

排序

快速排序：快速划分：LUG版

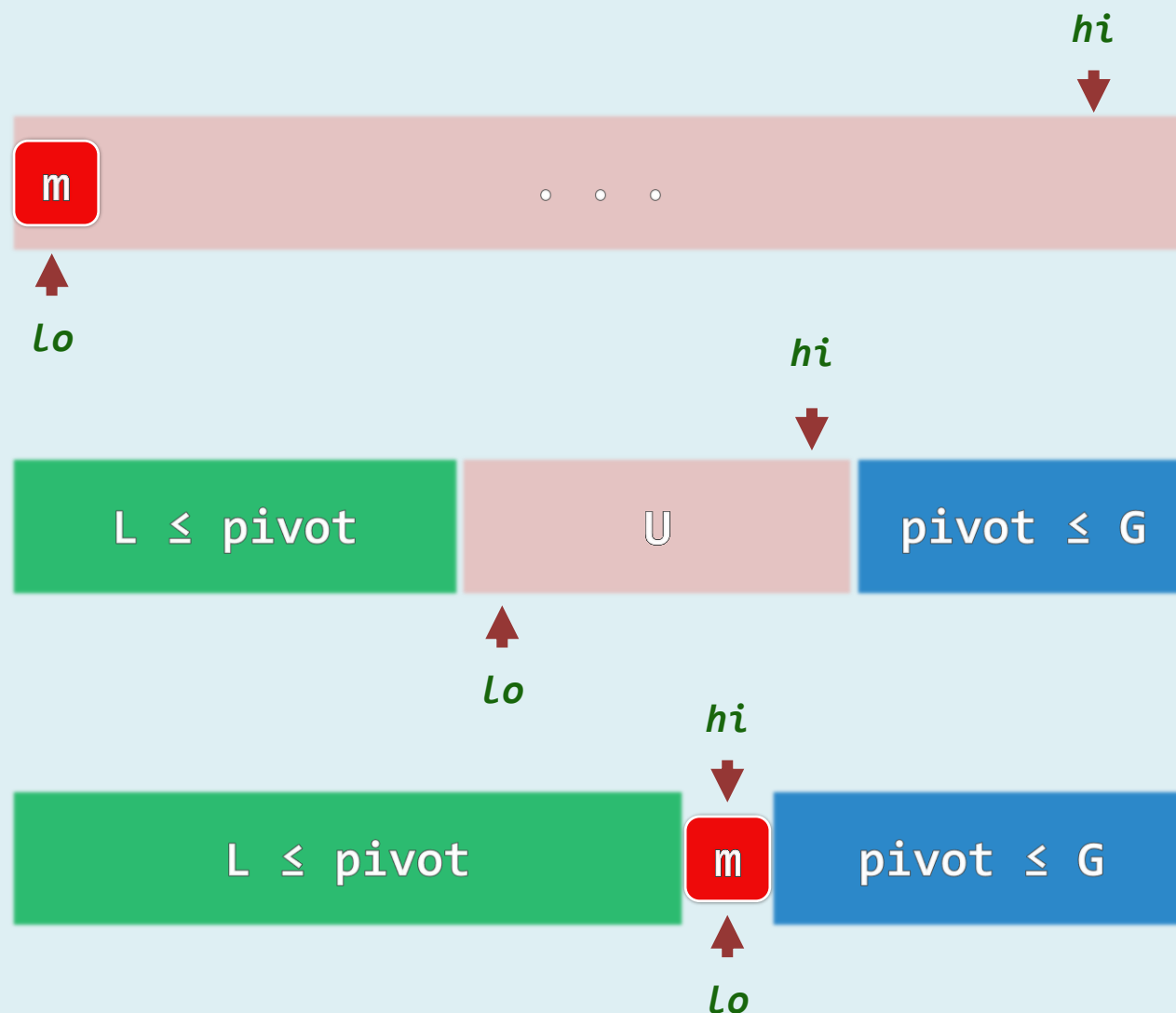
14-A2

邓俊辉

deng@tsinghua.edu.cn

# 减而治之，相向而行

- ❖ 任取一个**候选者**（如 $[0]$ ）
- ❖  $L + U + G$
- ❖ **交替地向内**移动 $lo$ 和 $hi$
- ❖ 逐个检查当前元素：  
若更小/大，则转移归入 $L/G$
- ❖ 当 $lo = hi$ 时，只需  
将候选者**嵌入**于 $L$ 、 $G$ 之间，即成轴点！
- ❖ 各元素最多移动一次（候选者两次）  
——累计 $O(n)$ 时间、 $O(1)$ 辅助空间



