

慕课网《看得见的算法》

# 看得见的算法

讲师：liuyubobobo

版权所有，侵权必究

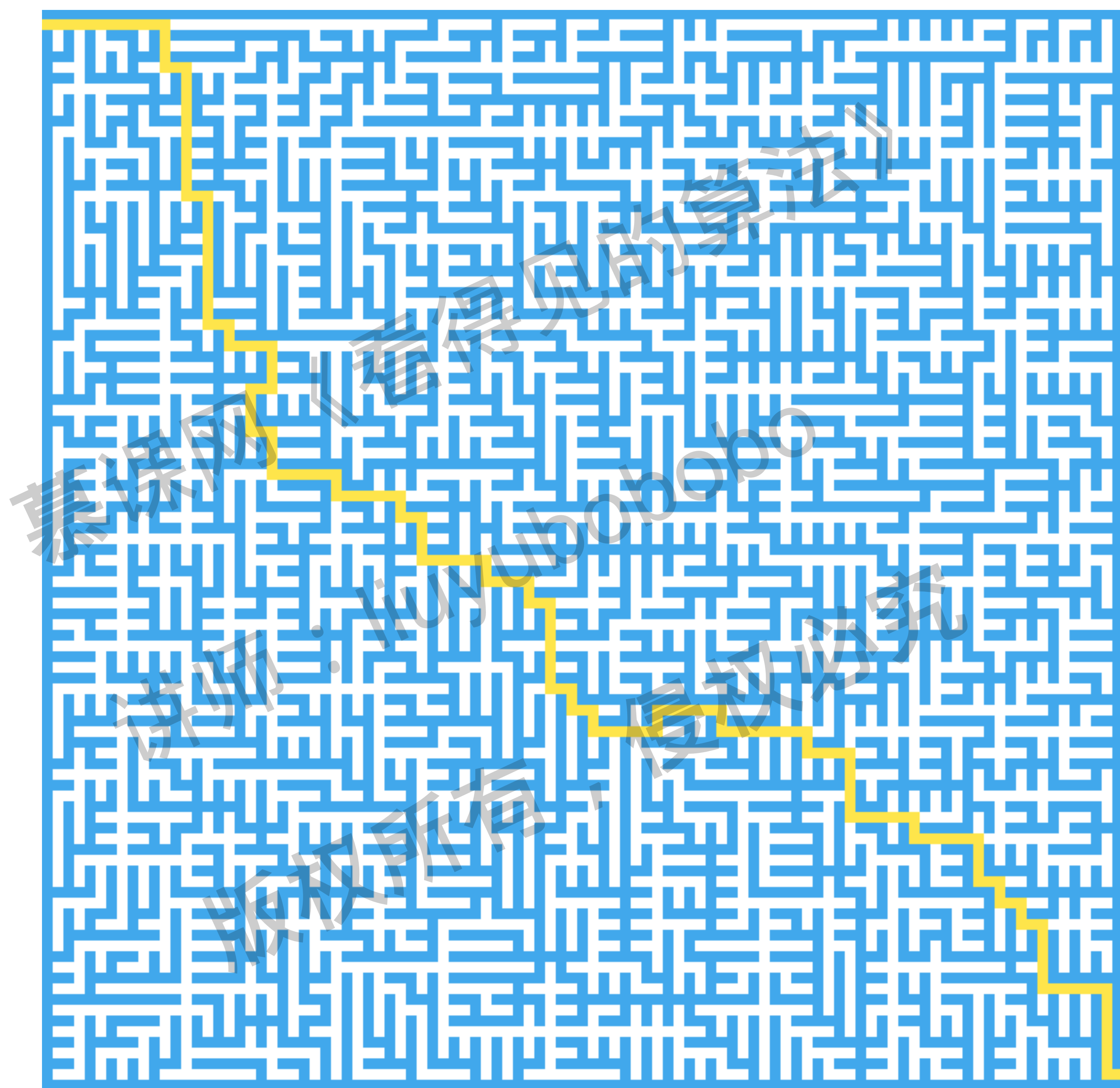
liuyubobobo

慕课网《看得见的算法》

# 迷宫问题求解

讲师：luyubobobo

版权所有，侵权必究



慕课网《看得见的算法》  
讲师：liuyubobobo  
版权所有，侵权必究

慕课网《看得见的算法》

# 迷宫文件的处理

讲师：liuyubobobo

版权所有，侵权必究

# 迷宫文件的处理

演示：迷宫文件的处理

慕课网《看得见的算法》

# 迷宫的可视化

讲师：luyaoobobo

版权所有，侵权必究

