

# 算法与数据结构体系课程

liuyubobobo

# 基础排序算法

liuyubobobo

# 基础排序算法

排序算法：让数据有序

排序算法中蕴含着重要的算法设计思想

选择排序法

插入排序法

# 选择排序法

liuyubobobo

# 选择排序法

先把最小的拿出来

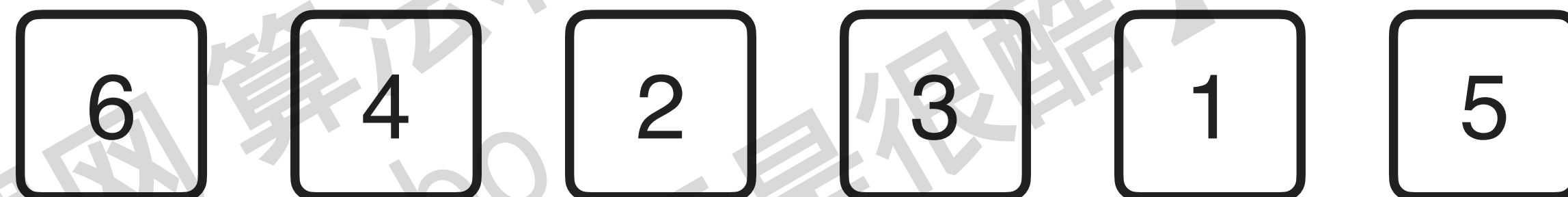
剩下的，再把最小的拿出来

剩下的，再把最小的拿出来

.....

