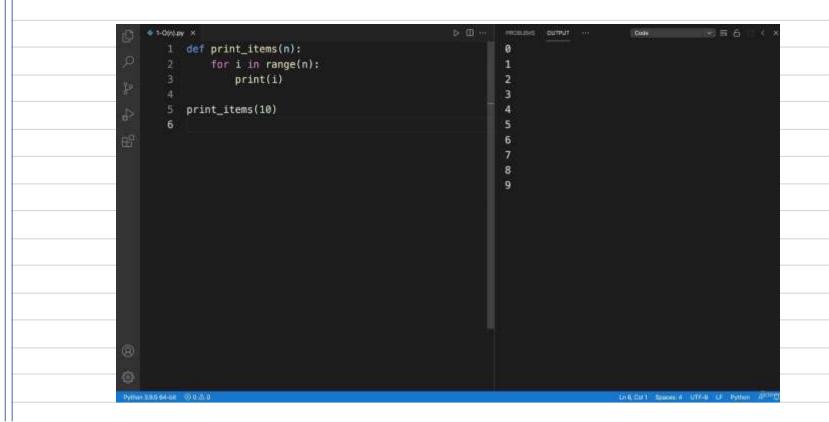




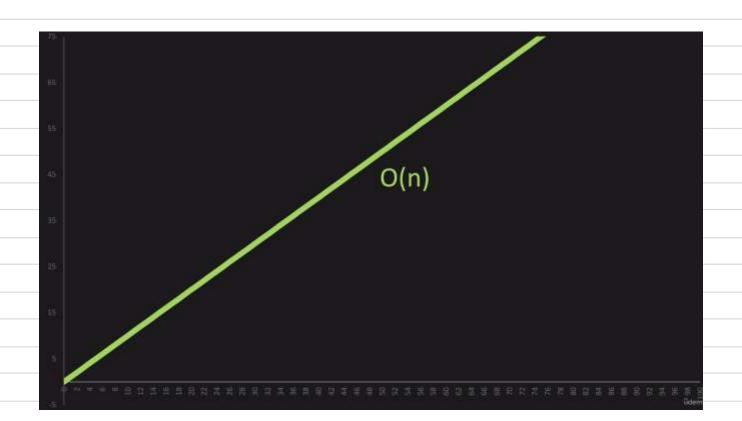




Notasi #1 : O(n)

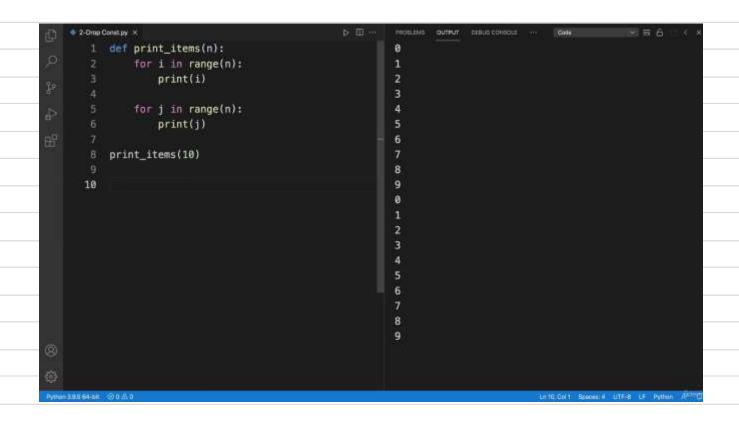


Notasi #1: O(n)



```
def print_items(n):
   for i in range(n):
      print(i)
             0(n)
```

```
def print_items(n):
    for i in range(n):
        print(i)
   for j in range(n):
        print(j)
```