# 迷宫的可视化

演示：迷宫的可视化

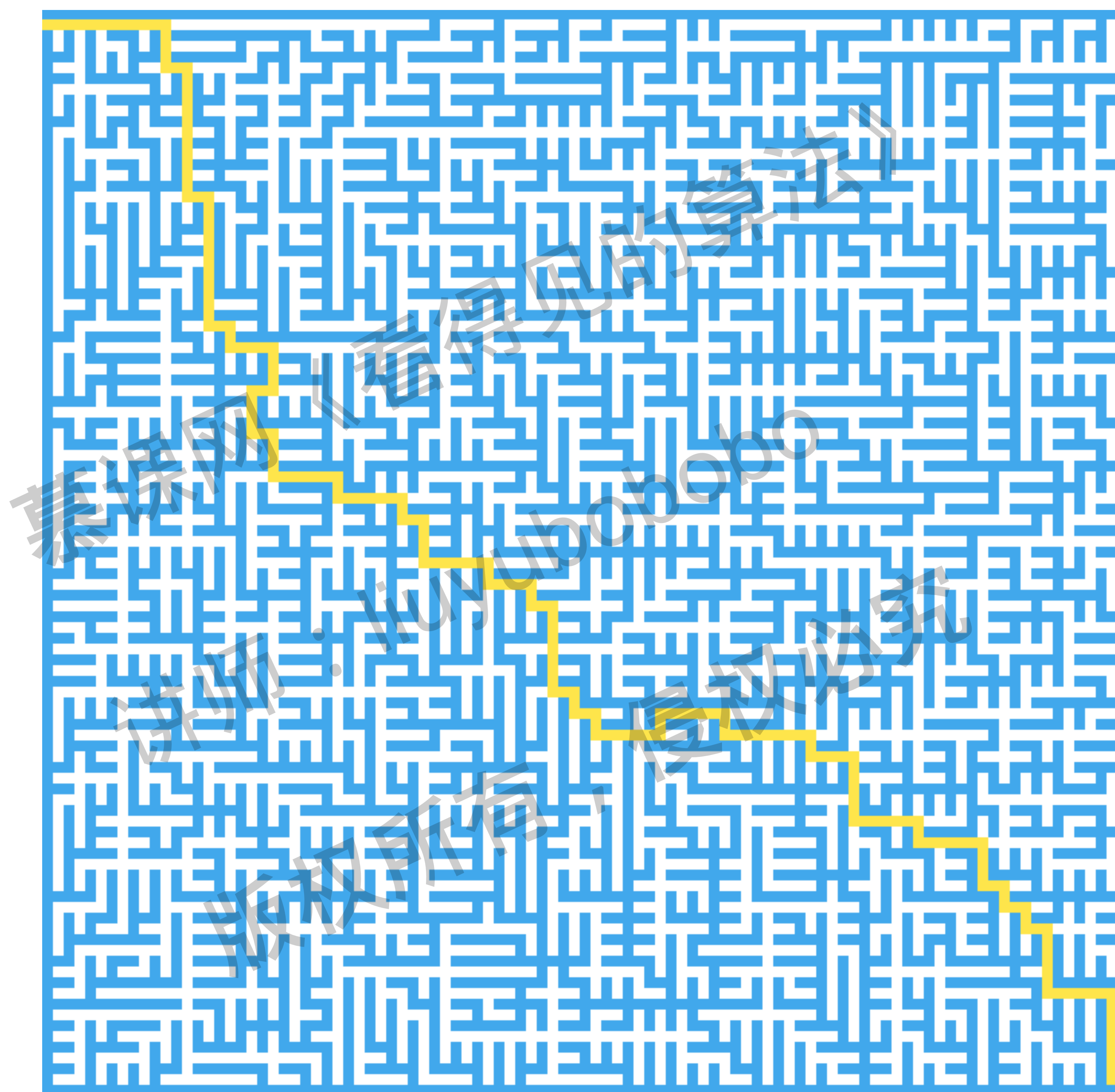
慕课网《看得见的算法》

# 迷宫问题求解

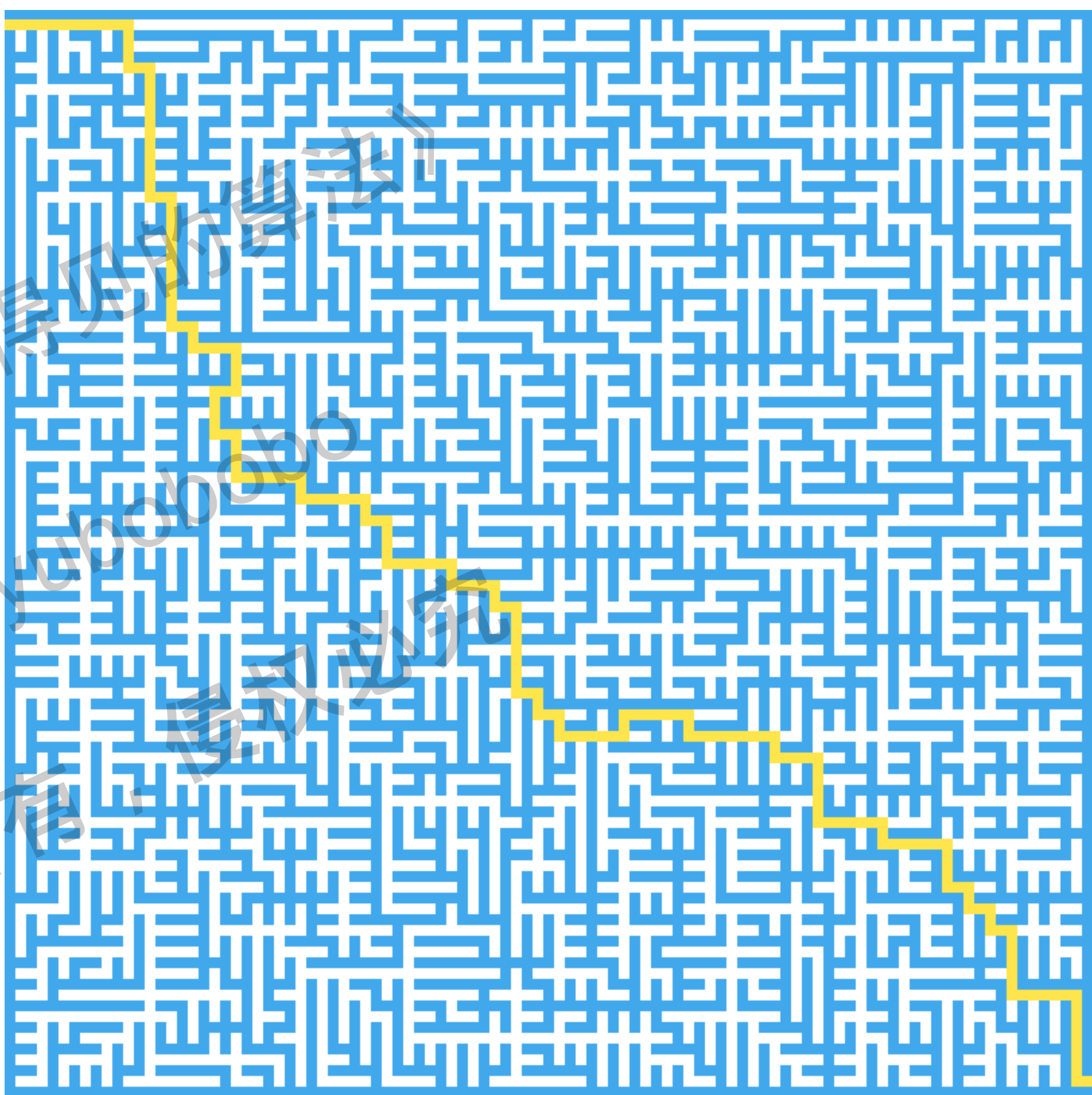
讲师：liuyubobobo

版权所有，侵权必究





迷宫求解问题等价于  
图的遍历

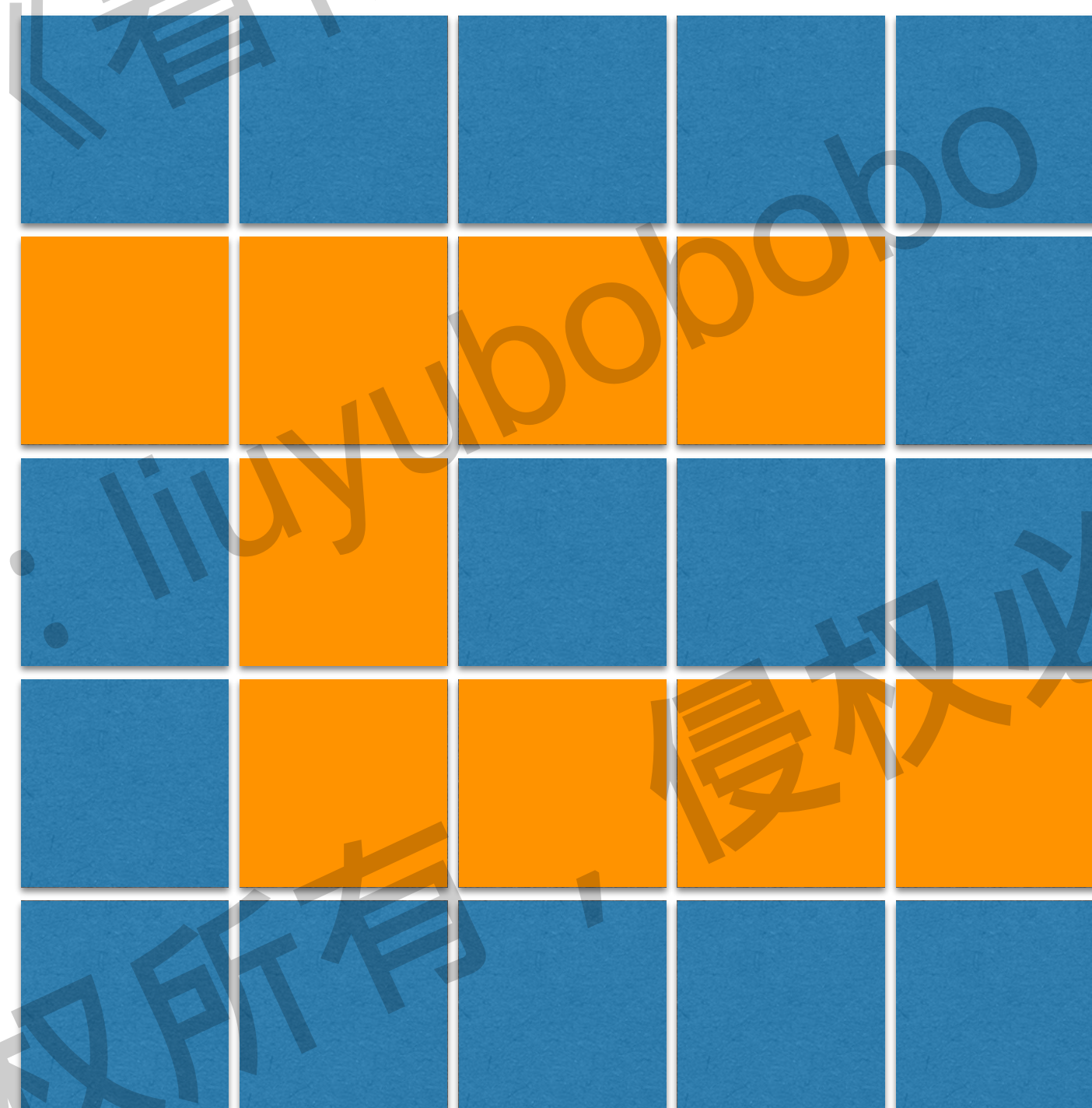


# 图的遍历

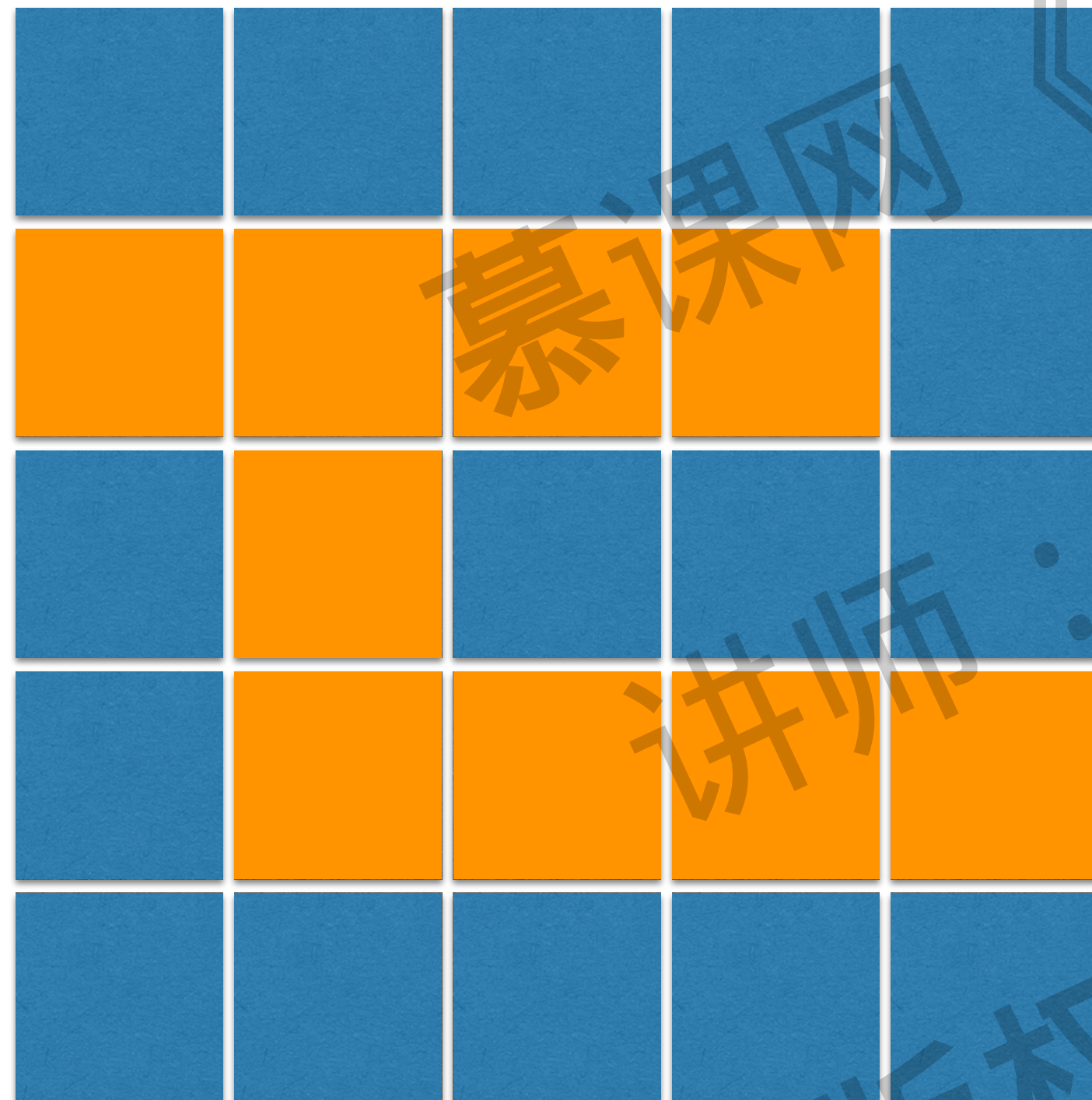
深度优先遍历

广度优先遍历

# 图的深度遍历求解迷宫问题



# 图的深度遍历求解迷宫问题



求解迷宫(x, y):