每次选择还没处理的元素里最小的元素

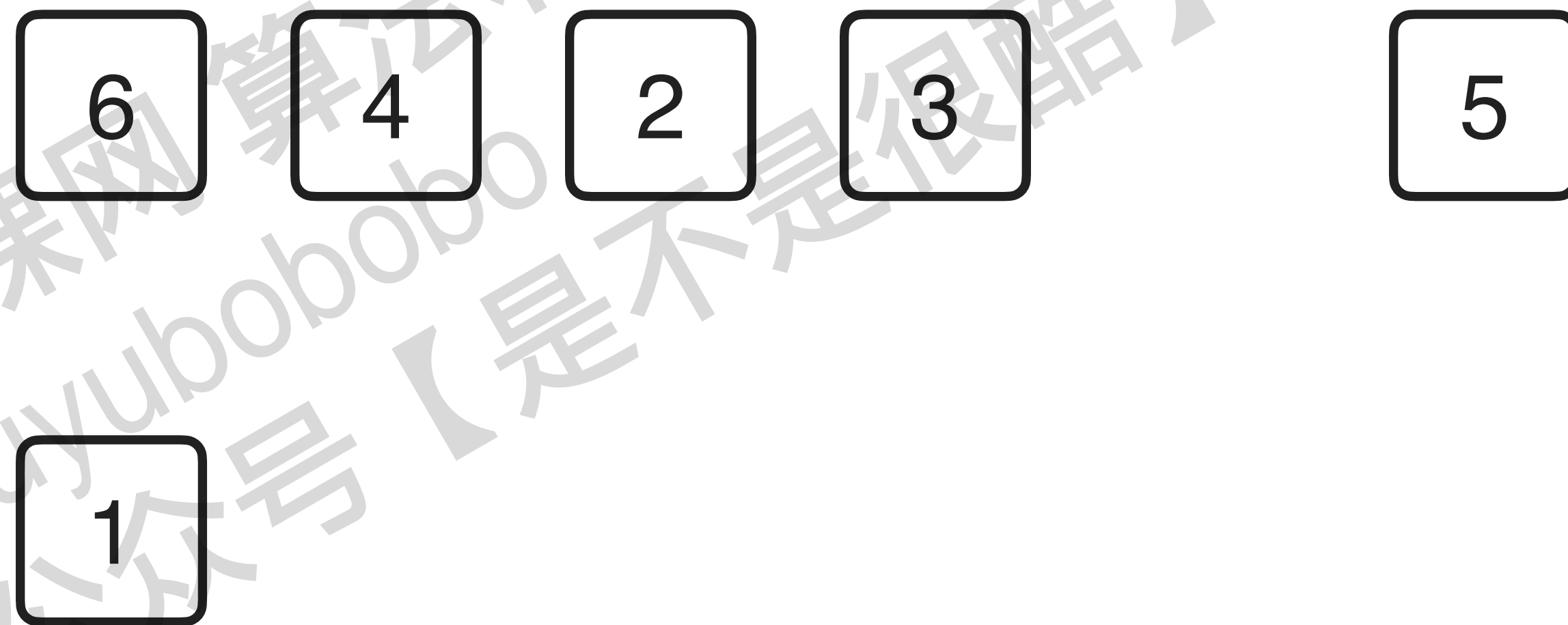
# 选择排序法



# 选择排序法

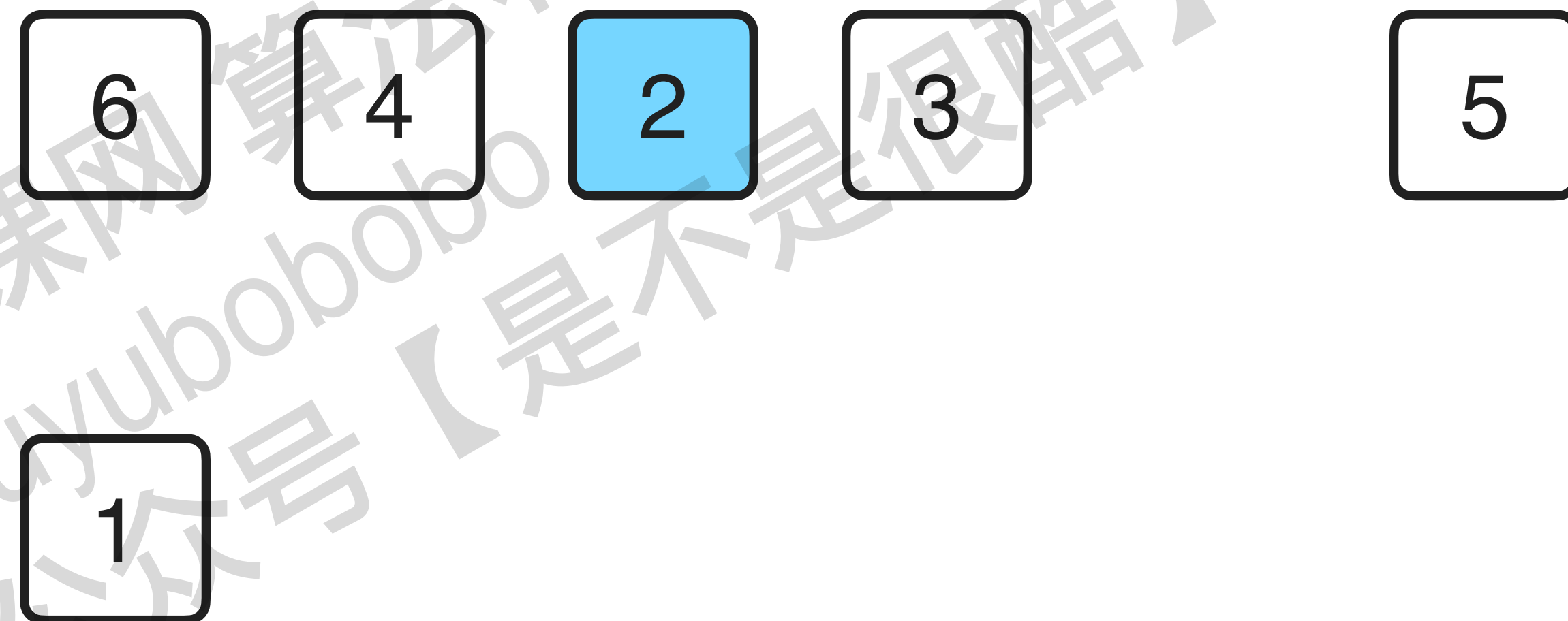


# 选择排序法





# 选择排序法



# 选择排序法

6

4

3

5

1

2

# 选择排序法

6

4

3

5

1

2

# 选择排序法

6

4

5

1

2

3

# 选择排序法

6

4

5

1

2

3

# 选择排序法

6

5

1

2

3

4

# 选择排序法

6

5

1

2

3

4

# 选择排序法

6

1

2

3

4

5



# 选择排序法

6

1

2

3

4

5

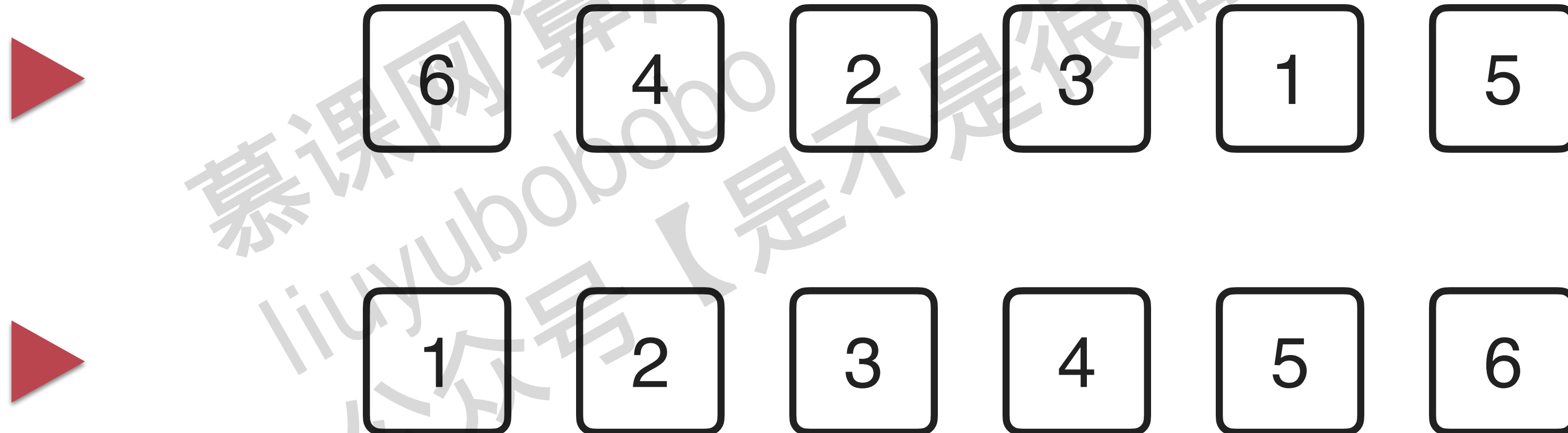
# 选择排序法

1 2 3 4 5 6