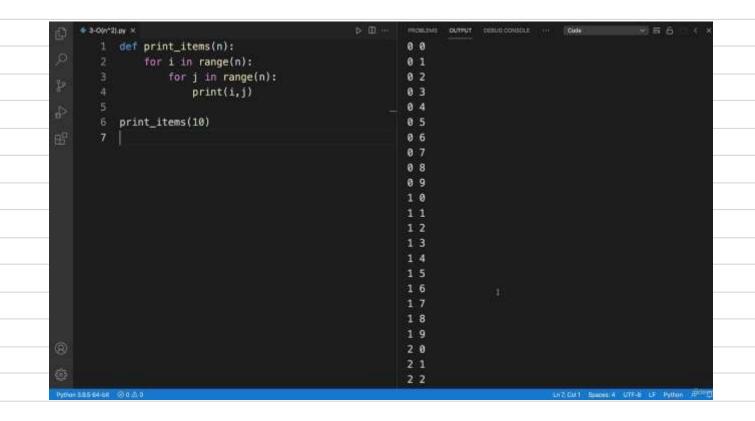


```
n + n = 2n
```

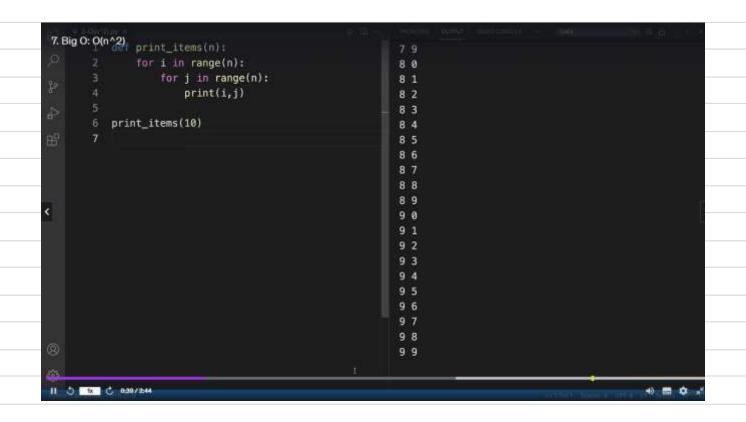
```
0(2n)
```

```
0(n)
```

Notasi #2 : O(n²)



Notasi #2 : O(n²)



Notasi #2 : $O(n^2)$

Notasi #2 : $O(n^2)$

```
n * n = n^2
```

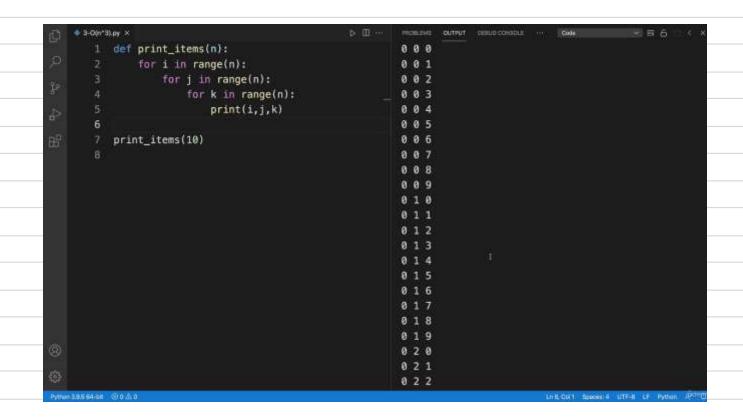
Notasi #2 : O(n²)

```
def print_items(n):
   for i in range(n):
       for j in range(n):
          print(i,j)
             0(n^2)
```

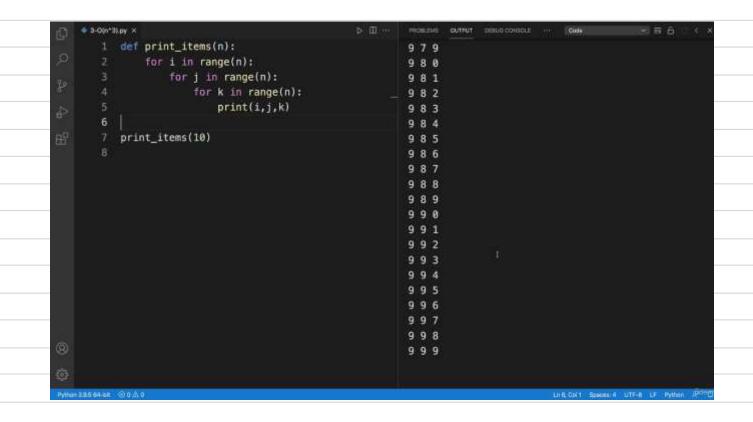
Notasi #2 : $O(n^2)$

```
def print_items(n):
    for i in range(n):
        for j in range(n):
            for k in range(n):
                print(i,j,k)
```

Notasi #2 : O(n²)



Notasi #2 : O(n²)