尝试向上走, 继续求解迷宫

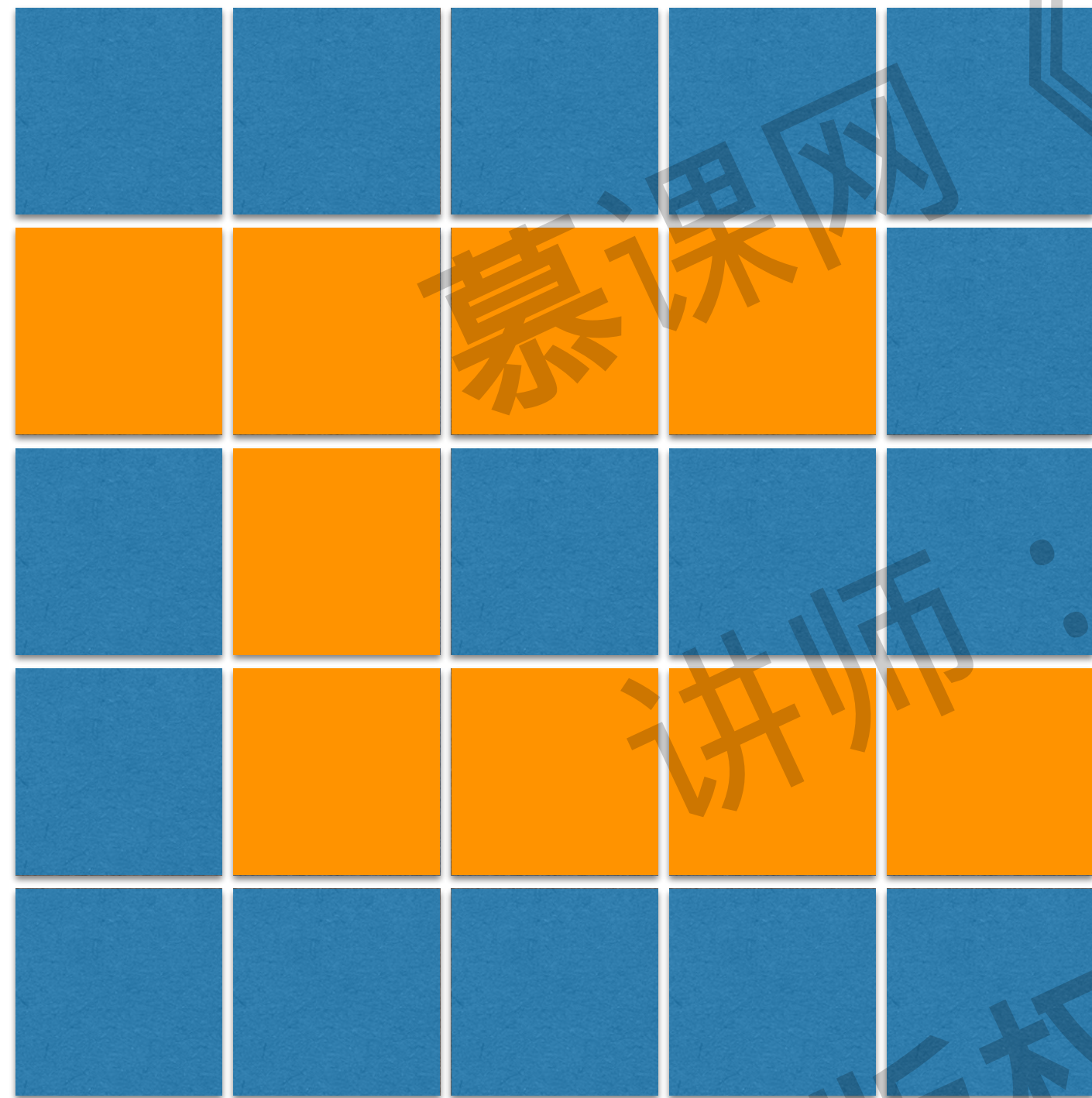
尝试向右走, 继续求解迷宫

尝试向下走, 继续求解迷宫

尝试向左走, 继续求解迷宫



# 图的深度遍历求解迷宫问题



`solveMaze(x, y):`

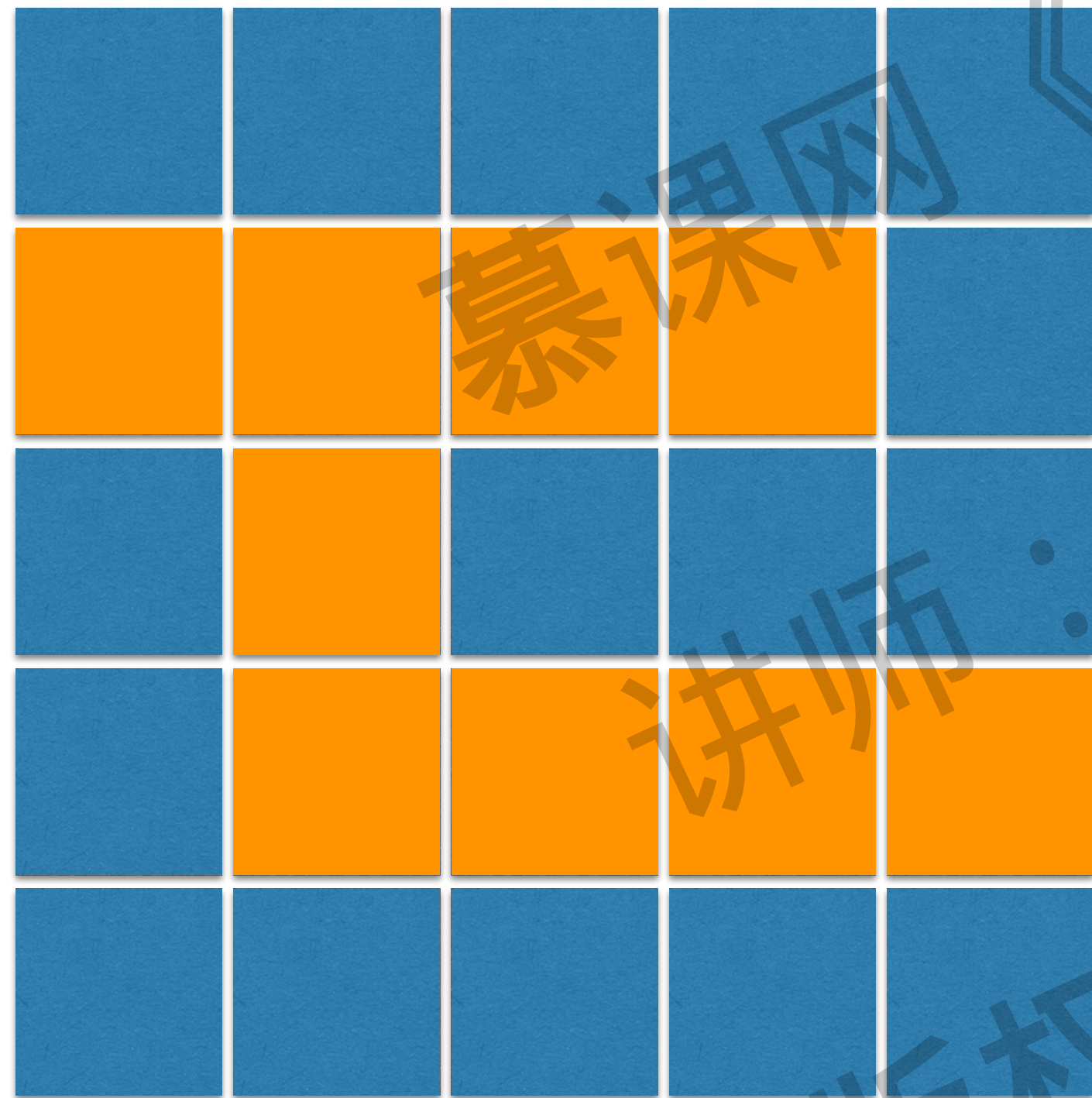
`solveMaze(x, y-1)`

`solveMaze(x+1, y)`

`solveMaze(x, y+1)`

`solveMaze(x-1, y)`

# 图的深度遍历求解迷宫问题



`solveMaze(x, y):`

if (x,y) 是出口: 成功

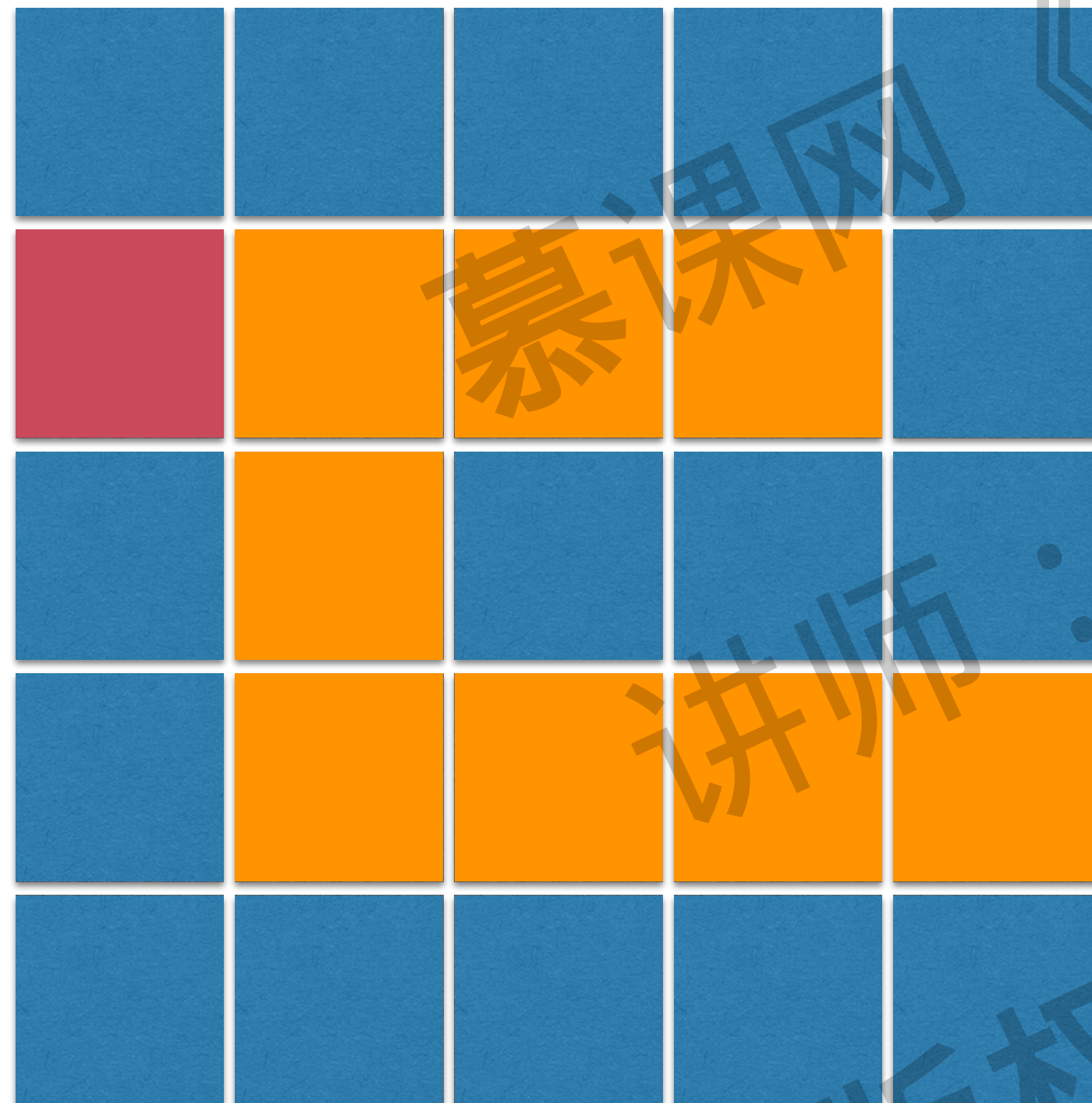
`solveMaze(x, y-1)`

`solveMaze(x+1, y)`

`solveMaze(x, y+1)`

`solveMaze(x-1, y)`

# 图的深度遍历求解迷宫问题



`solveMaze(x, y):`

if (x,y) 是出口: 成功

`solveMaze(x, y-1)`

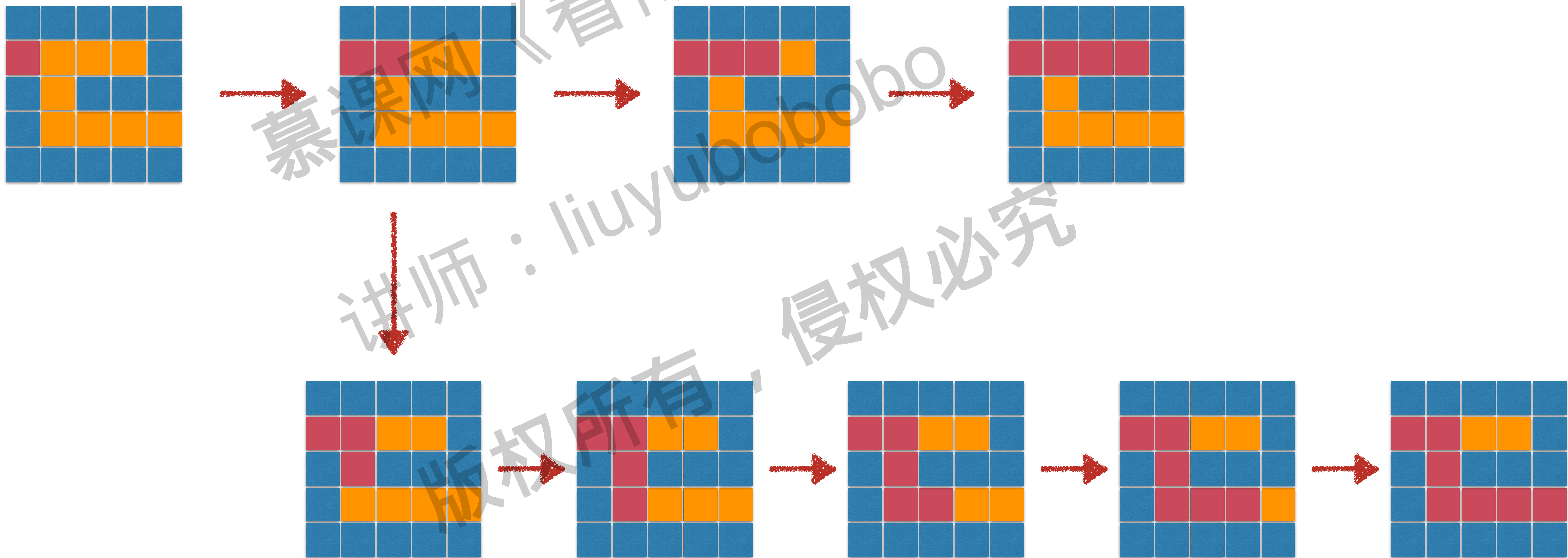
`solveMaze(x+1, y)`

`solveMaze(x, y+1)`

`solveMaze(x-1, y)`



# 图的深度遍历求解迷宫问题



# 图的深度遍历求解迷宫问题

演示：图的深度遍历求解迷宫问题

更多关于使用深度优先遍历实现迷宫求解