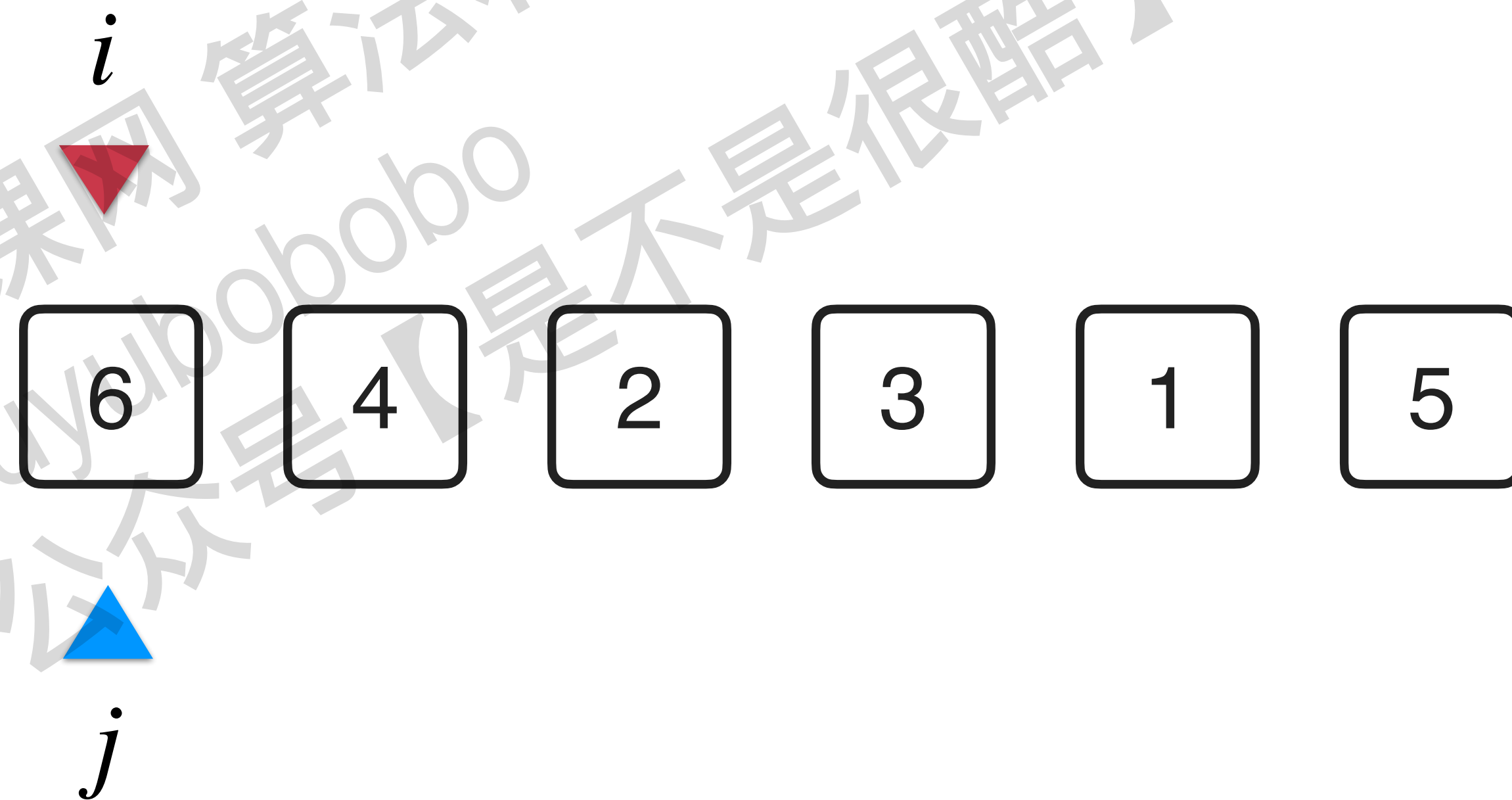
# 选择排序法

排序过程占用了额外的空间

可否原地完成? 原地排序



# 选择排序法



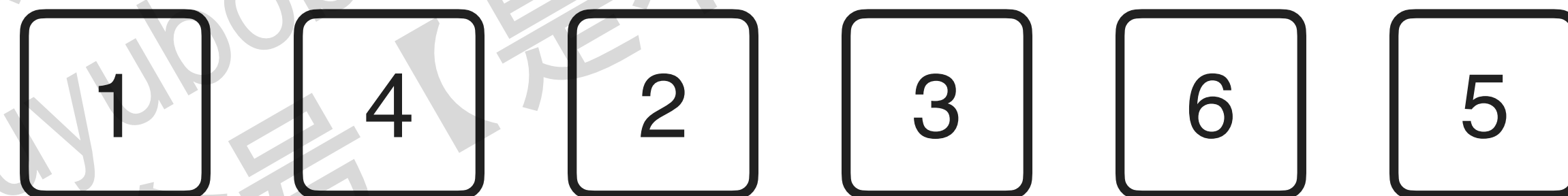
# 选择排序法



# 选择排序法



# 选择排序法



# 选择排序法

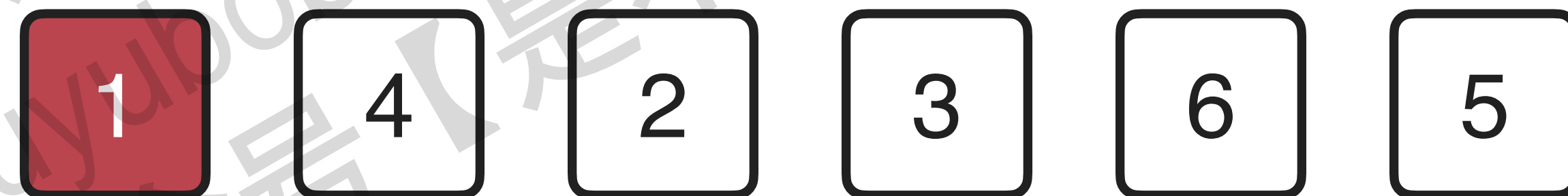




# 选择排序法

$\text{arr}[i \dots n)$  未排序

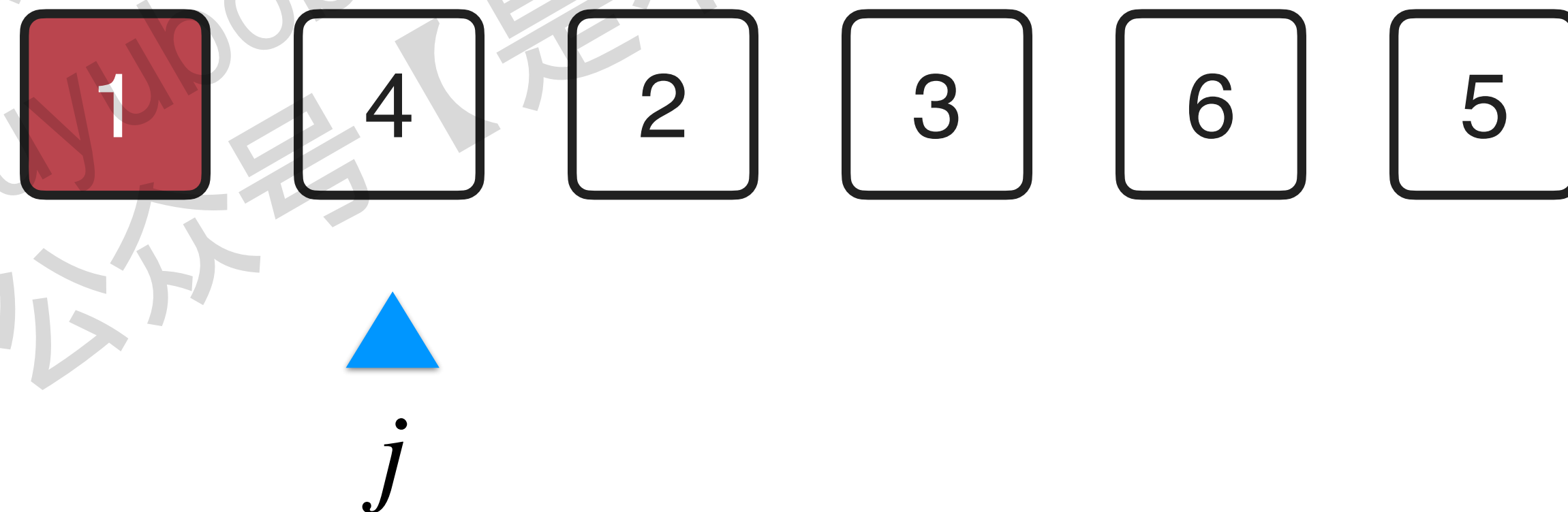
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

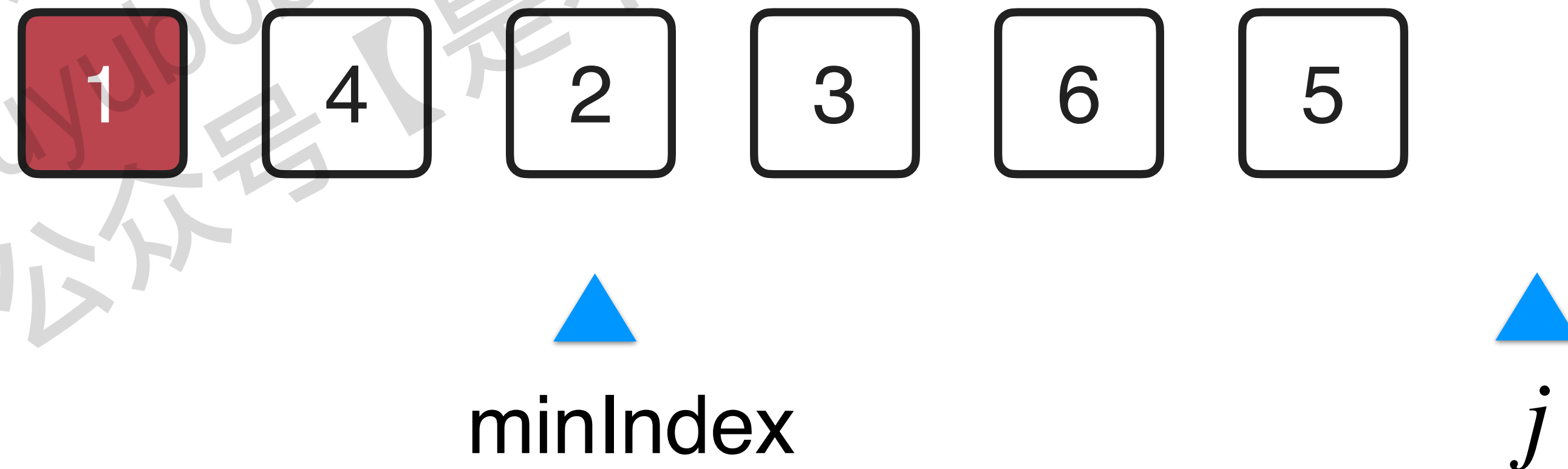
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