Notasi #2 : $O(n^2)$

```
n * n * n
```

Notasi $\#2:O(n^2)$

```
n \times n \times n = n^3
```

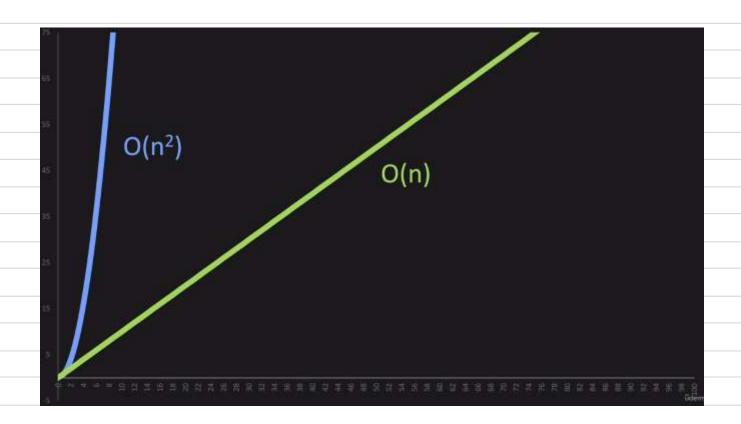
Notasi $\#2:O(n^2)$

```
0 (n^3)
```

Notasi $\#2:O(n^2)$

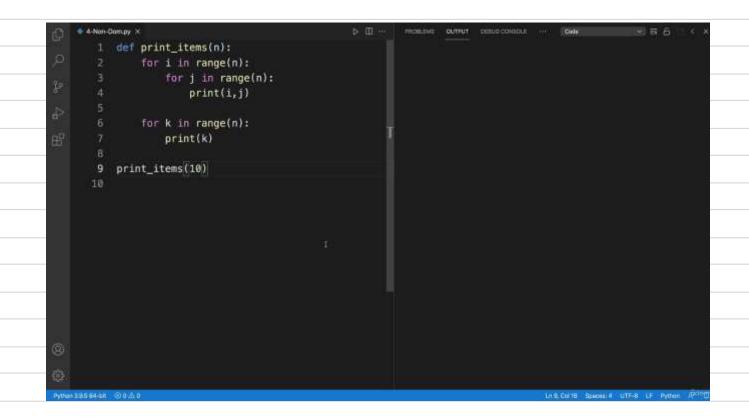
```
0 (n<sup>2</sup>)
```

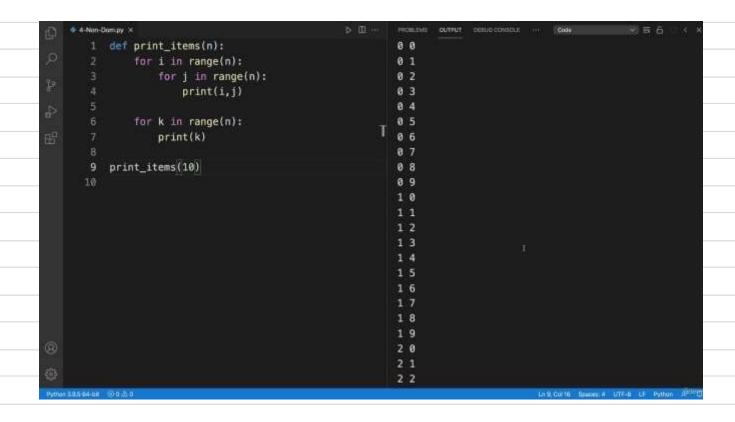
Notasi #2 : O(n²)

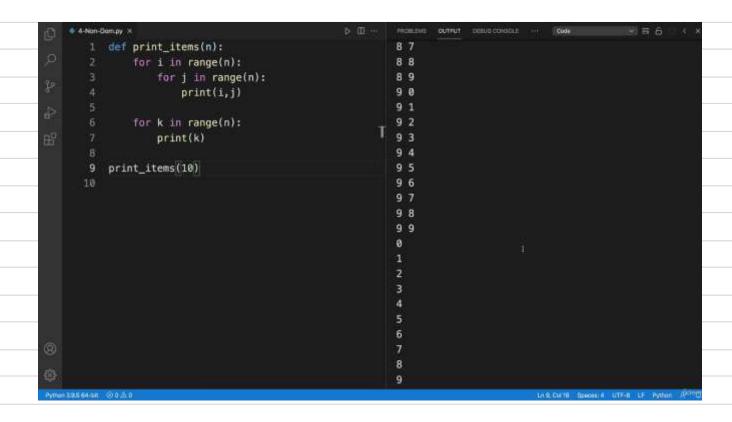


```
def print_items(n):
   for i in range(n):
      print(i)
             0(n)
```

```
def print_items(n):
    for i in range(n):
        for j in range(n):
            print(i,j)
    for k in range(n):
        print(k)
```

