

## 01-D-3 循环 vs. 级数

#数据结构邓神

### 二重循环

```
for (int i = 0; i < n; i++)  
    for(int j = 0; j < n; j++)  
        O1Operation(i, j);
```

算数级数：

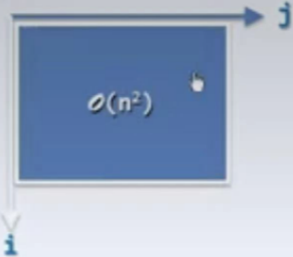
内循环循环一次  $n$  次，一共被外循环循环  $n$  次

$n + n + \dots + n$  (一共  $n$  个)  $= O(n^2)$  // 二次方复杂度

**循环 vs. 级数**

```
❖ for (int i = 0; i < n; i++)  
    for (int j = 0; j < n; j++)  
        O1Operation(i, j);
```

算术级数：

$$\sum_{i=0}^{n-1} n = n + n + \dots + n = n * n = O(n^2)$$


```
for (int i = 0; i < n; i++)  
    for(int j = 0; j < i; j++)  
        O1Operation(i, j);
```

算数级数

$0 + 1 + 2 + \dots + (n-1) = n(n-1)/2 = O(n^2)$  // 也是二次方复杂度

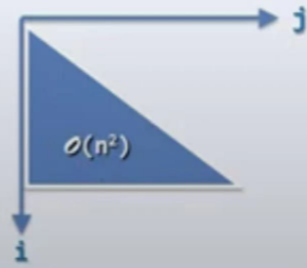
这边可以回到算数级数的定理： 复杂度等于末项的平方

虽然在形式上相对于上面的例子有所变化，但是在复杂度上是完全一样的

```
for (int i = 0; i < n; i++)
for (int j = 0; j < i; j++)
    O1Operation(i, j);
```

算术级数：

$$\sum_{i=0}^{n-1} i = 0 + 1 + \dots + (n-1) = \frac{n(n-1)}{2} = O(n^2)$$



从图像上虽然说这个例子的面积只有上面一个的一半，  
但是从渐进的角度分析，只有常数C的差距，并不影响其二次增长  
 $C = 1/2$

```
for (int i = 0; i < n; i++)
    for (int j = 0; j < i; j++)
        O1Operation(i, j);
```

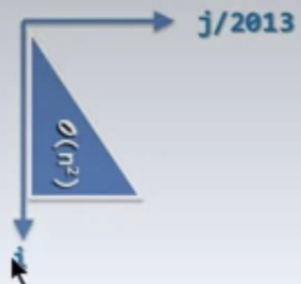
算数级数 复杂度  $O(n^2)$

与上面一样，对于渐进复杂度毫无影响，依然只有常数差距，对于渐进复杂度分析来说完全可以忽略  
只说这个三角形被压扁了，但是只是常系数的缩小。

### 循环 vs. 级数

```
❖ for (int i = 0; i < n; i++)
    for (int j = 0; j < i; j += 2013)
        O1Operation(i, j);
```

算术级数： ...



```
for (int i = 0; i < n; i <= 1)
    for (int j = 0; j < i; j++)
        O1Operation(i, j);
```

$i <= 1$

而是每次都是左移一位 等效于  $i *= 2$

这次相当于一个几何级数的形式

根据几何级数的复杂度与最后同阶，也就是直接是  $n$

$$1 + 2 + 4 + \dots + 2^{\lfloor \log_2 (n-1) \rfloor}$$

$$O(n)$$

```
❖ for (int i = 1; i < n; i <= 1)
```

```
    for (int j = 0; j < i; j++)
```

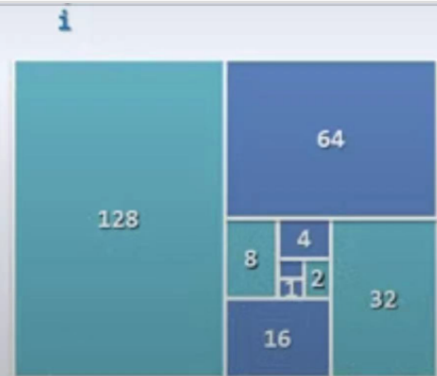
```
        O1Operation(i, j);
```

几何级数：

$$1 + 2 + 4 + \dots + 2^{\lfloor \log_2 (n-1) \rfloor}$$

$$= \sum_{k=0}^{\lfloor \log_2 (n-1) \rfloor} 2^k \quad (\text{let } k = \log_2 i)$$

$$= 2^{\lceil \log_2 n \rceil} - 1 = O(n)$$



Homework

```
for (int i = 0 ; i <= n ; i++)
    for (int j = 0 ; j < i ; j+=j)
        O1Operation(i,j);
```

循环 vs. 级数

```
❖ for (int i = 0; i <= n; i++)
```

```
    for (int j = 1; j < i; j += j)
```

```
        O1Operation(i, j);
```

几何级数：  $\sum_{k=0}^n \lceil \log_2 i \rceil = O(n \log n)$

(i = 0, 1, 2, 3~4, 5~8, 9~16, ...)

$$= 0 + 0 + 1 + 2*2 + 3*4 + 4*8 + \dots$$

$$= \sum_{k=0}^{\log n} (k * 2^{k-1})$$

$$= O(\log n * 2^{\log n}) \quad (\text{CM page\#33})$$