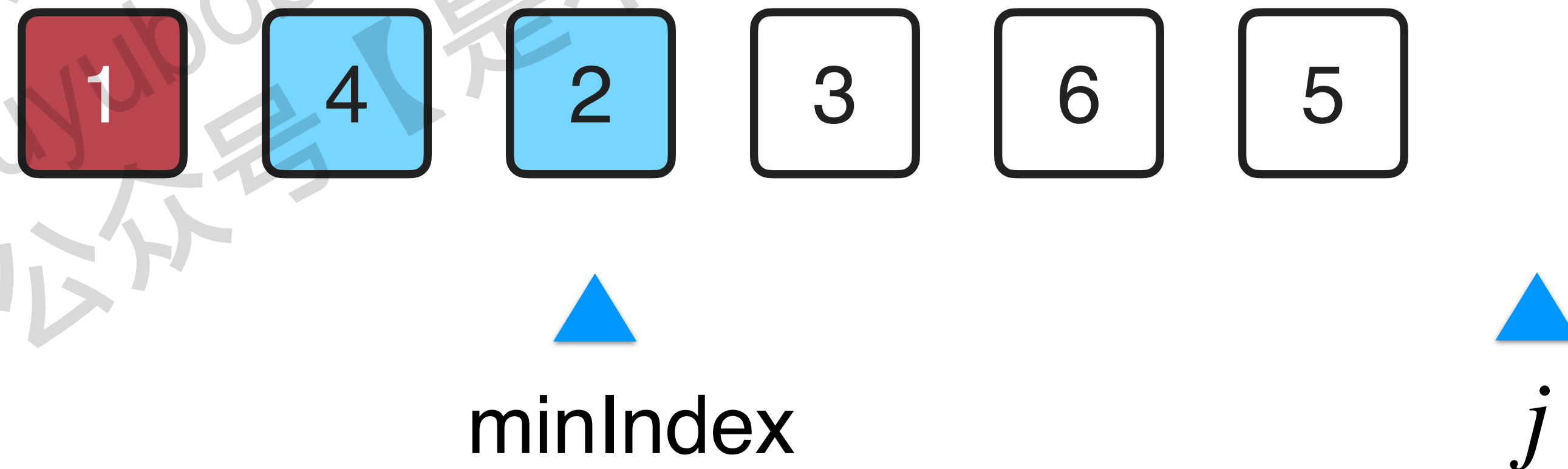
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

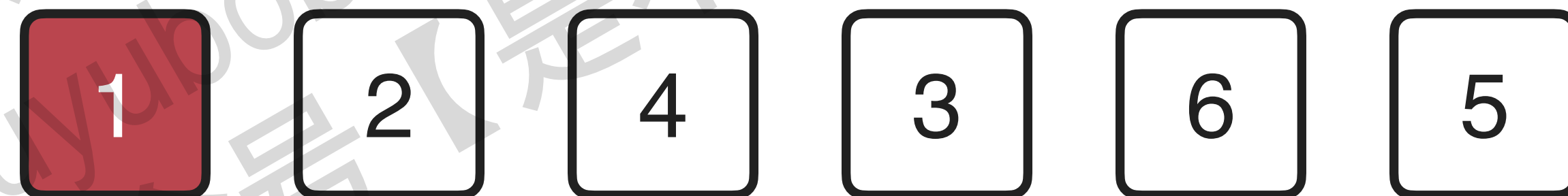
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

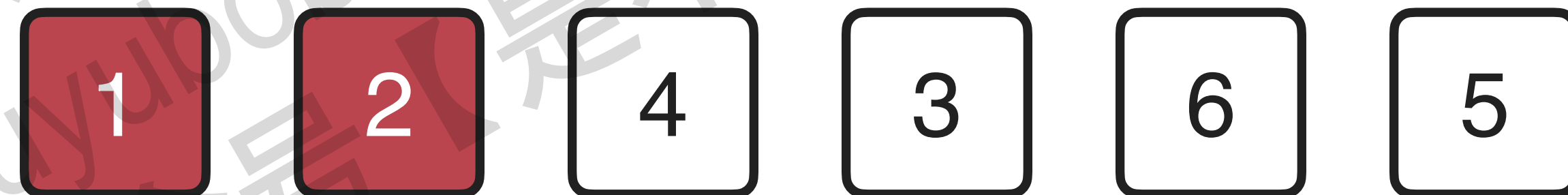
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

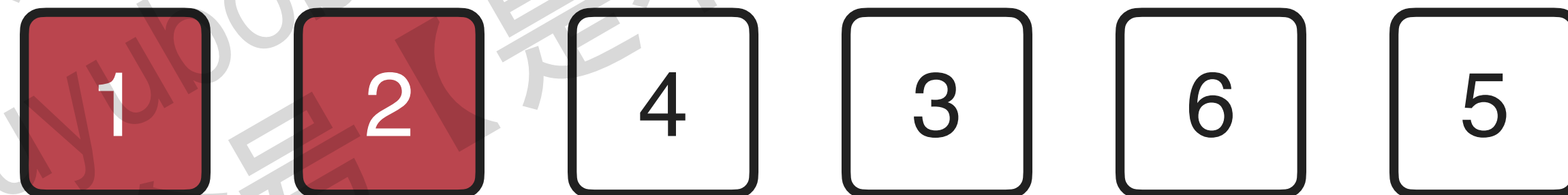
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

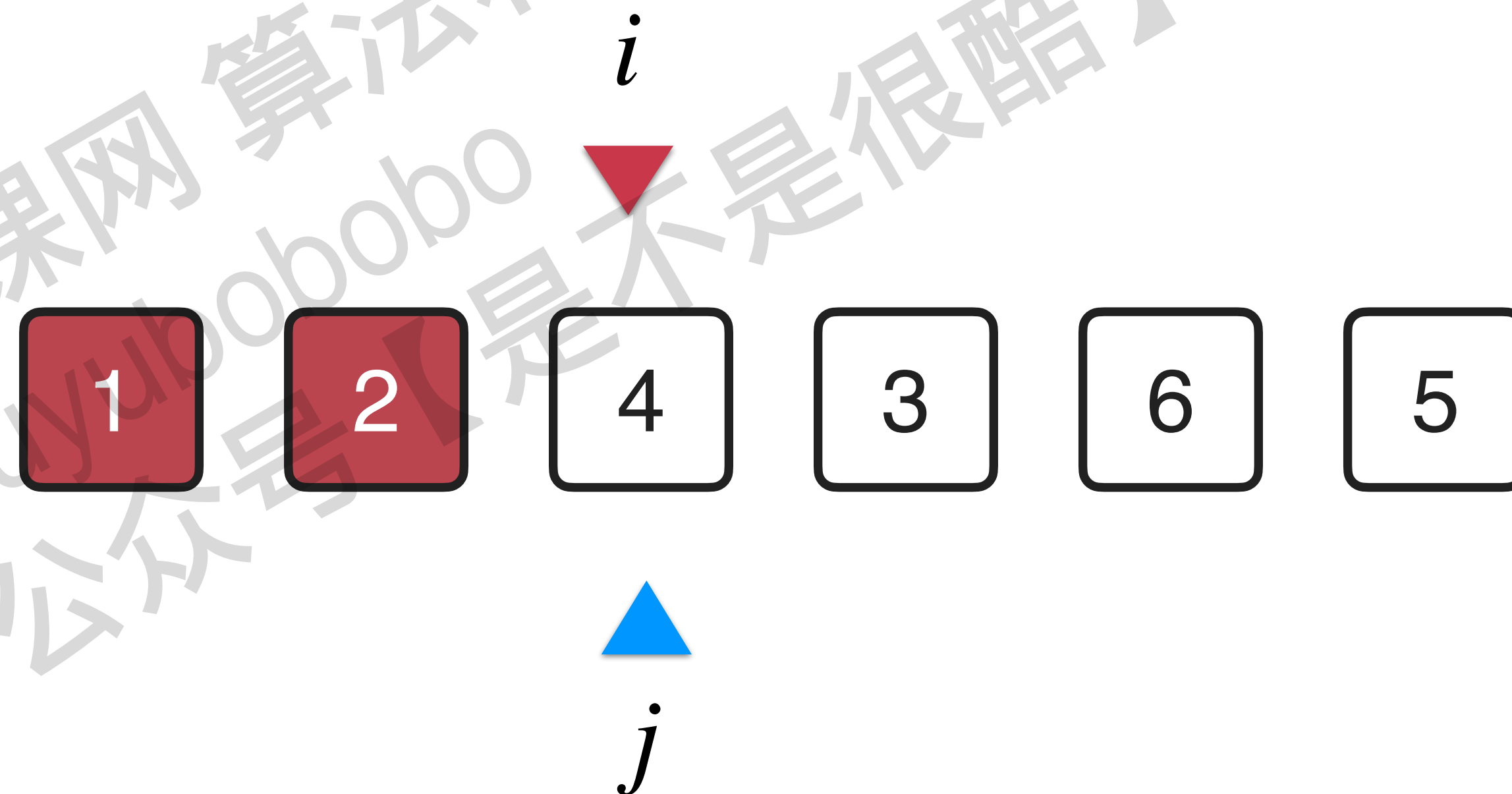
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置