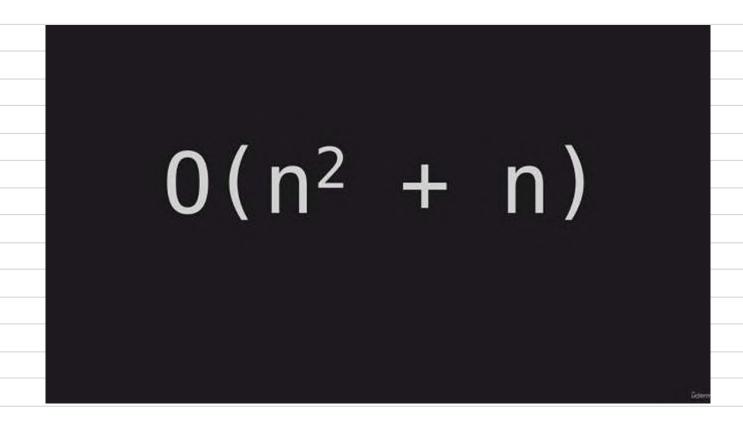


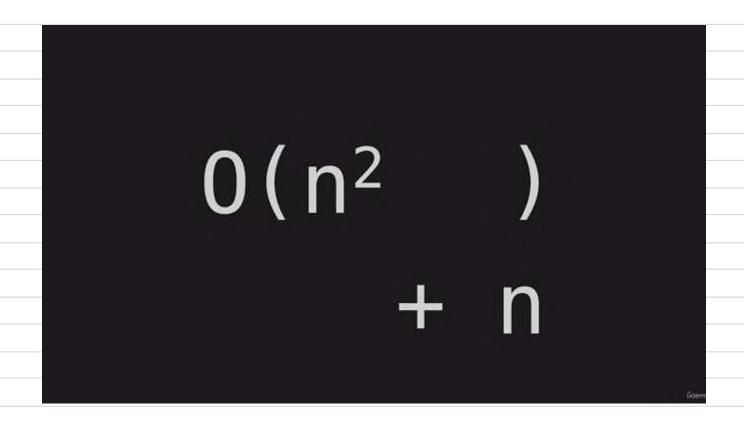


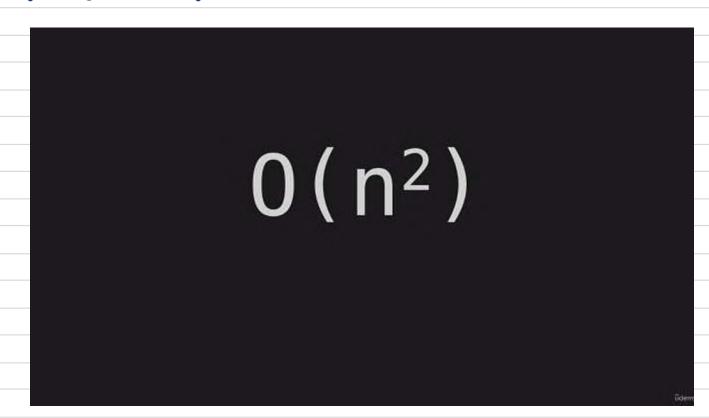


```
0(n<sup>2</sup>)
for k in range(n):
                          0(n)
   print(k)
```

```
def print_items(n):
    for i in range(n):
                               0 ( n<sup>2</sup> )
        for j in range(n):
            print(i,j)
    for k in range(n):
                                 0(n)
        print(k)
```







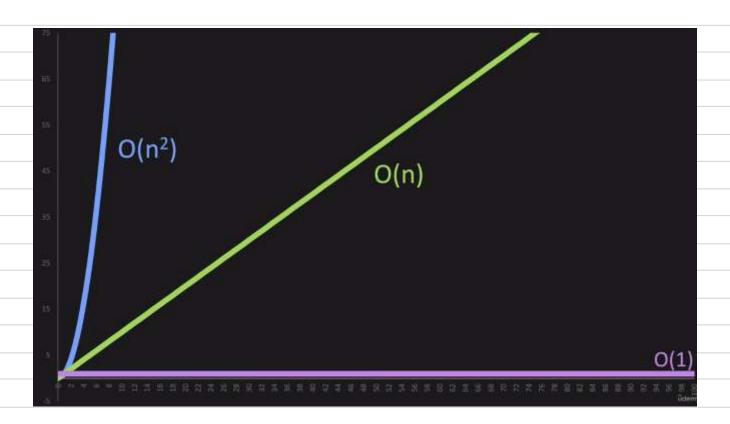
```
def add_items(n):
    return n + n
```