# 更多关于使用深度优先遍历实现迷宫求解

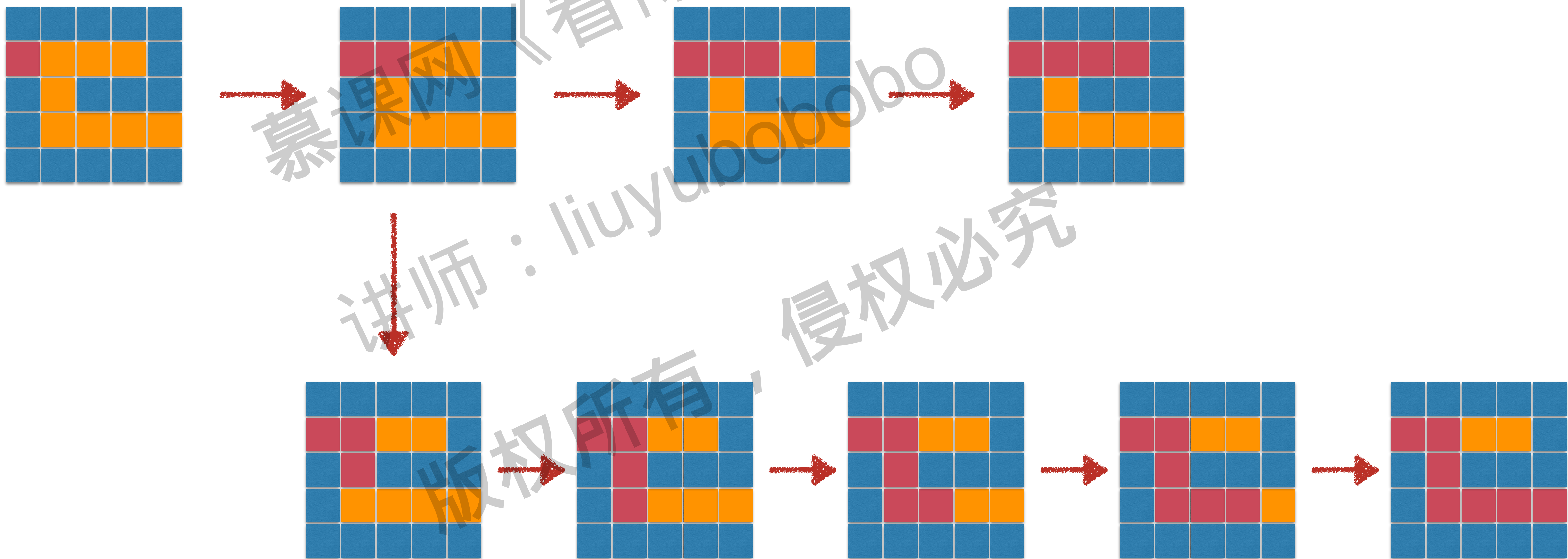
演示：展示回溯过程

# 更多关于使用深度优先遍历实现迷宫求解

演示：让递归函数有返回值

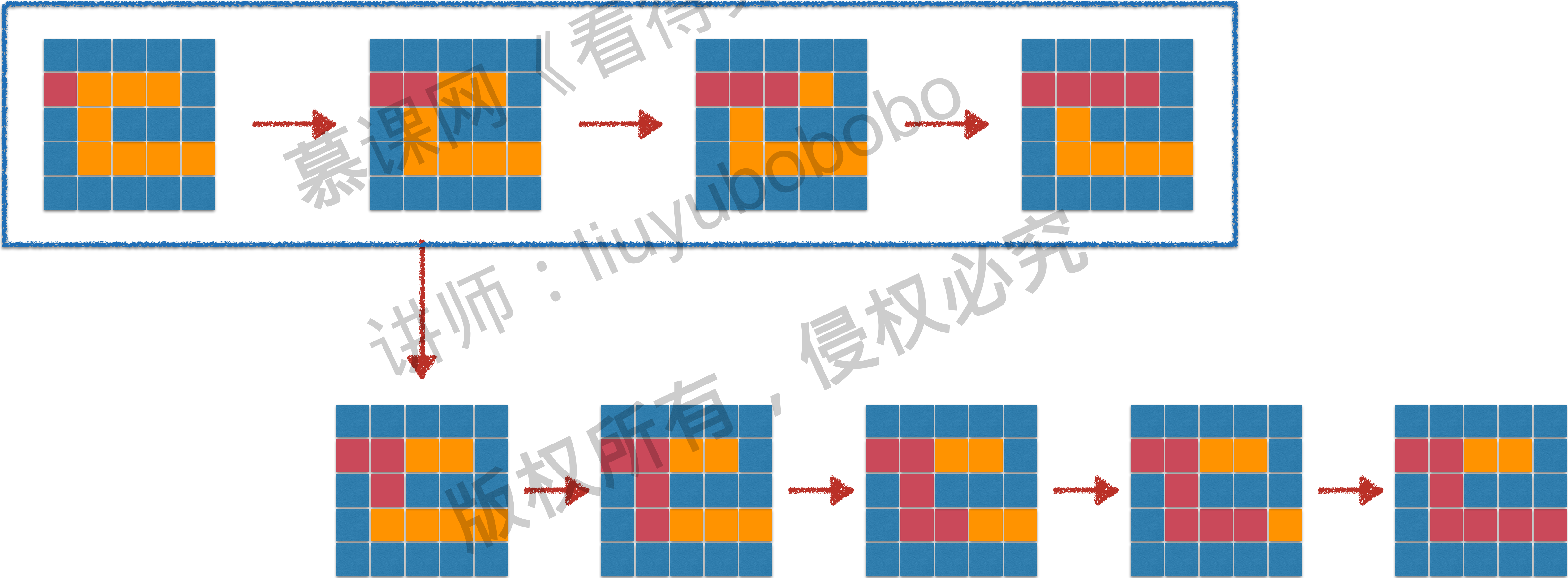
# 深度优先遍历求解迷宫的非递归算法

# 深度优先遍历非递归求解迷宫问题



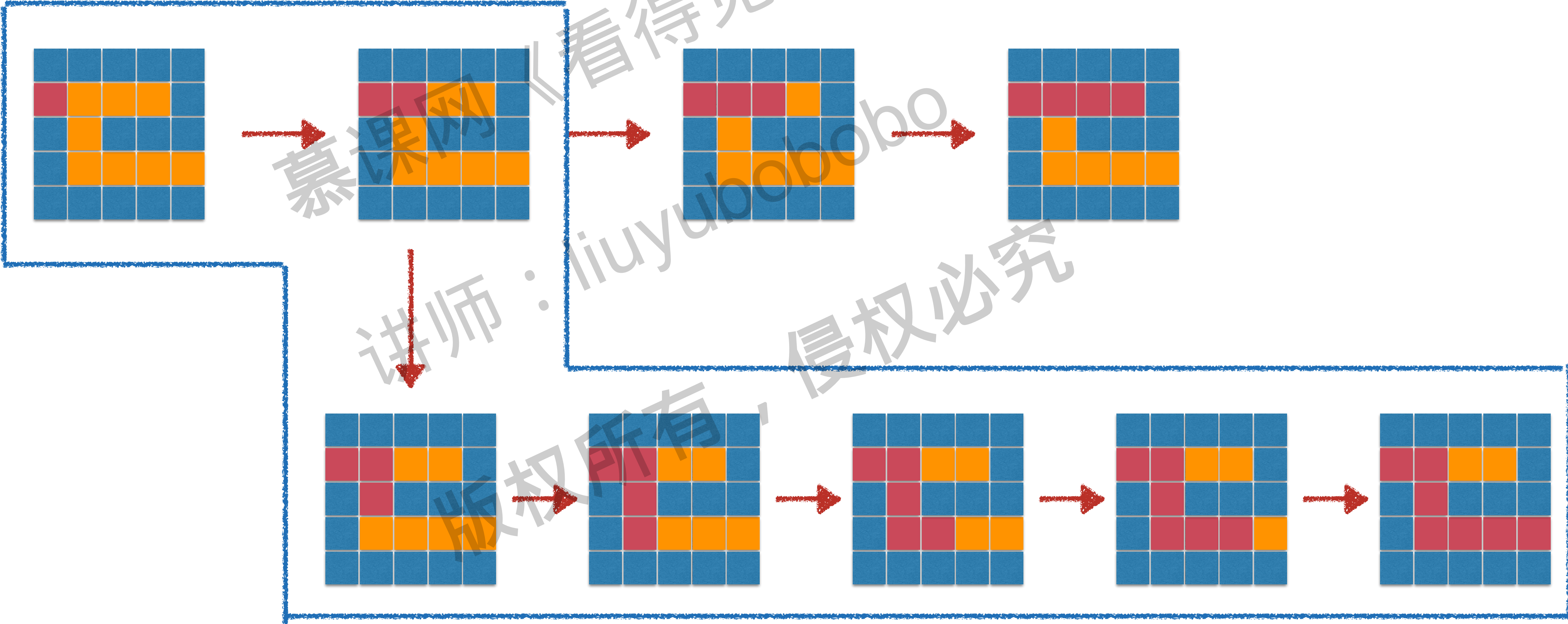


# 深度优先遍历非递归求解迷宫问题

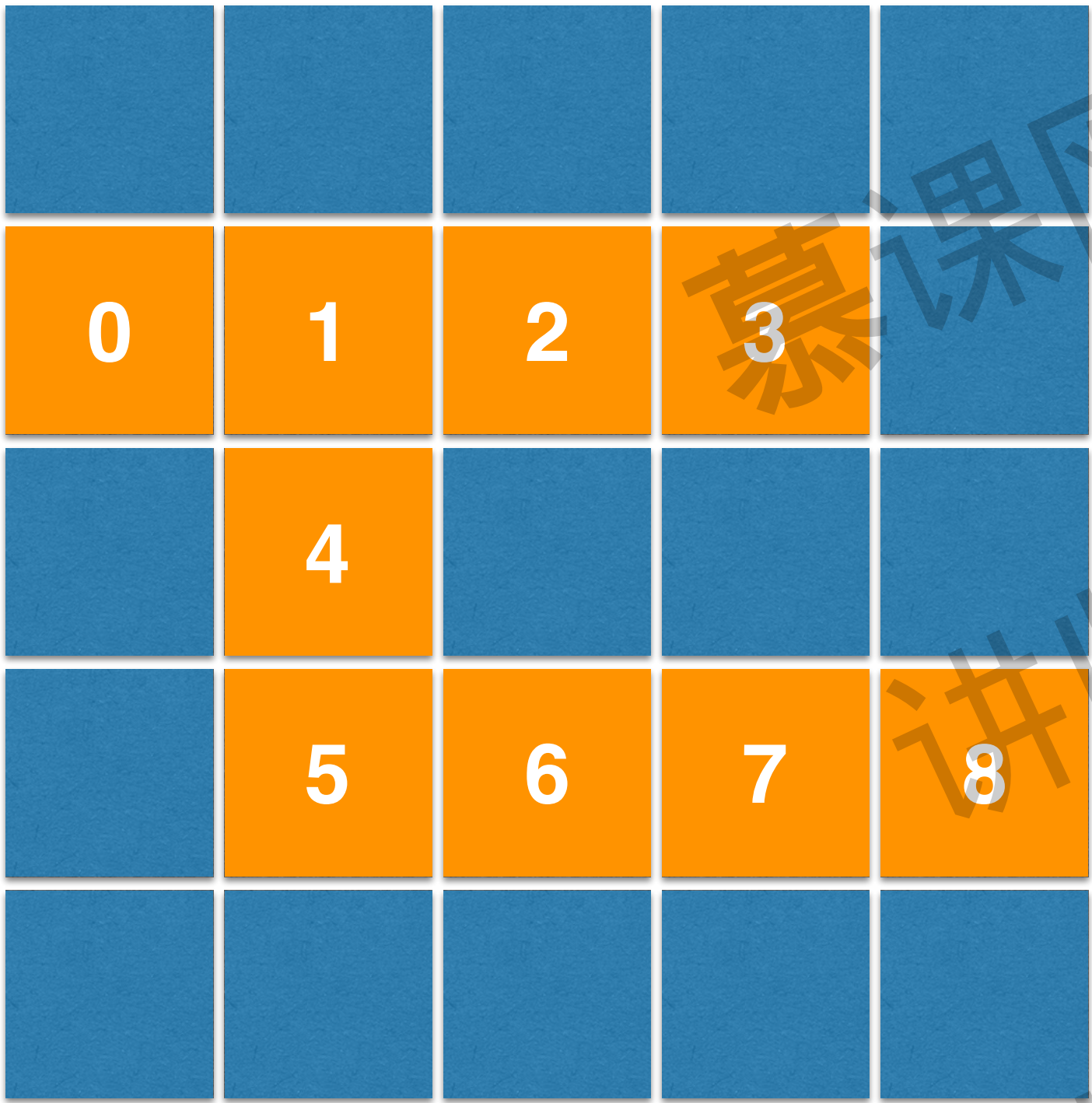




# 深度优先遍历非递归求解迷宫问题

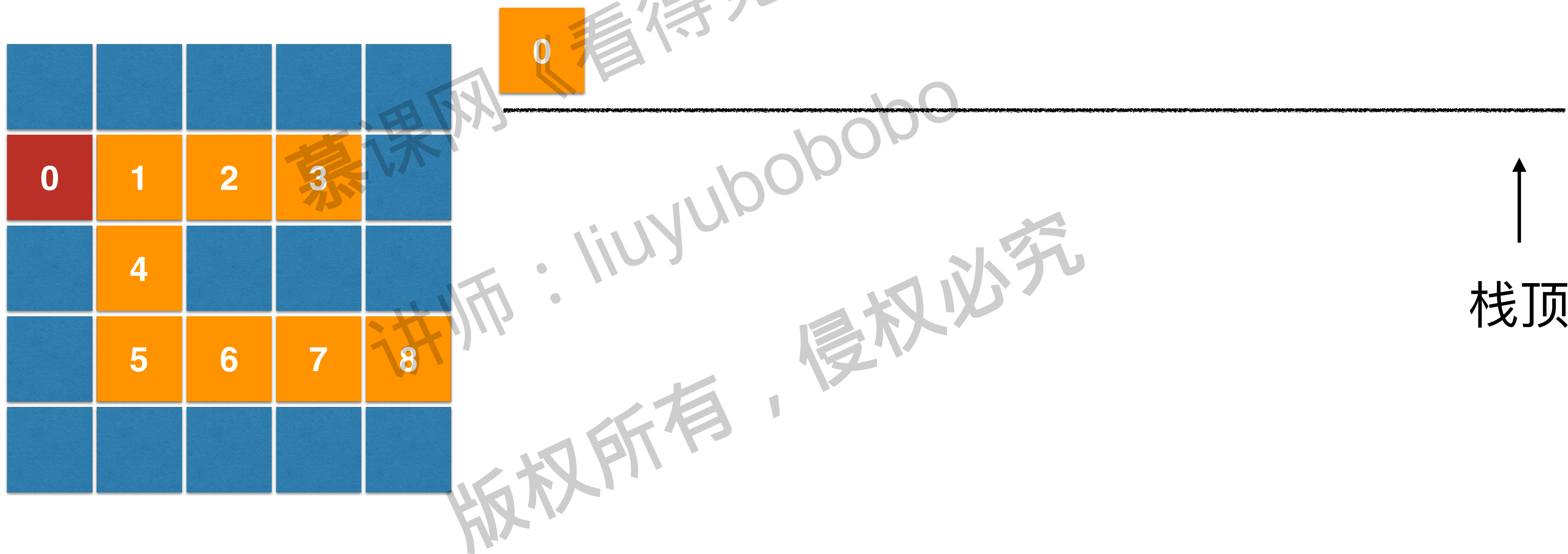


# 深度优先遍历非递归求解迷宫问题



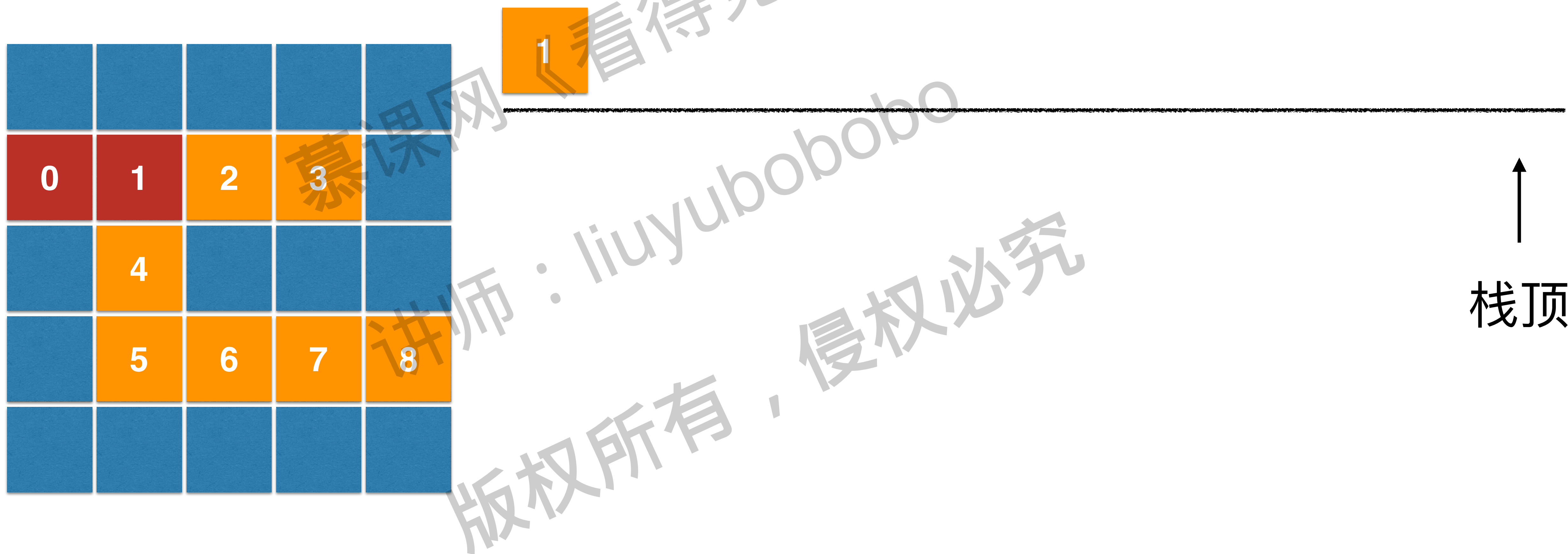
↑  
栈顶

# 深度优先遍历非递归求解迷宫问题

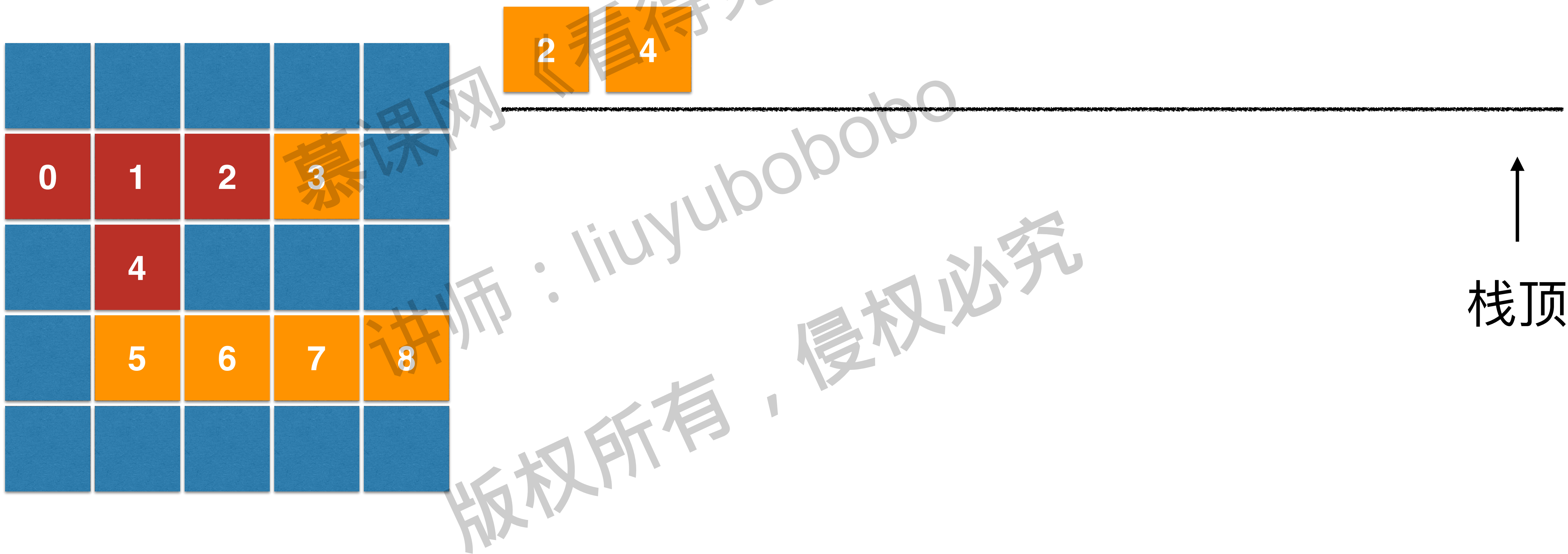




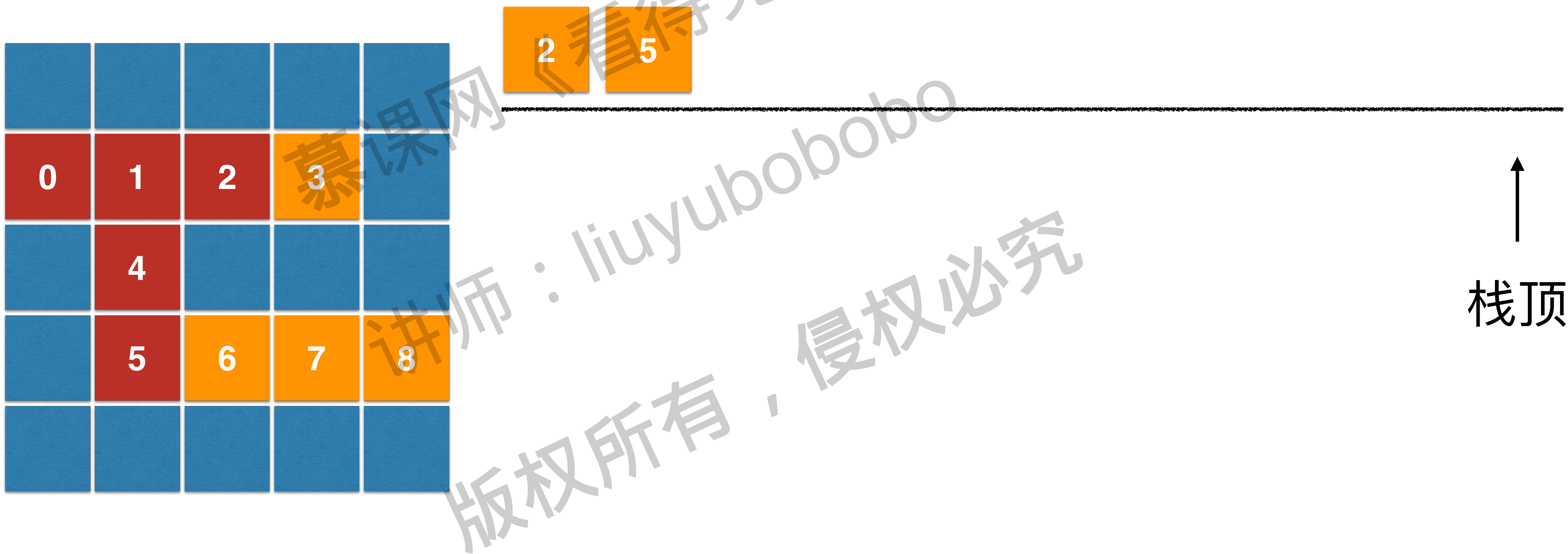
# 深度优先遍历非递归求解迷宫问题



# 深度优先遍历非递归求解迷宫问题

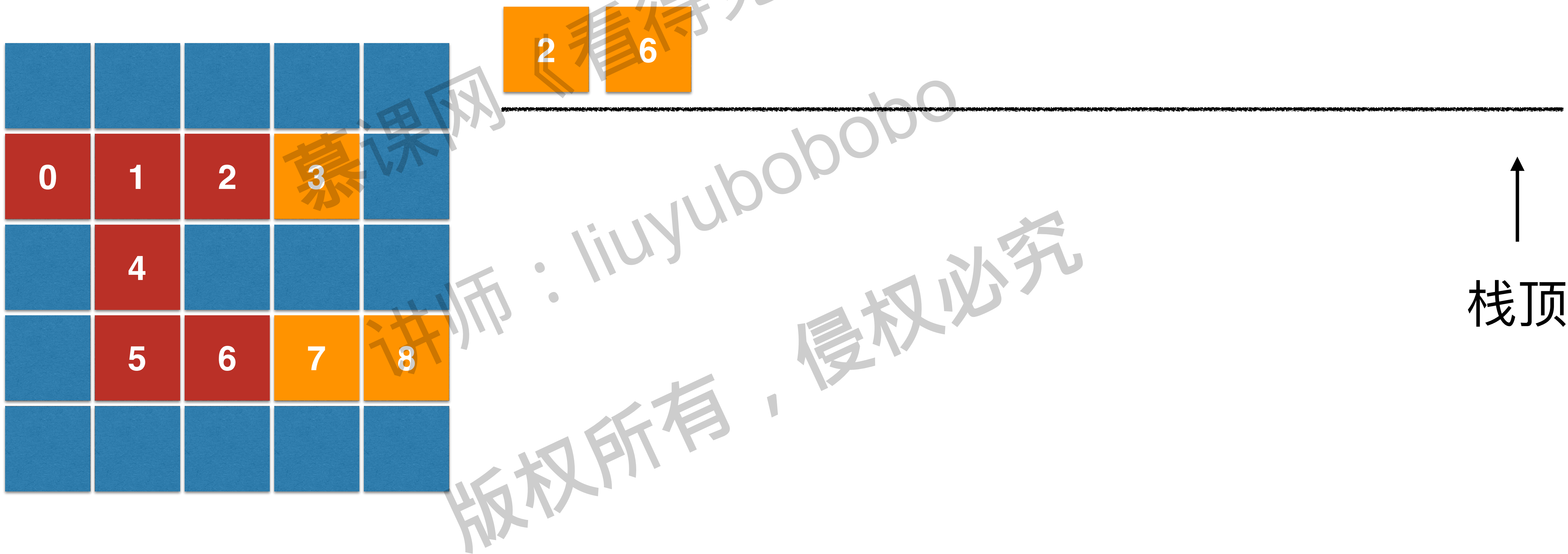


# 深度优先遍历非递归求解迷宫问题

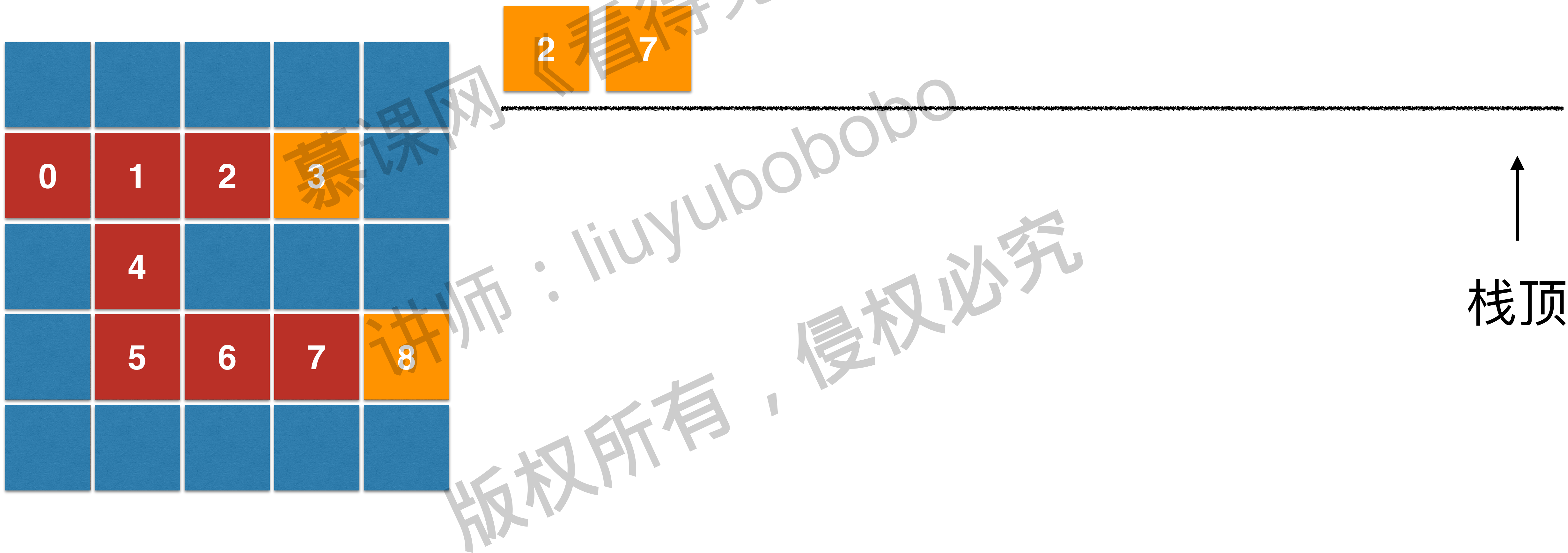




# 深度优先遍历非递归求解迷宫问题

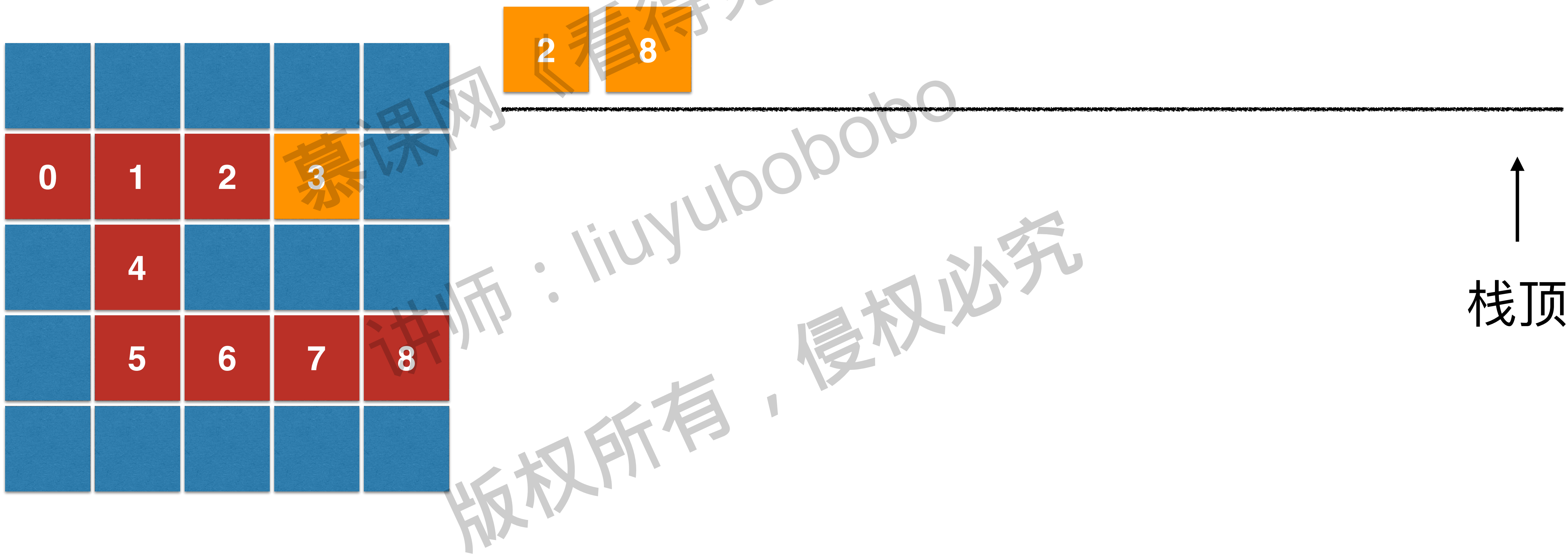