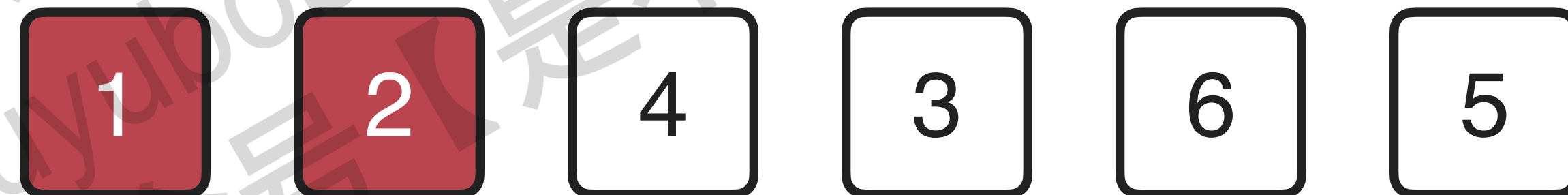




# 选择排序法

$\text{arr}[i \dots n)$  未排序

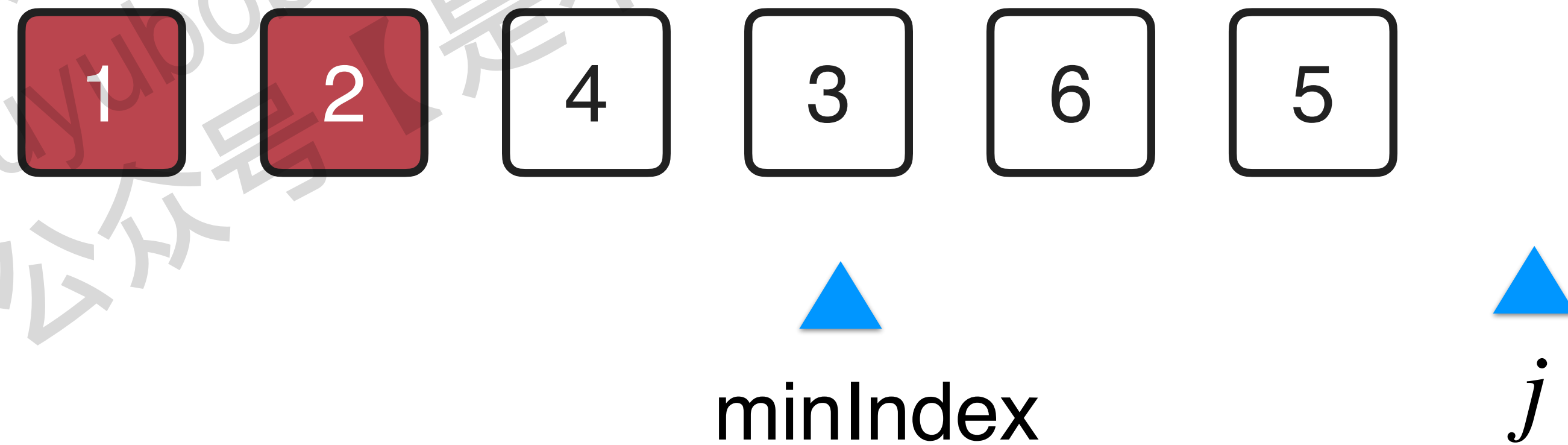
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

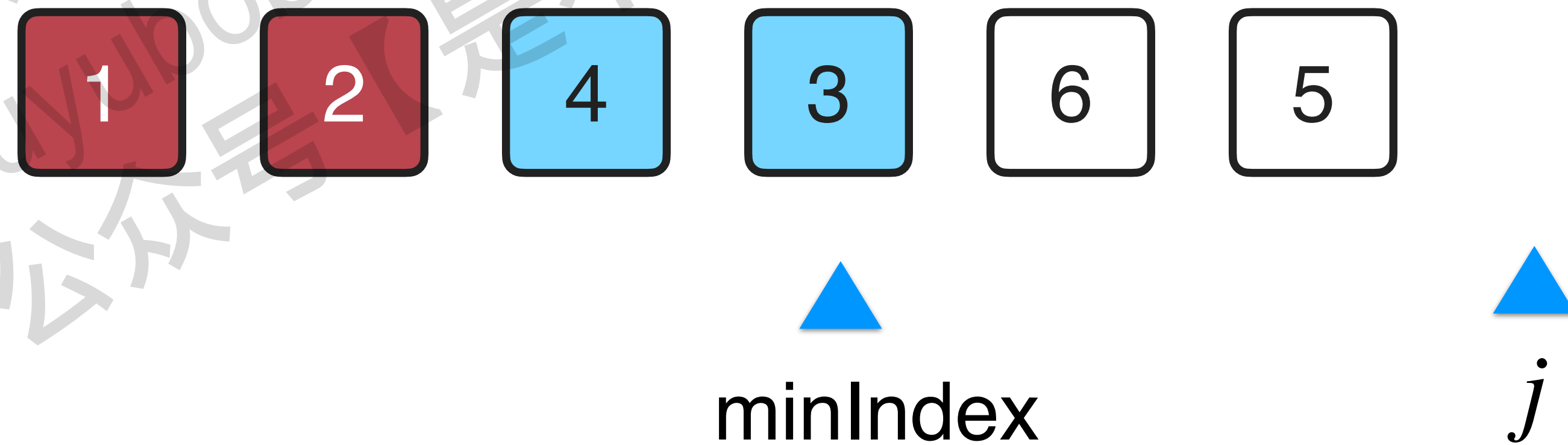
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