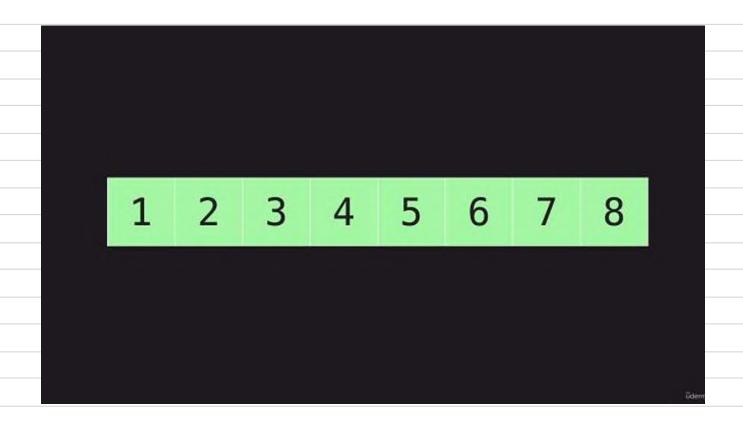
```
def add_items(n):
   return n + n
            0(1)
```

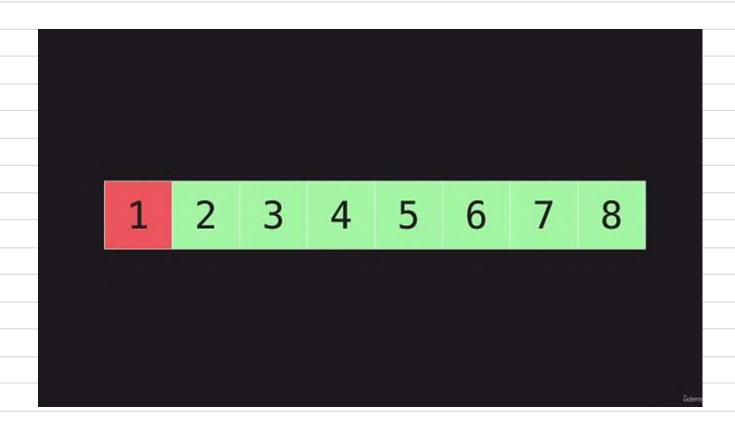
```
def add_items(n):
    return n + n + n
```

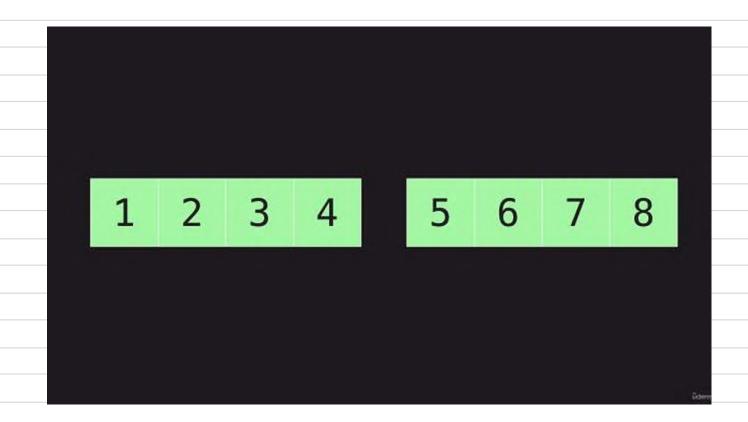
```
def add_items(n):
   return n + n + n
            0(2)
```

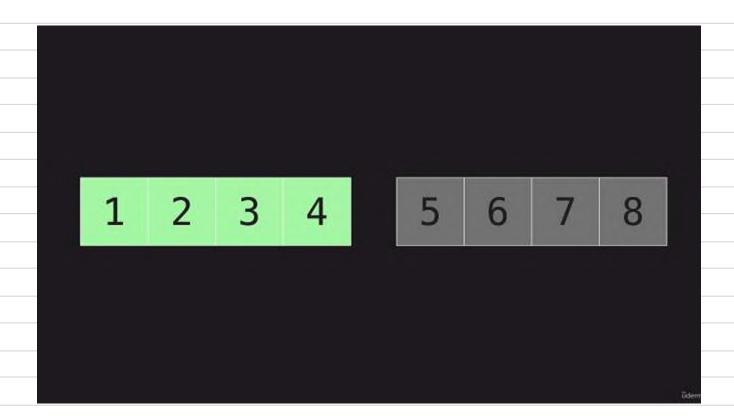
```
def add_items(n):
   return n + n + n
            0(1)
```