# 深度优先遍历非递归求解迷宫问题





# 深度优先遍历非递归求解迷宫问题



# 深度优先遍历非递归求解迷宫问题

演示：深度优先遍历非递归求解迷宫问题

# 更多关于深度优先遍历的非递归算法 求解迷宫问题

慕课网《看得见的算法》  
讲师：www.yubodoo.com  
版权所有，侵权必究

# 更多关于深度优先遍历的非递归算法 求解迷宫问题

演示：对迷宫无解的情况的处理

# 更多关于深度优先遍历的非递归算法 求解迷宫问题

演示：寻找最终的路径

# 广度优先遍历求解迷宫问题

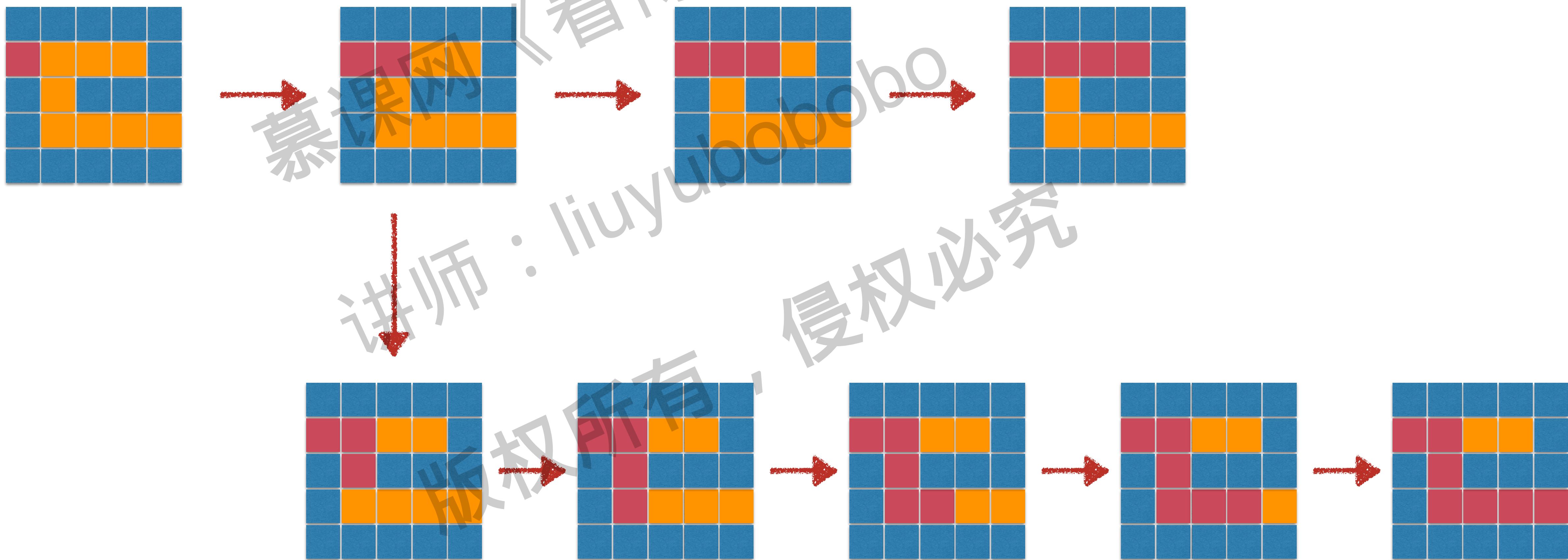
慕课网《看得见的算法》

讲师：lilyubobobo

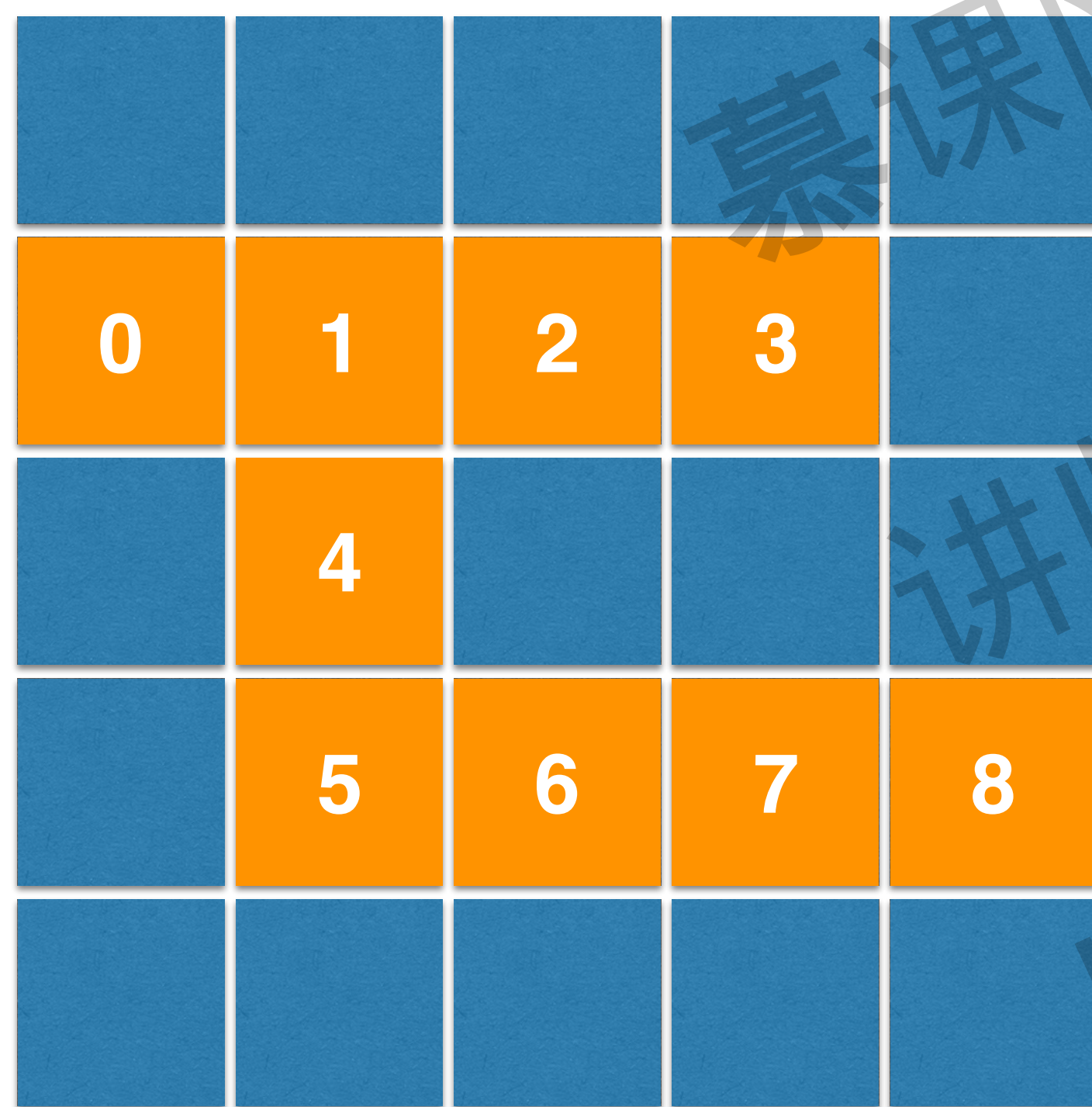
版权所有，侵权必究



# 广度优先遍历求解迷宫问题



# 广度优先遍历求解迷宫问题

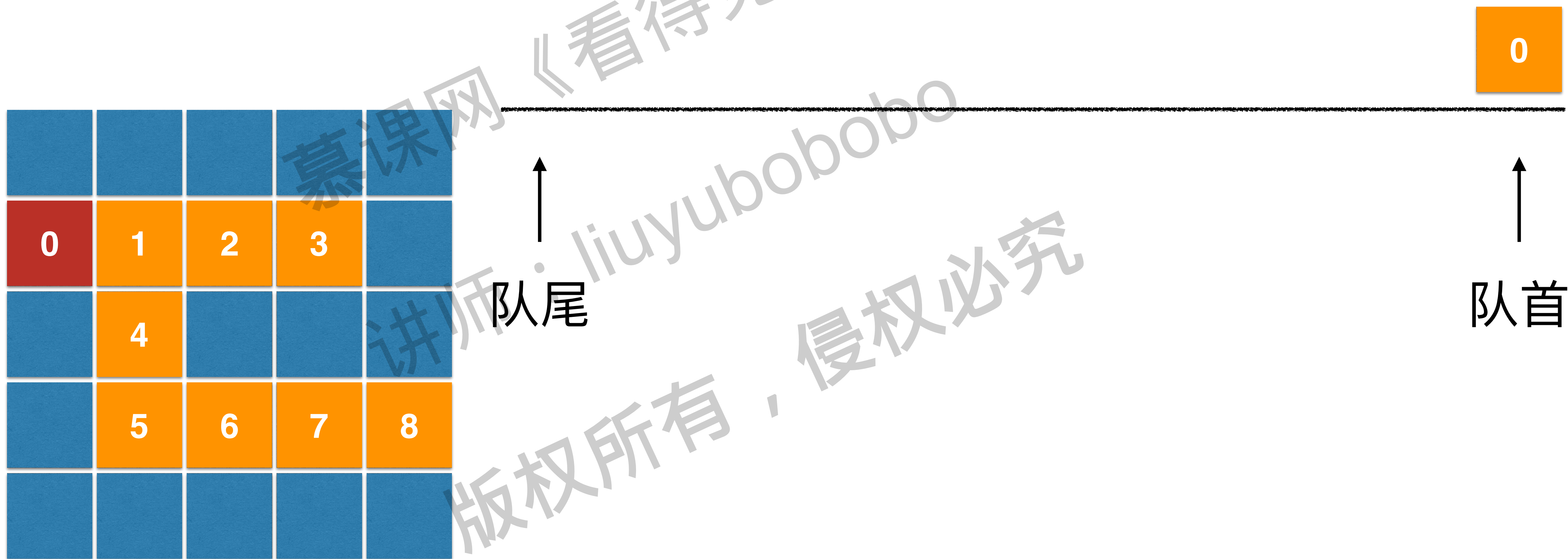


↑  
队尾

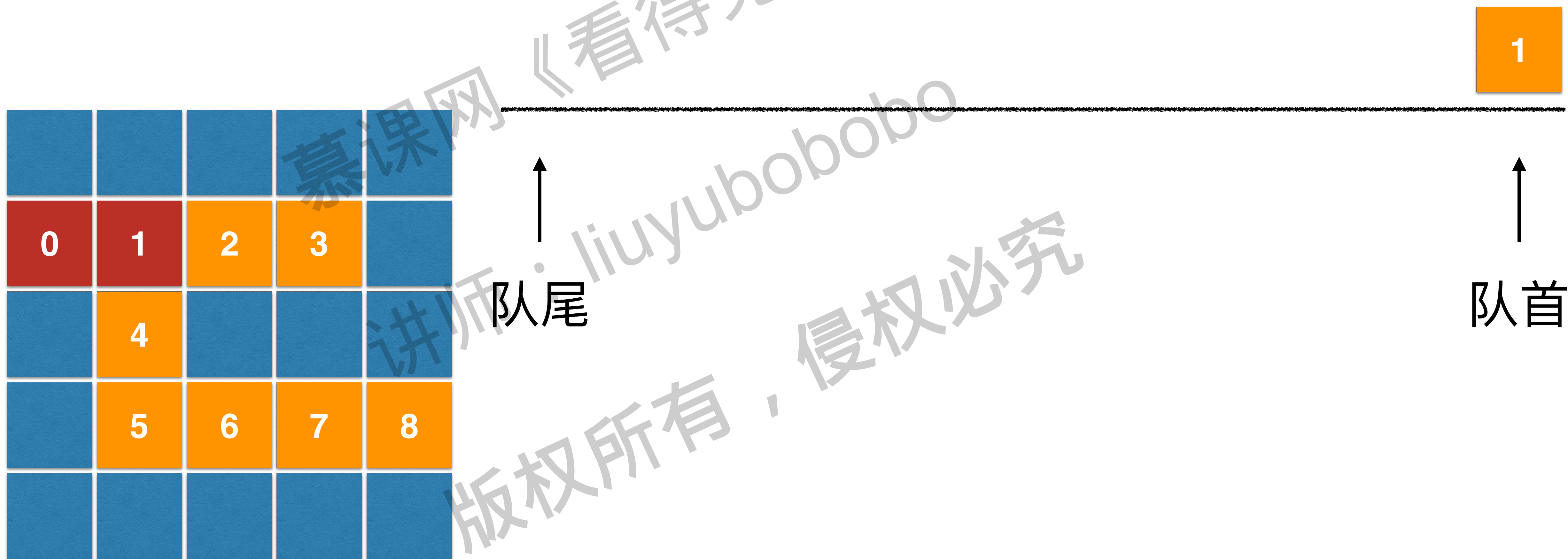
↑  
队首



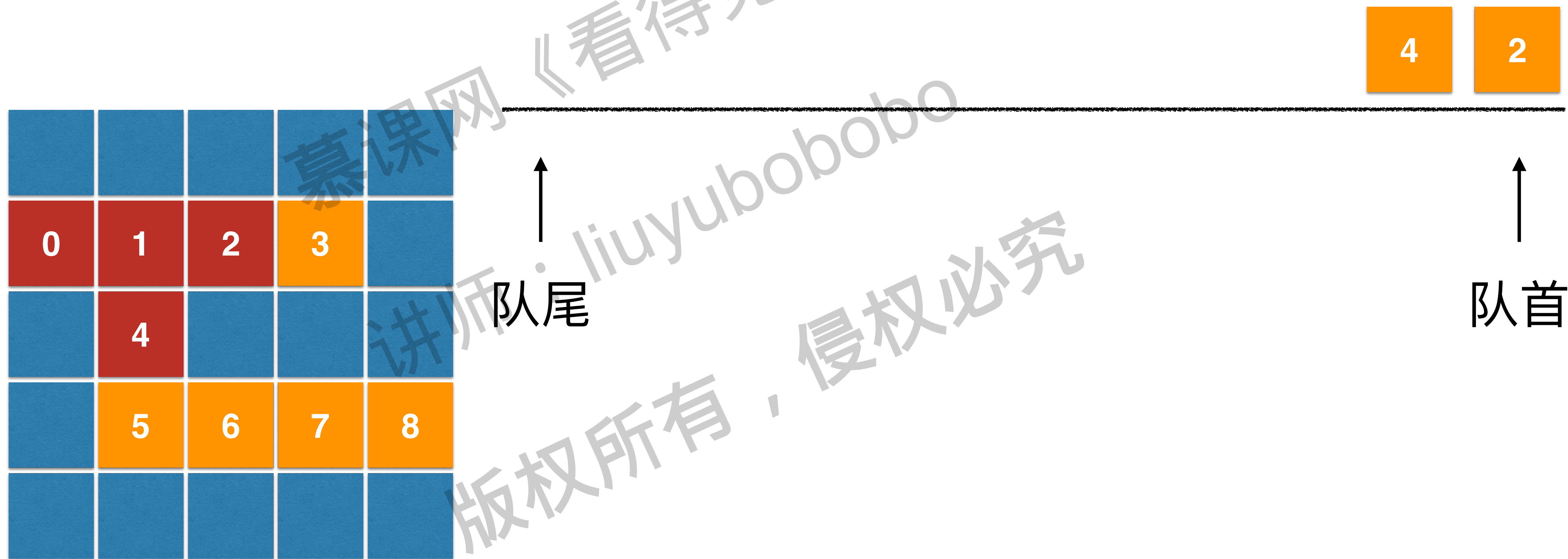
# 广度优先遍历求解迷宫问题



# 广度优先遍历求解迷宫问题

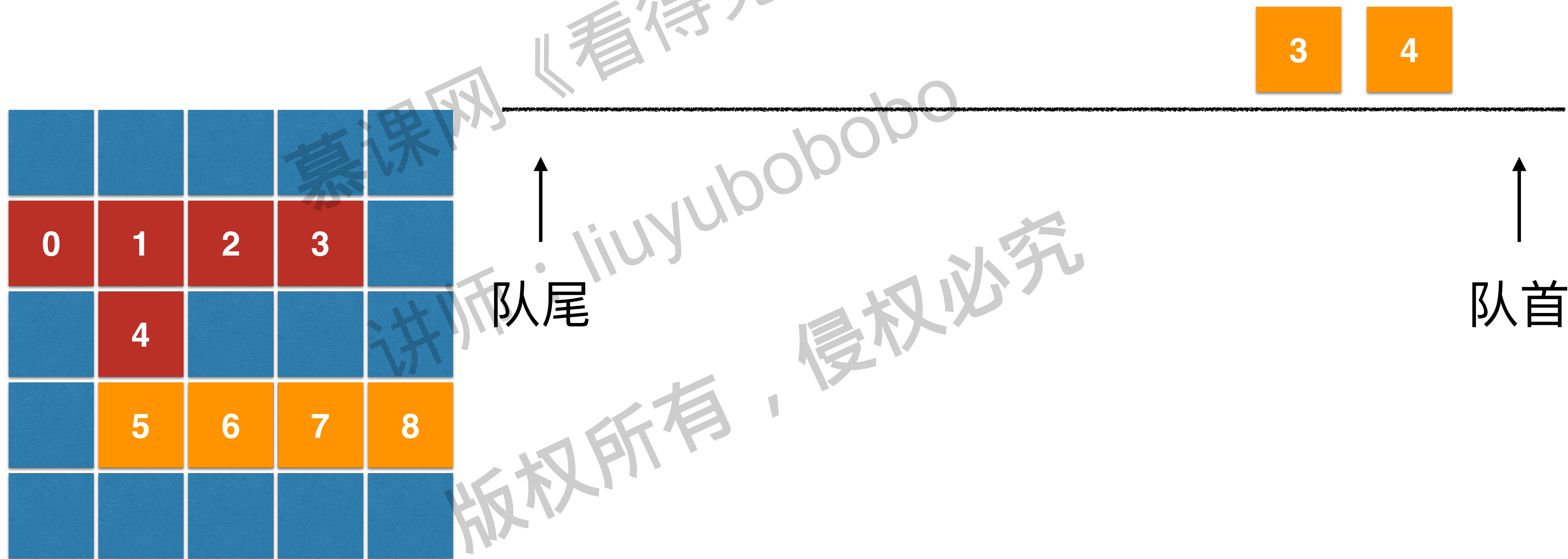


# 广度优先遍历求解迷宫问题

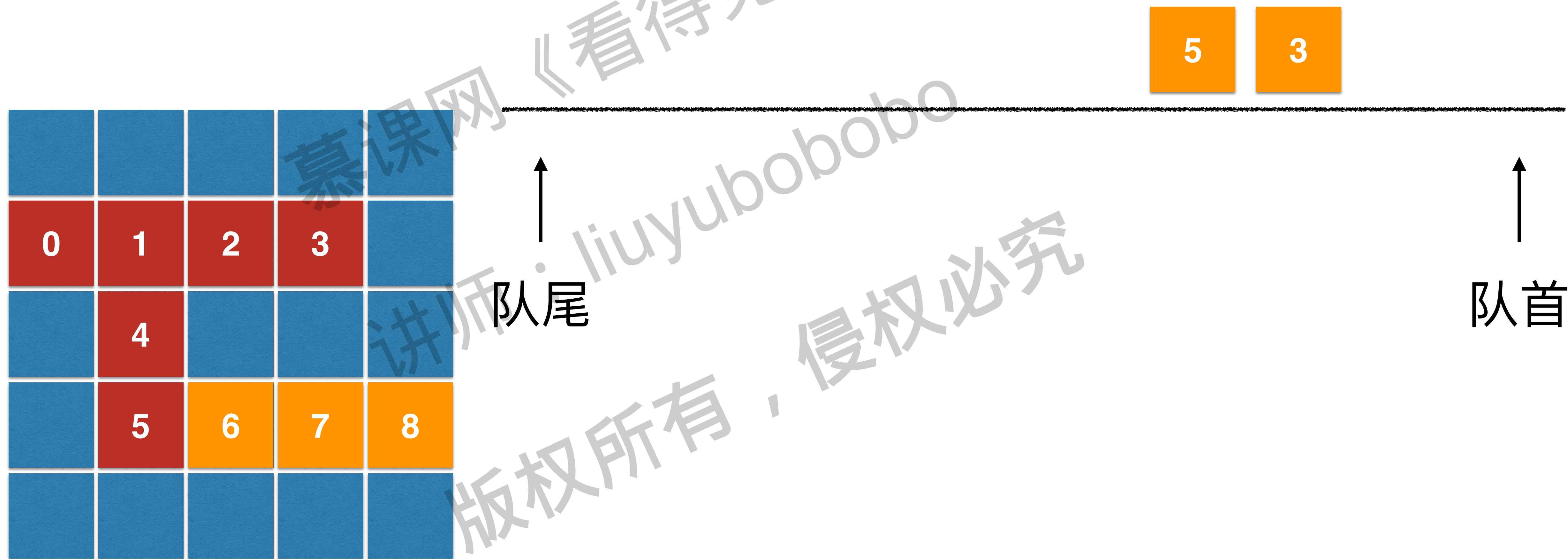