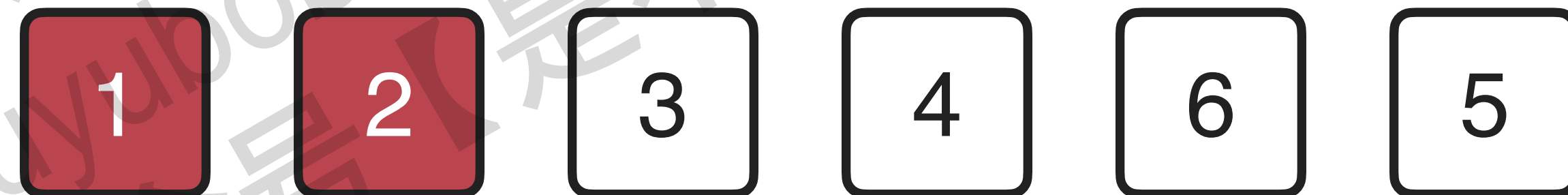
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

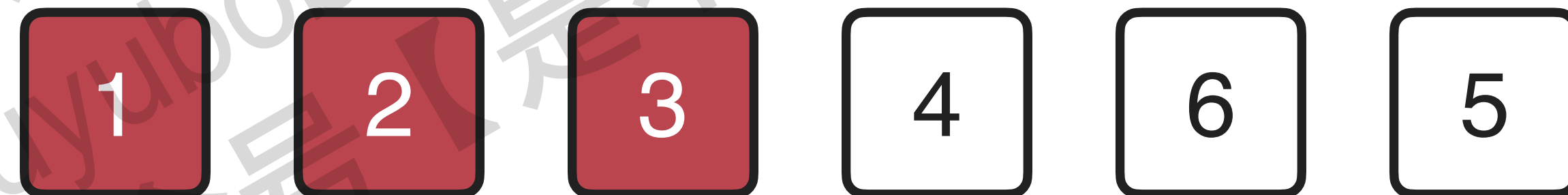
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

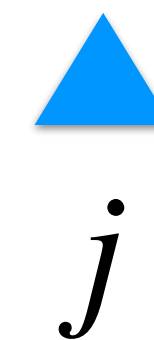
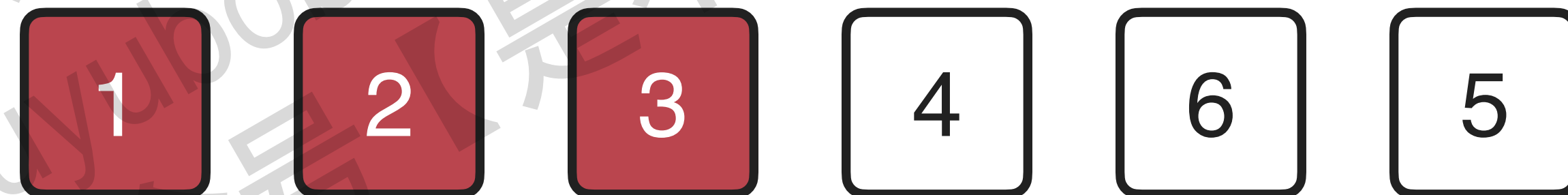
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

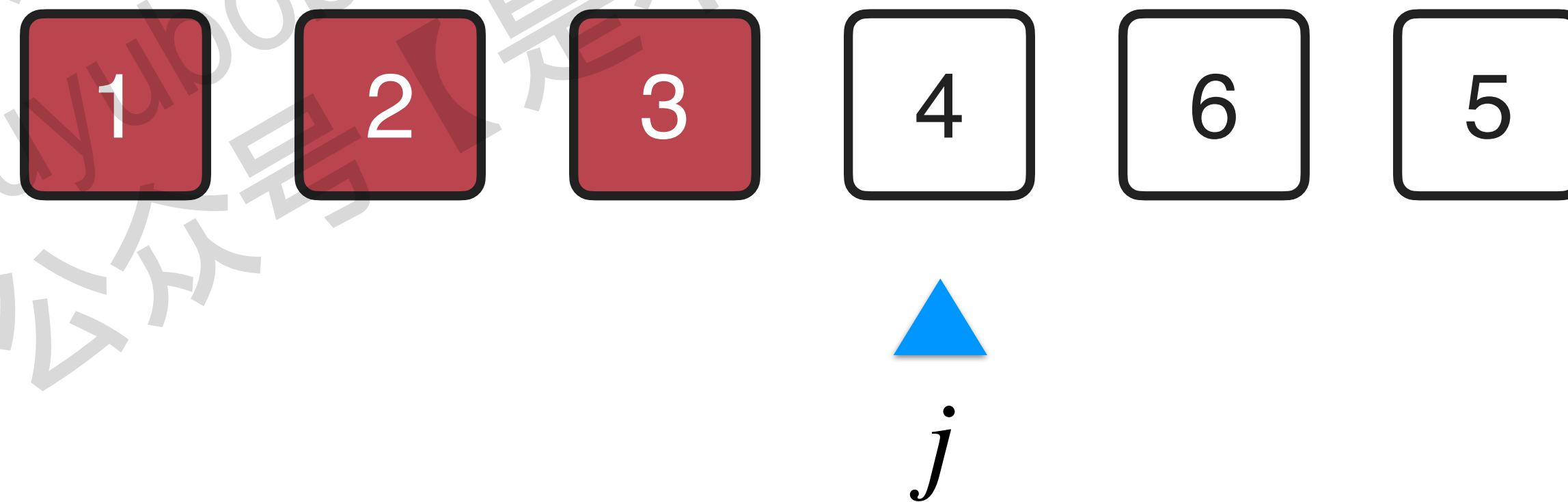
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

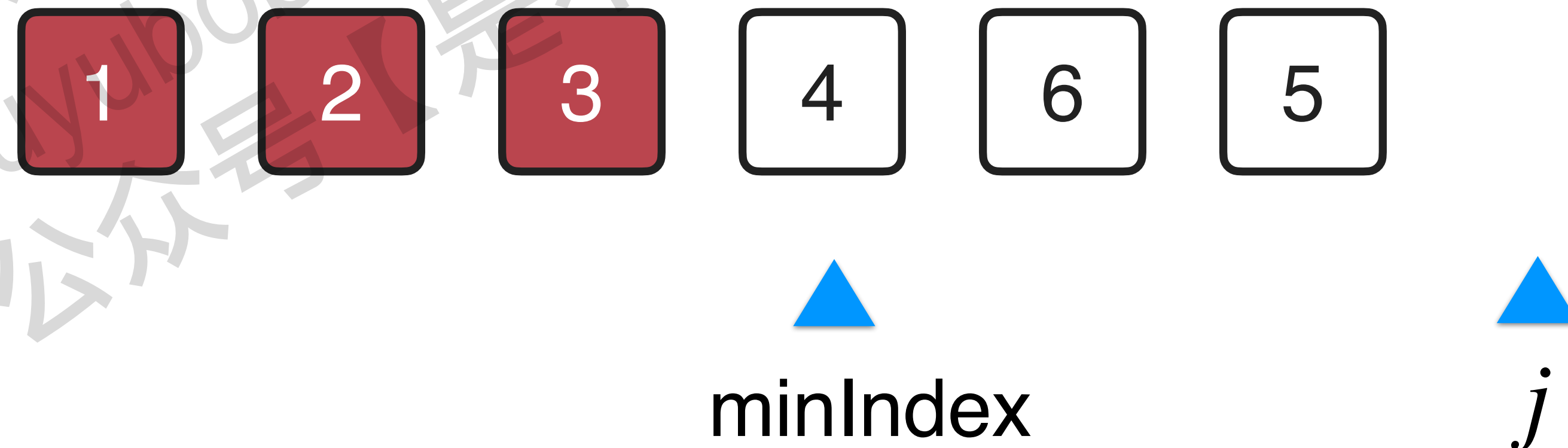
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置