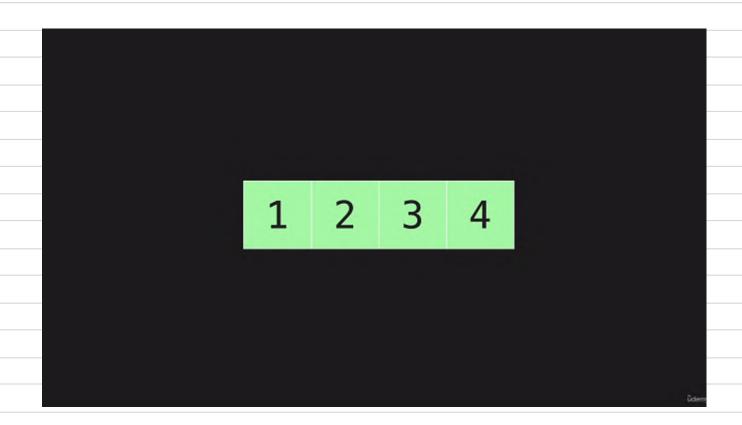


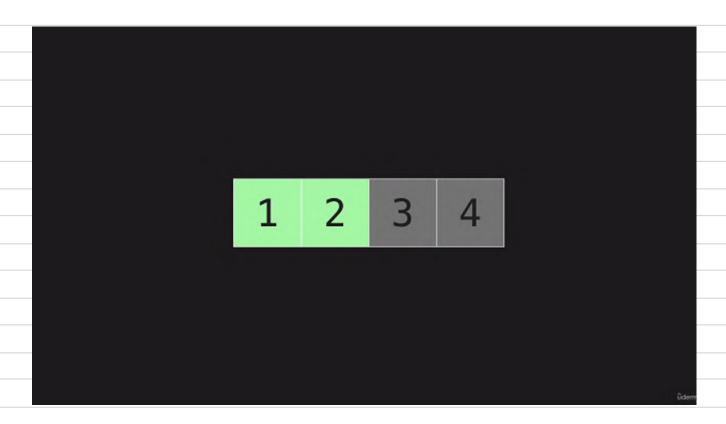






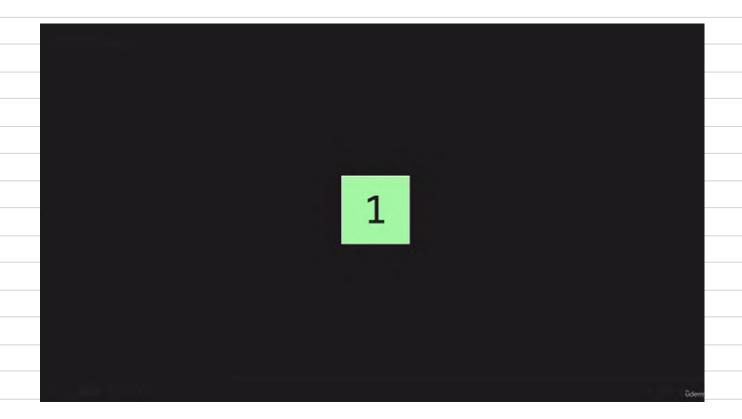


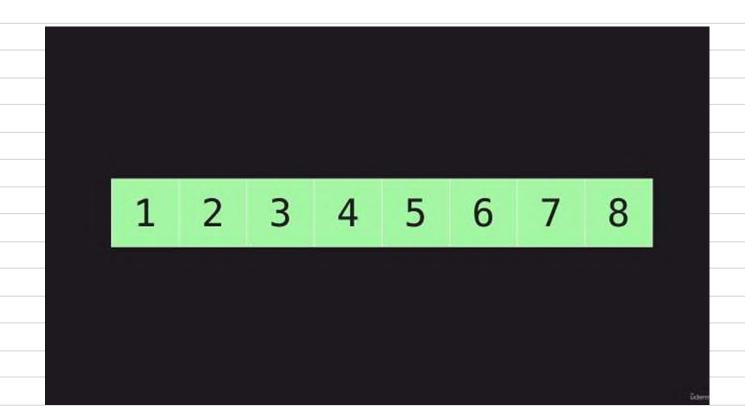


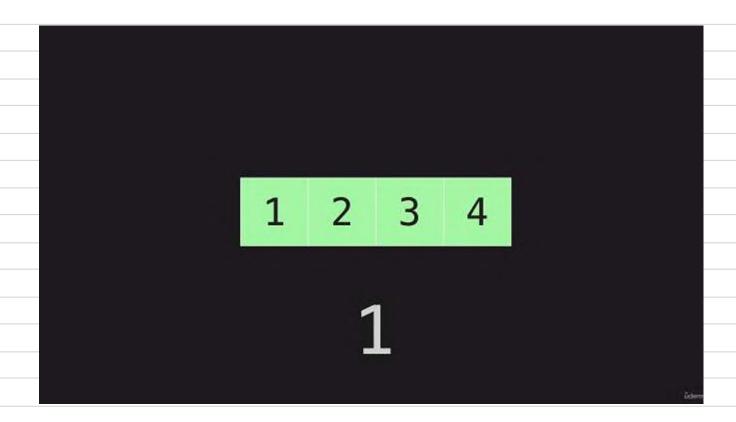


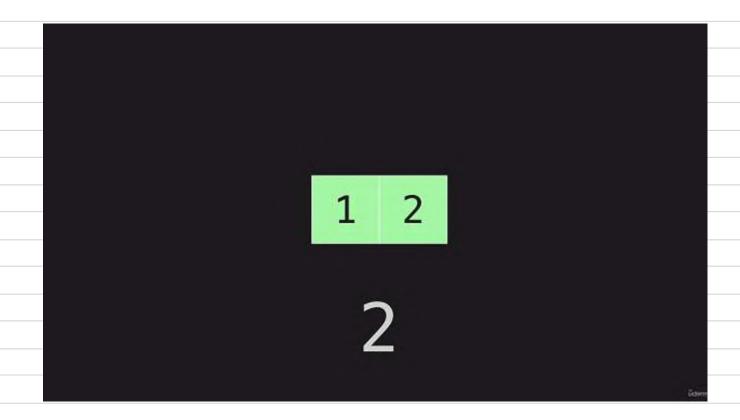




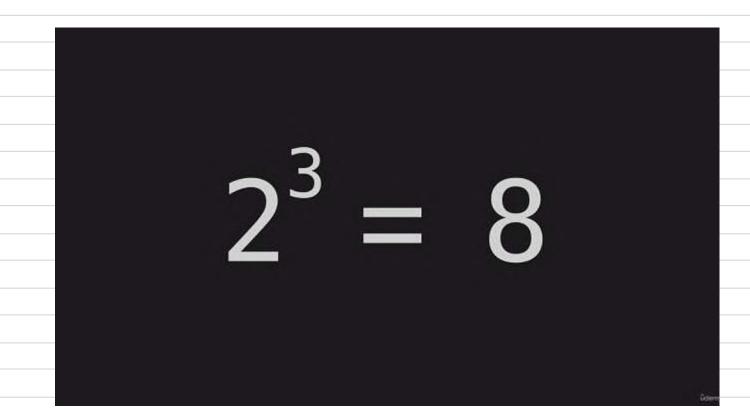






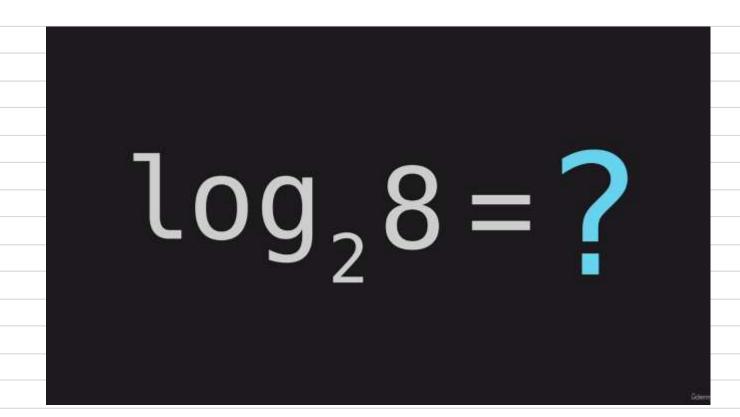






 $\log_{2} 8 = 3$

iidem



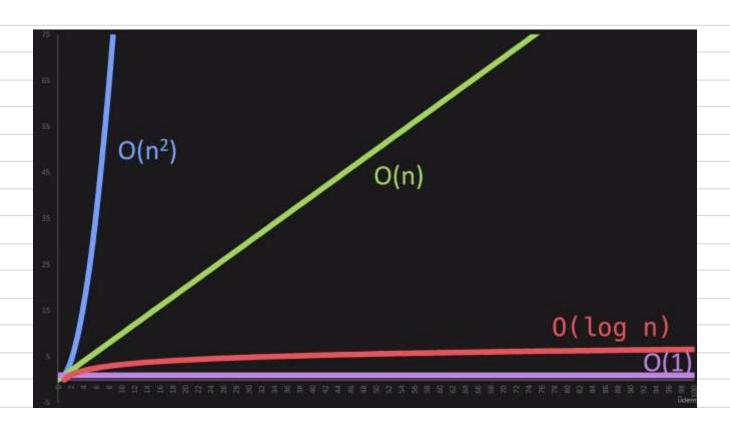
log₂1,073,741,824

log₂1,073,741,824

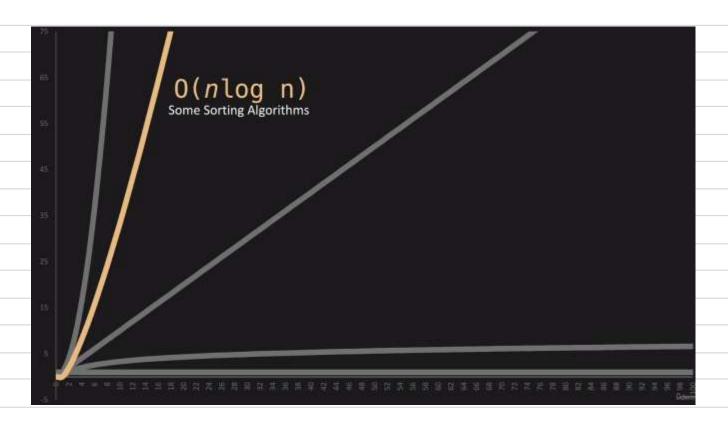
31

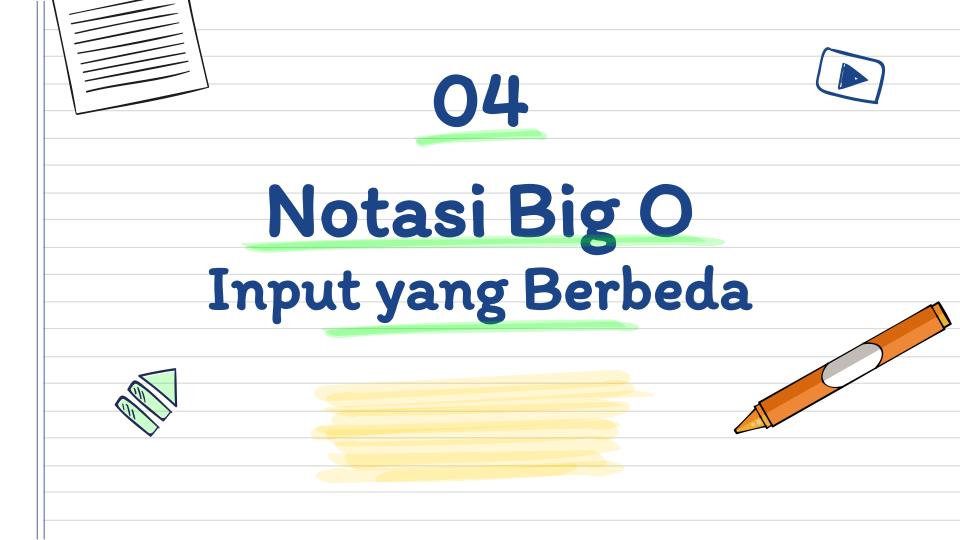
```
O(log n)
```

Notasi #4 : O(log n)



Notasi #4 : O(log n)





```
def print_items(n):
    for i in range(n):
        print(i)
    for j in range(n):
        print(j)
```

```
0(2n)
```

```
0(n)
```

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

```
def print_items(a, b):
                             0(n)
   for i in range(a):
       print(i)
                             0(n)
   for j in range(b):
       print(j)
```

```
def print_items(a, b):
   for i in range(a):
       print(i)
   for j in range(b):
       print(j)
               0(2n)
```

```
def print_items(a, b):
   for i in range(a):
       print(i)
   for j in range(b):
       print(j)
                0(n)
```

```
def print_items(a, b):
                              0(a)
+
   for i in range(a):
       print(i)
                              0(b)
   for j in range(b):
       print(j)
```

```
def print_items(a, b):
   for i in range(a):
      print(i)
   for j in range(b):
      print(j)
         0(a + b)
```

```
def print_items(a, b):
   for i in range(a):
      for j in range(b):
          print(i,j)
         0(a * b)
```

Terima Kasih

Ada Pertanyaan?