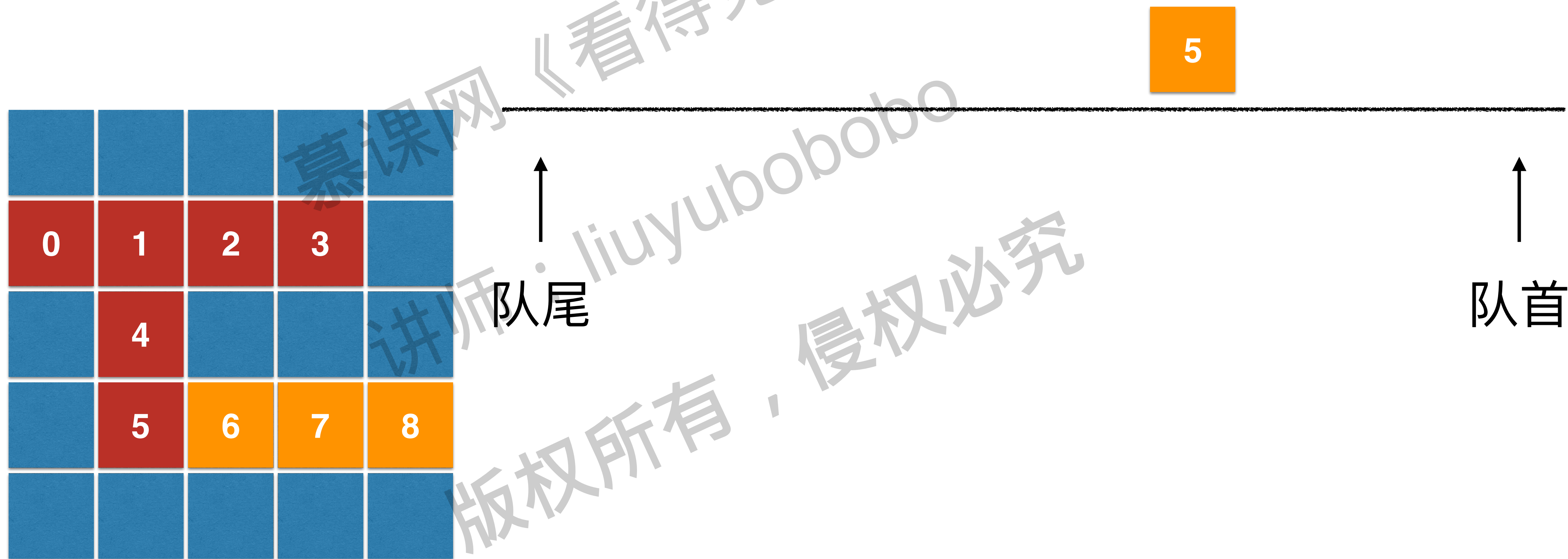
# 广度优先遍历求解迷宫问题



# 广度优先遍历求解迷宫问题

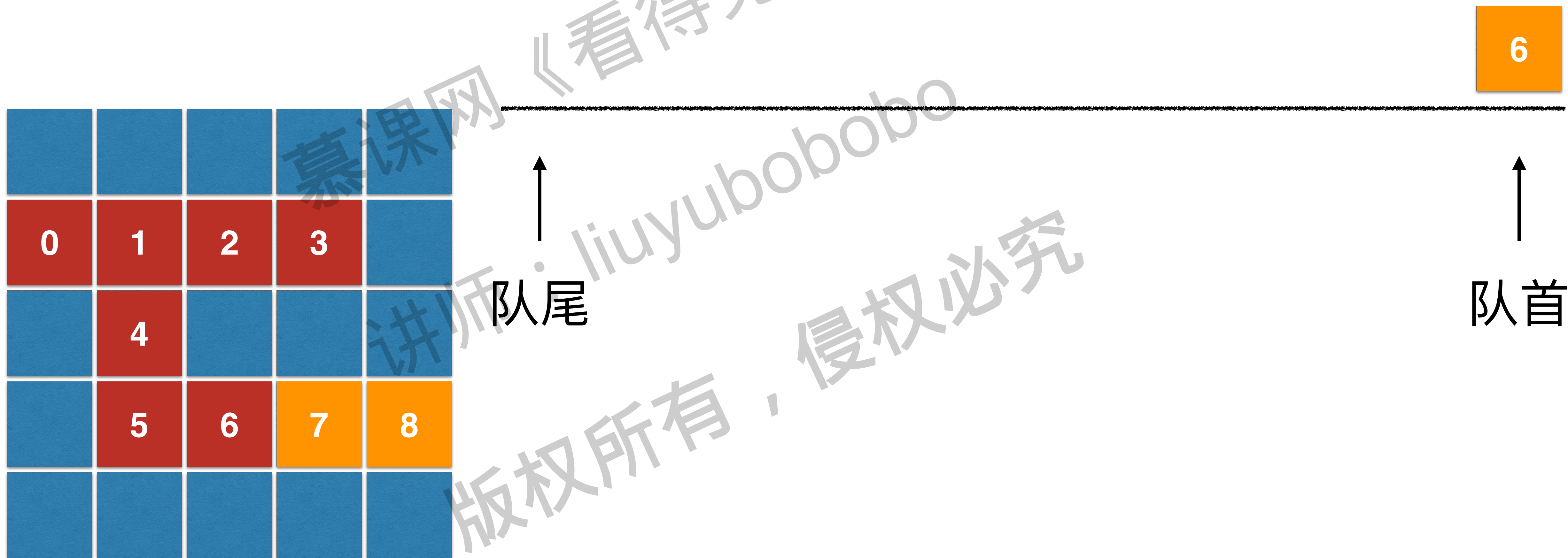


# 广度优先遍历求解迷宫问题

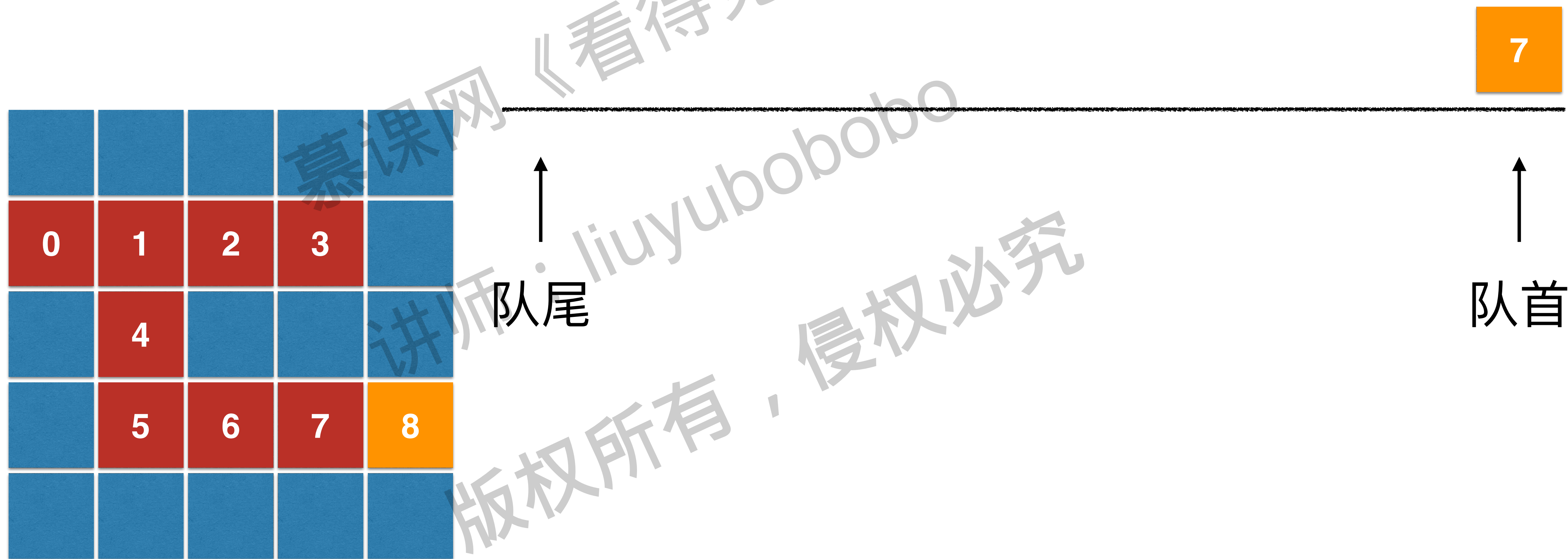




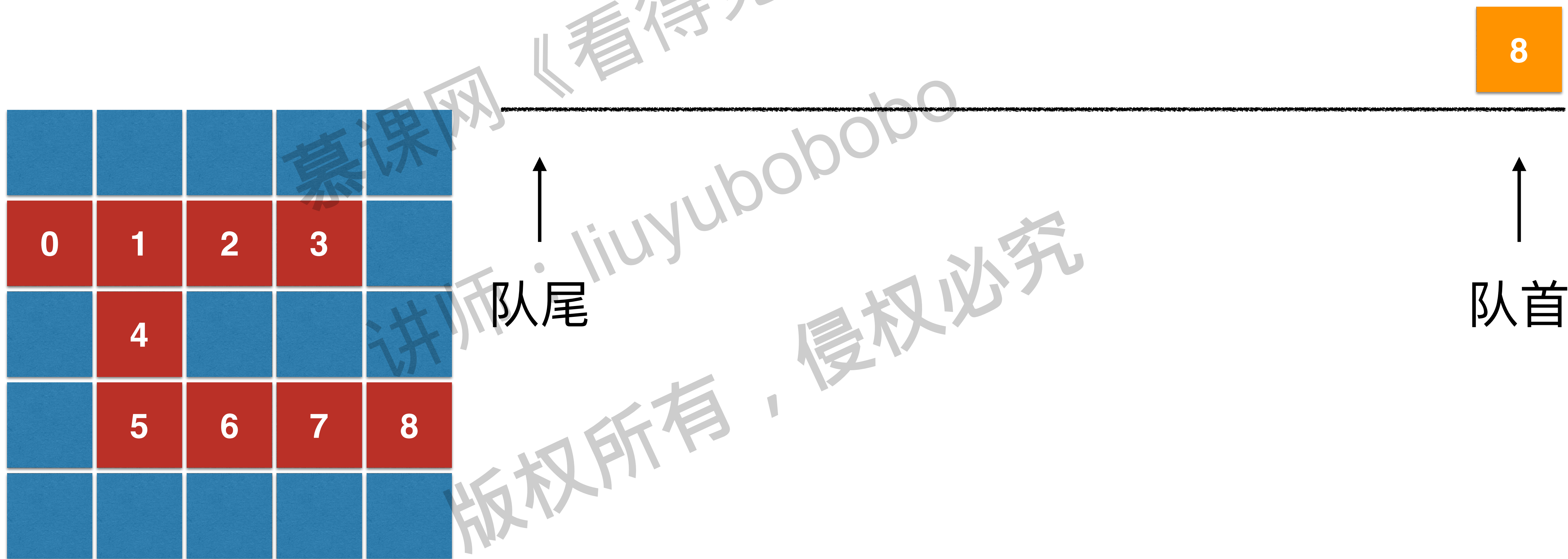
# 广度优先遍历求解迷宫问题



# 广度优先遍历求解迷宫问题



# 广度优先遍历求解迷宫问题





# 广度优先遍历求解迷宫问题

演示：广度优先遍历求解迷宫问题

慕课网《看得见的算法》

## 本章小结

讲师：liuyubobobo

版权所有，侵权必究



慕课网《看得见的算法》

# 递归算法

讲师：lilyubobobo

版权所有，侵权必究

# 广度优先遍历求解迷宫问题

如果迷宫有多个解

广度优先遍历求得的是最短解

# 深度优先遍历和广度优先遍历的关系

# 深度优先遍历和广度优先遍历的关系

```
stack.push(入口)
```

```
while( !stack.empty() )
```

```
    curPos = stack.pop()
```

```
    if( curPos == 出口 ) break
```

```
    对和curPos相邻的每一个可能的方向
```

```
        if( newPos可达 )
```

```
            stack.push(newPos)