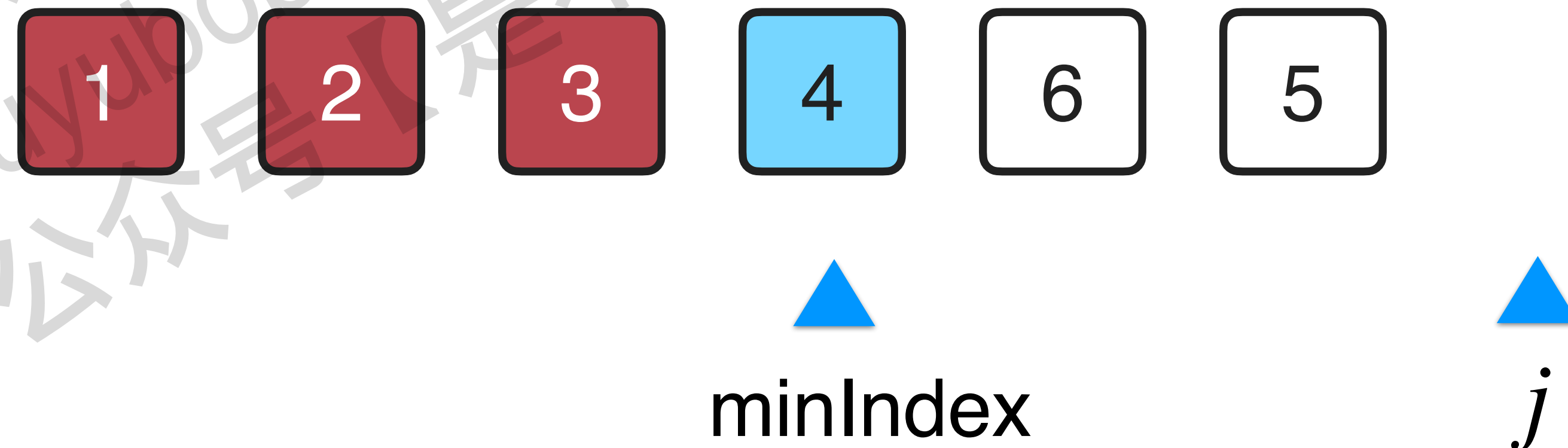




# 选择排序法

$\text{arr}[i \dots n)$  未排序

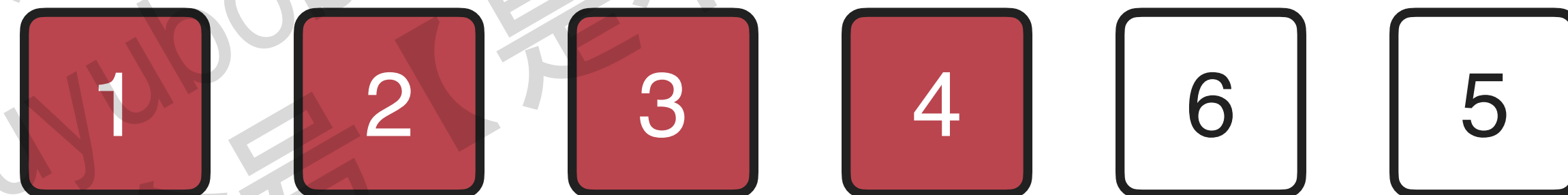
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

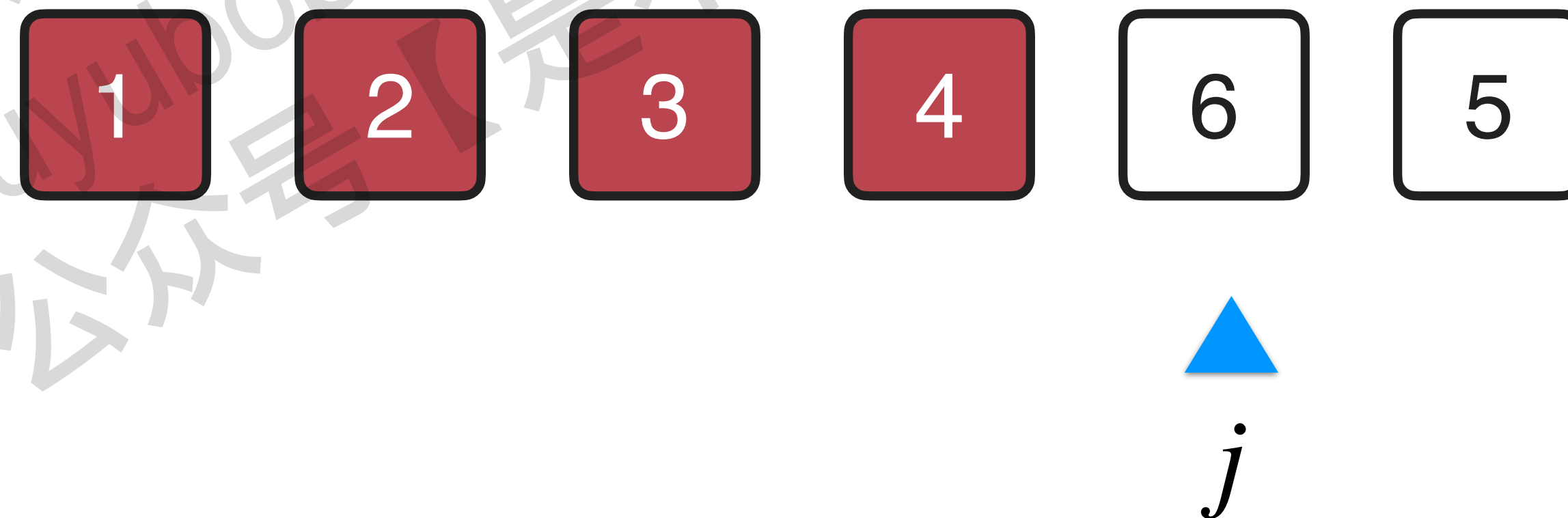
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序