## 快速划分：LUG版

```
template <typename T> Rank Vector<T>::partition( Rank lo, Rank hi ) { //[lo, hi)

    swap( _elem[ lo ], _elem[ lo + rand() % ( hi - lo ) ] ); //随机交换

    hi--; T pivot = _elem[ lo ]; //经以上交换，等效于随机选取候选轴点

    while ( lo < hi ) { //从两端交替地向中间扫描，彼此靠拢

        while ( lo < hi && pivot <= _elem[ hi ] ) hi--; //向左拓展G

        _elem[ lo ] = _elem[ hi ]; //凡小于轴点者，皆归入L

        while ( lo < hi && _elem[ lo ] <= pivot ) lo++; //向右拓展L

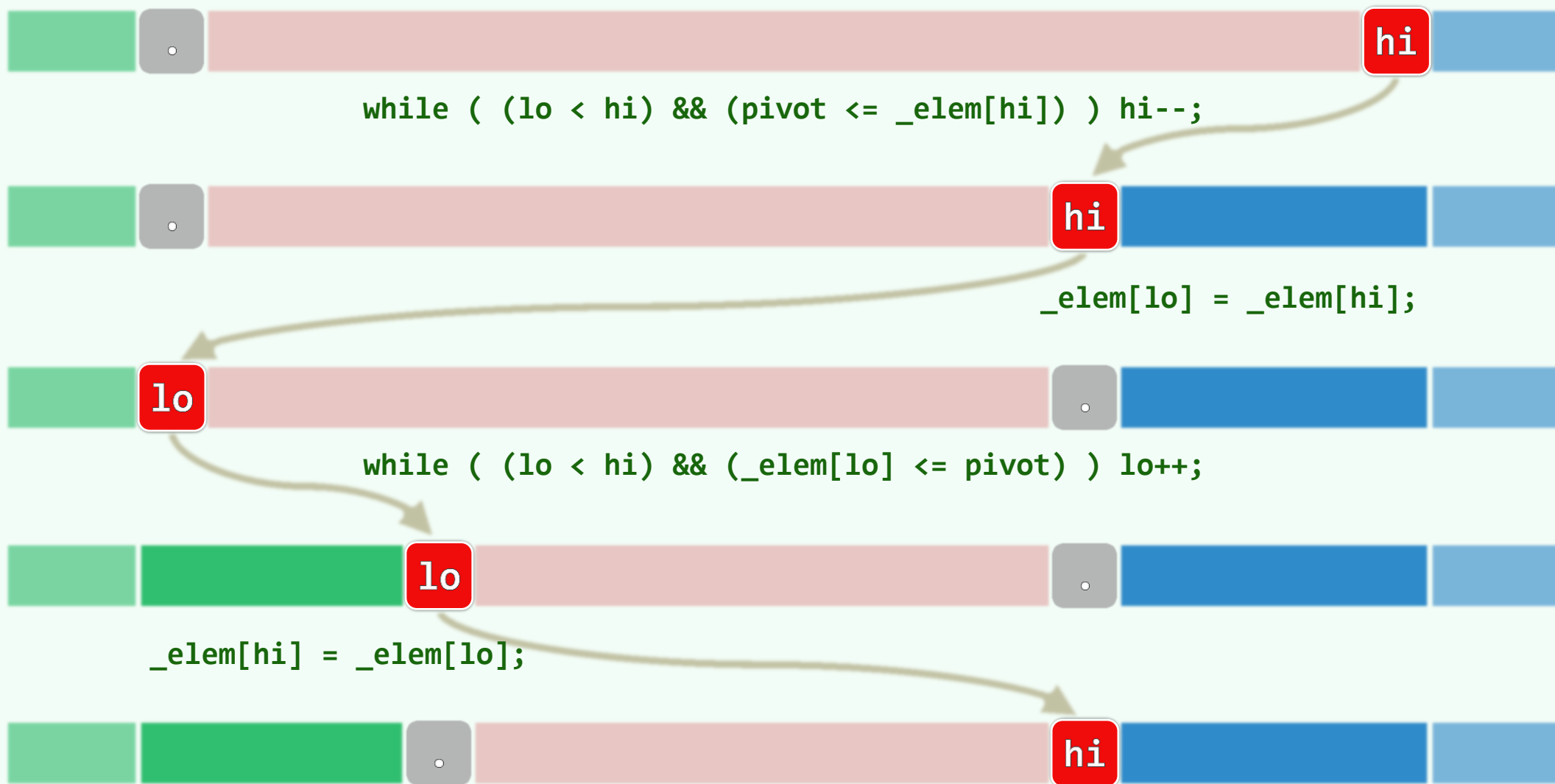
        _elem[ hi ] = _elem[ lo ]; //凡大于轴点者，皆归入G

    } //assert: lo == hi

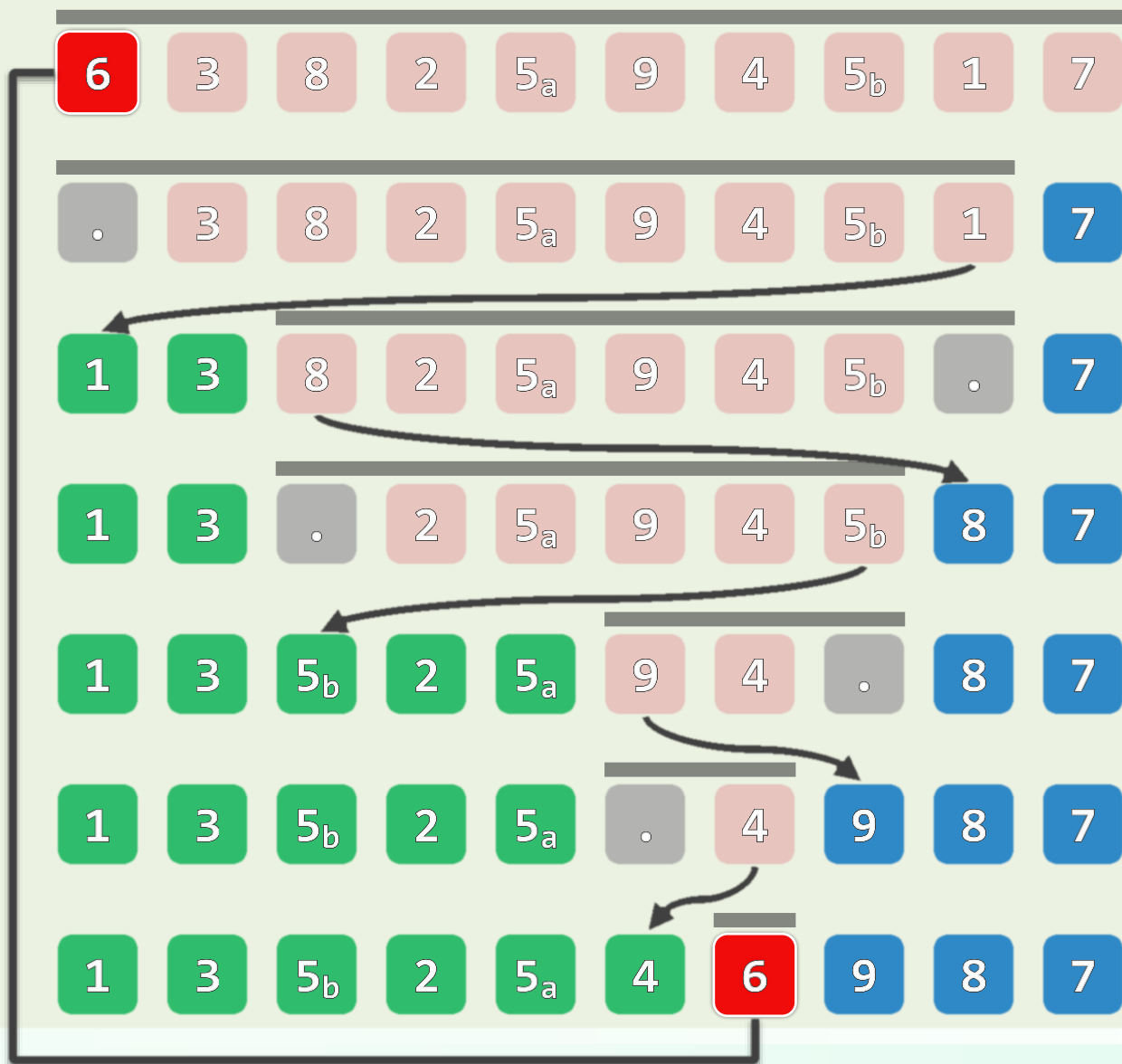
    _elem[ lo ] = pivot; return lo; //候选轴点归位；返回其秩

}
```

# 不变性 + 单调性 : $L \leq \text{pivot} \leq G$ ; $U = [\text{lo}, \text{hi}]$ 中 , $[\text{lo}]$ 和 $[\text{hi}]$ 交替空闲



# 实例



## ❖ 线性时间

- 尽管  $lo$ 、 $hi$  交替移动
- 累计移动距离不过  $O(n)$

## ❖ in-place

- 只需  $O(1)$  附加空间

## ❖ unstable

- $lo/hi$  的移动方向相反
- 左/右侧的大/小重复元素

可能前/后颠倒