```

```
queue.enqueue(入口)
```

```
while( queue.size() != 0 )
```

```
    curPos = queue.dequeue()
```

```
    if( curPos == 出口 ) break
```

```
    对和curPos相邻的每一个可能的方向
```

```
        if( newPos可达 )
```

```
            queue.enqueue(newPos)
```

# 深度优先遍历和广度优先遍历的关系

stack.push(入口)

while( !stack.empty() )

curPos = stack.pop()

if( curPos == 出口 ) break

对和curPos相邻的每一个可能的方向

if( newPos可达 )

stack.push(newPos)

queue.enqueue(入口)

while( queue.size() != 0 )

curPos = queue.dequeue()

if( curPos == 出口 ) break

对和curPos相邻的每一个可能的方向

if( newPos可达 )

queue.enqueue(newPos)



# 深度优先遍历和广度优先遍历的关系

```
stack.add(入口)
```

```
while( !stack.empty() )
```

```
    curPos = stack.remove()
```

```
    if( curPos == 出口 ) break
```

对和curPos相邻的每一个可能的方向

```
    if( newPos可达 )
```

```
        stack.add(newPos)
```

```
queue.add(入口)
```

```
while( !queue.empty() )
```

```
    curPos = queue.remove()
```

```
    if( curPos == 出口 ) break
```

对和curPos相邻的每一个可能的方向

```
    if( newPos可达 )
```

```
        queue.add(newPos)
```

# 深度优先遍历和广度优先遍历的关系

```
q.add(入口)
```

```
while( !q.empty() )
```

```
    curPos = q.remove()
```

```
    if( curPos == 出口 ) break
```

对和curPos相邻的每一个可能的方向

```
    if( newPos可达 )
```

```
        q.add(newPos)
```

# 深度优先遍历和广度优先遍历的关系

抽象队列：

可入队，可出队

LIFO(stack)；FIFO(queue)；优先队列(堆) 都可以成为是一个抽象队列

入队，出队的规则不同

# 其他

欢迎大家关注我的个人公众号：是不是很酷



慕课网《看得见的算法》

# 看得见的算法

讲师：liuyubobobo

版权所有，侵权必究

liuyubobobo