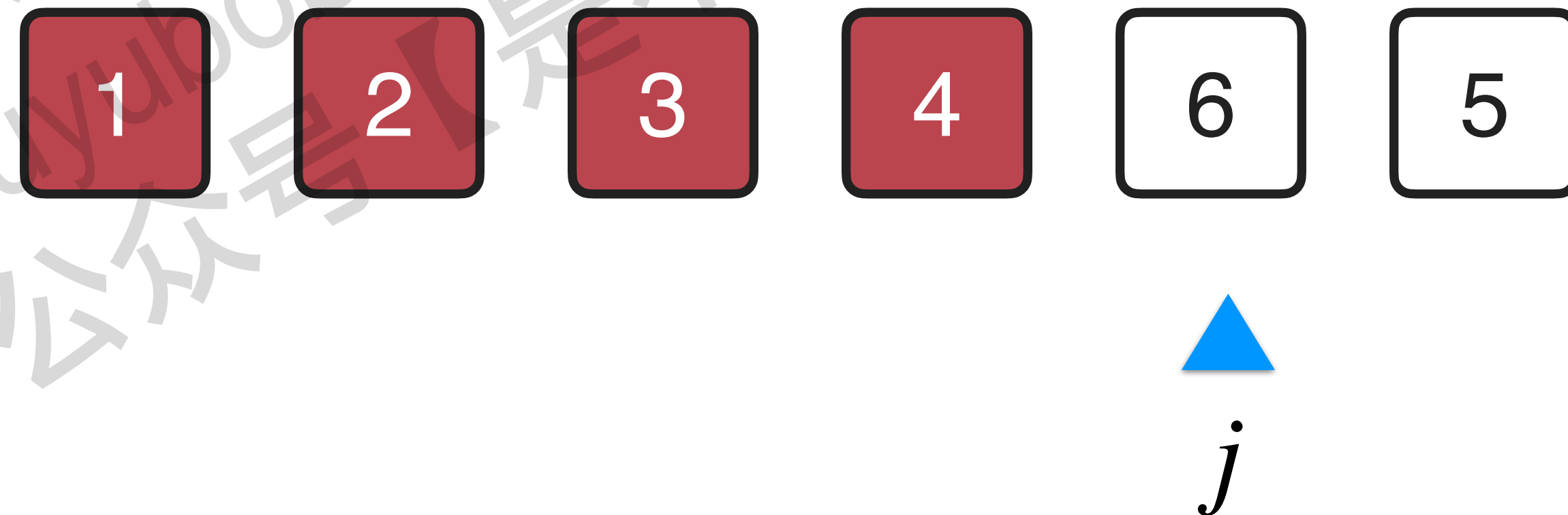
$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 选择排序法

$\text{arr}[i \dots n)$  未排序     $\text{arr}[0 \dots i)$  已排序    循环不变量

$\text{arr}[i \dots n)$  中的最小值要放到  $\text{arr}[i]$  的位置



# 实现选择排序法

liuyubobobo

# 实现选择排序法

实现选择排序法

慕课网 算法和数据结构 体系课程  
liuyubobobo  
公众号【是不是很有趣】

# 使用泛型

liuyubobobo

# 使用泛型

使用泛型

慕课网 算法和数据结构 体系课程  
liuyubobobo  
公众号【是不是很酷】



# 在算法中使用自定义类

liuyubobobo

# 在算法中使用自定义类

设计 Student 类

慕课网 算法和数据结构 体系课程  
liuyubobobo  
公众号【是不是很棒】

# 选择排序法的复杂度分析

liuyubobobo

# 选择排序法的复杂度分析

```
for(int i = 0; i < arr.length; i++){  
    // 选择 arr[i...n) 中的最小值的索引  
    int minIndex = i;  
    for(int j = i; j < arr.length; j++){  
        if(arr[j].compareTo(arr[minIndex]) < 0)  
            minIndex = j;  
    }  
    swap(arr, i, minIndex);  
}
```

$$1 + 2 + 3 + \dots + n$$

$$= \frac{(1 + n) * n}{2}$$

$$= \frac{1}{2}n^2 + \frac{1}{2}n$$

$$O(n^2)$$

# 选择排序法的复杂度分析

选择排序法性能测试代码

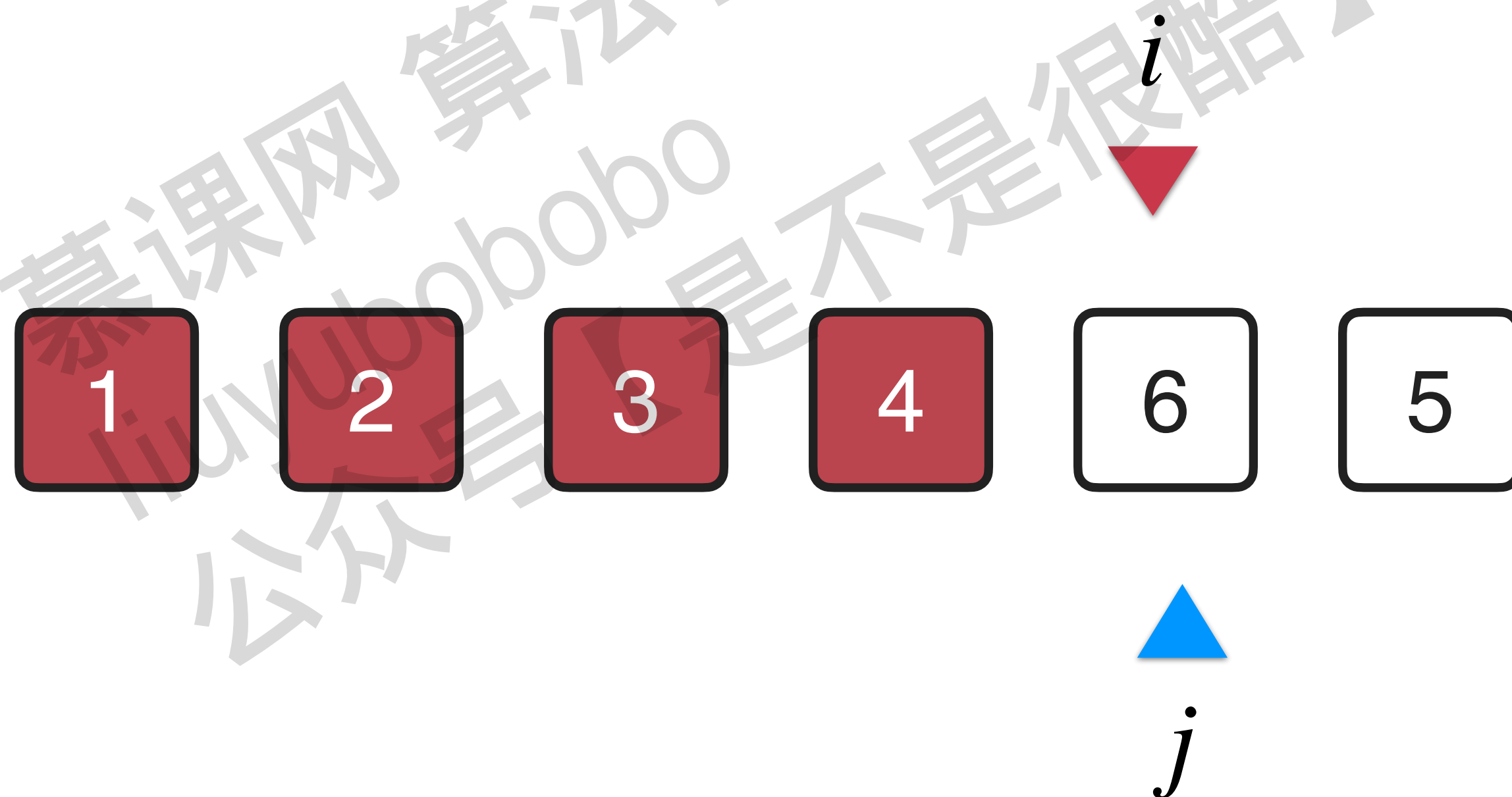
作业：换个方式实现选择排序法？

liuyubobobo

# 换个方式实现选择排序法?

$\text{arr}[i \dots n)$  未排序     $\text{arr}[0 \dots i)$  已排序    循环不变量

$\text{arr}[i \dots n)$  已排序     $\text{arr}[0 \dots i)$  未排序



# 换个方式实现选择排序法?

$\text{arr}[i \dots n)$  已排序     $\text{arr}[0 \dots i)$  未排序    循环不变量



# 解析：换个方式实现选择排序法

liuyubobobo

# 其他

欢迎大家关注我的个人公众号：是不是很酷



# 算法与数据结构体系课程

liuyubobobo