

算法与数据结构体系课程

liuyubobobo

二分查找法

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

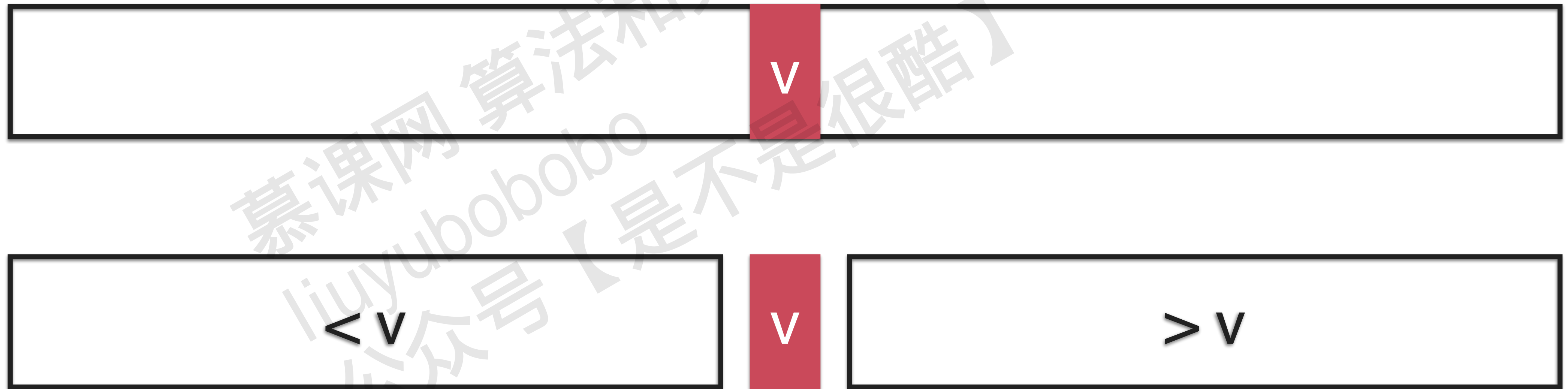
查找问题是计算机中非常重要的基础问题

二分查找法

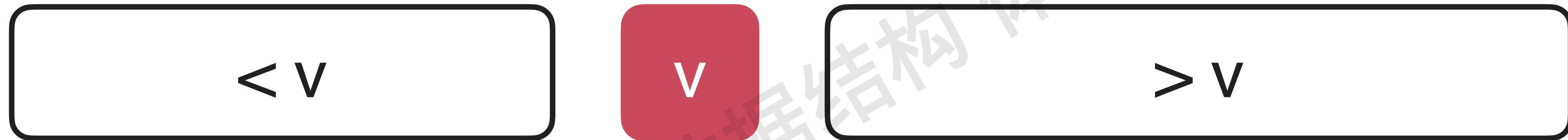
慕课网 算法和数据结构 体系课程
liuyubobobo
公众号【是不是很有趣】

二分查找法

对于有序数列，才能使用二分查找法（排序的作用）



Select K



p

k == p?

找到了!

k < p?

到左边找

k > p?

到右边找

Select K vs 二分查找

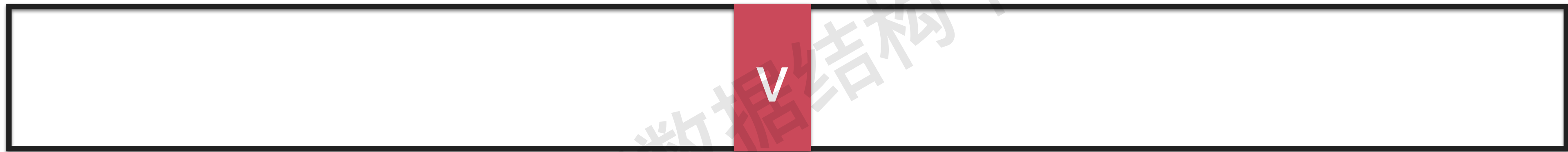


$$n + n/2 + n/4 + \dots + 1 = 2n = O(n)$$



$$1 + 1 + 1 + \dots + 1 = \log n = O(\log n)$$

二分查找法



$$1 + 1 + 1 + \dots + 1 = \log n = O(\log n)$$

我们没有把排序时间算进去

如果计算排序时间： $O(n \log n)$

排序叫做二分查找法的前置条件

应用： 多次查找

二分查找法 Binary Search

二分查找法的思想在1946年提出。

第一个没有bug的二分查找法在1962年才出现。

$mid = (l + r) / 2$ 可能整型溢出

$mid = l + (r - l) / 2$

递归实现二分查找法

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

实践：递归实现二分查找法

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

非递归实现二分查找法

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

实践：非递归实现二分查找法

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

作业：非递归实现 Select K

非递归实现 Select K

```
private int selectK(int[] arr, int l, int r, int k, Random rnd){  
    int p = partition(arr, l, r, rnd);  
  
    if(k == p)  
        return arr[p];  
  
    if(k < p)  
        return selectK(arr, l, p - 1, k, rnd);  
  
    return selectK(arr, p + 1, r, k, rnd);  
}
```

修改循环不变量定义 实现二分查找法

实践： 修改循环不变量定义
实现二分查找法

作业：修改循环不变量定义
重写算法

Select K

~~在 $\text{arr}[l, r]$ 的范围内寻找第 k 小元素~~

在 $\text{arr}[l, r)$ 的范围内寻找第 k 小元素

private int selectK(**int**[] arr, **int** l, **int** r, **int** k, Random rnd)

归并排序

~~合并 arr[l, mid] 和 arr[mid + 1, r]~~

合并 arr[l, mid) 和 arr[mid, r)

private static <E extends Comparable<E>> **void** merge(E[] arr, **int** l, **int** mid, **int** r, E[] aux)

~~对 arr[l, r] 进行归并排序~~

对 arr[l, r) 进行归并排序

private static <E extends Comparable<E>> **void** sort(E[] arr, **int** l, **int** r, E[] temp)

作业解析：修改循环不变量 重写 Select K

作业解析：修改循环不变量 重写归并排序算法

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