

The background features a light gray hexagonal grid pattern. On the left side, there is a large, stylized circular graphic composed of concentric rings in shades of green and blue, with some segments labeled 'TWO', 'THREE', 'FOUR', and 'FIVE'.

线段树

胡船长

初航我带你，远航靠自己

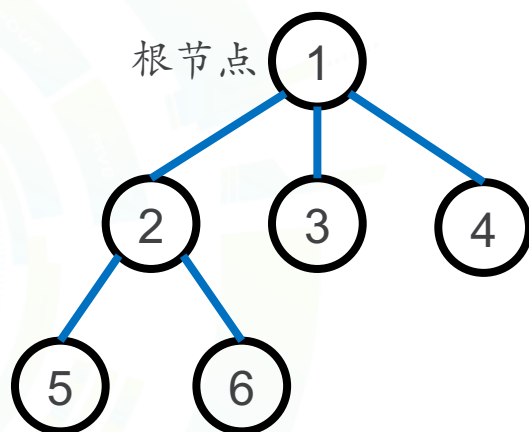
区间修改及查询

	0	1	2	3	4	5	6	7	8	9
	5	4	8	6	4	3	9	7	6	2
	0	1	2	3	4	5	6	7	8	9
Modify (7 , 9)	5	4	8	6	4	3	9	9	6	2
	0	1	2	3	4	5	6	7	8	9
Query (2 , 6)	5	4	8	6	4	3	9	9	6	2
	0	1	2	3	4	5	6	7	8	9
Modify (2 , 3)	5	4	3	6	4	3	9	9	6	2
	0	1	2	3	4	5	6	7	8	9
Query (2 , 6)	5	4	3	6	4	3	9	9	6	2

→ 30

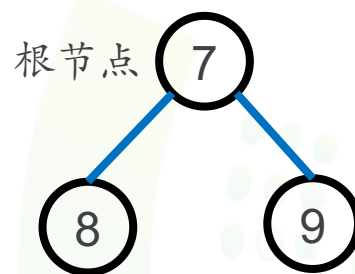
→ 25

树形结构



树的节点：代表集合

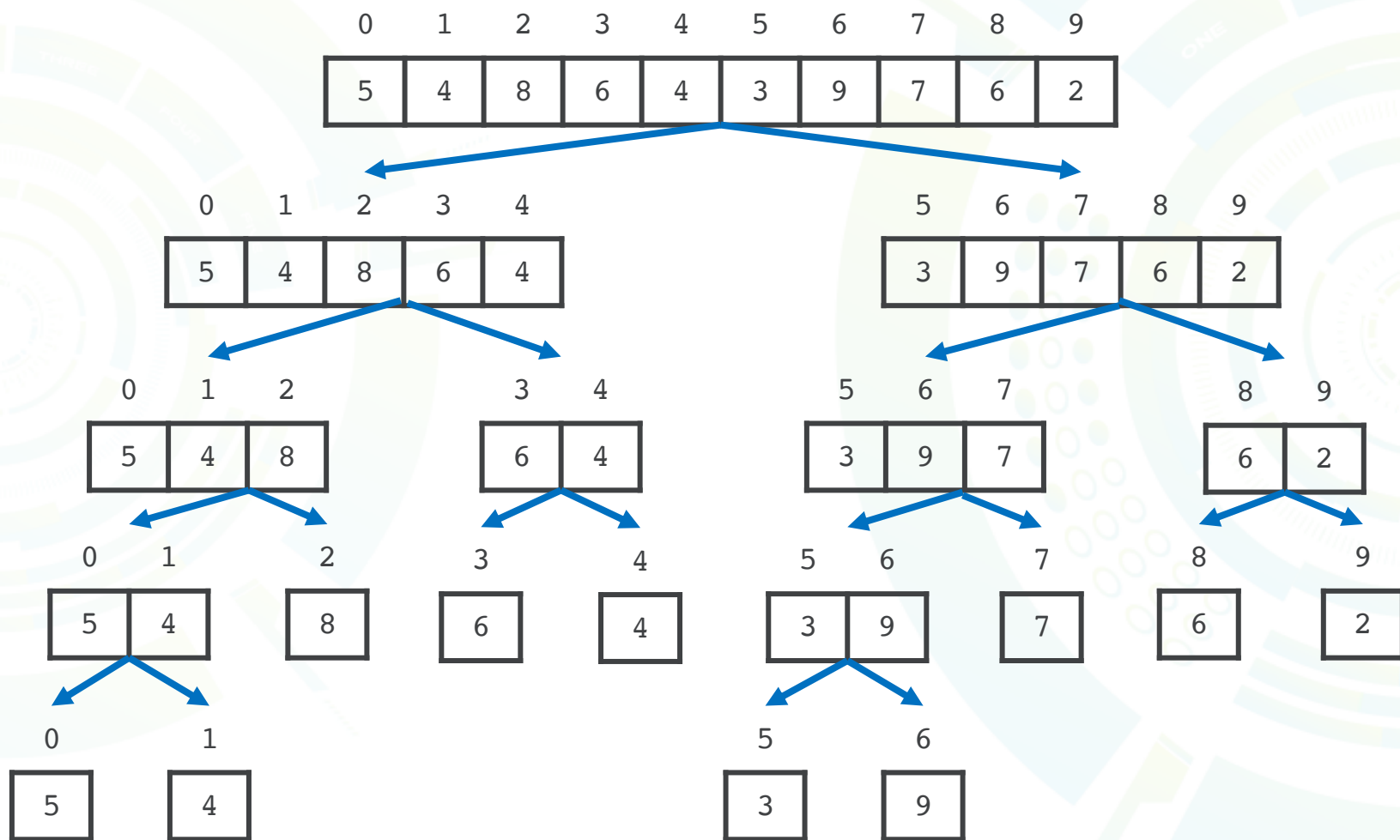
树的边：代表关系



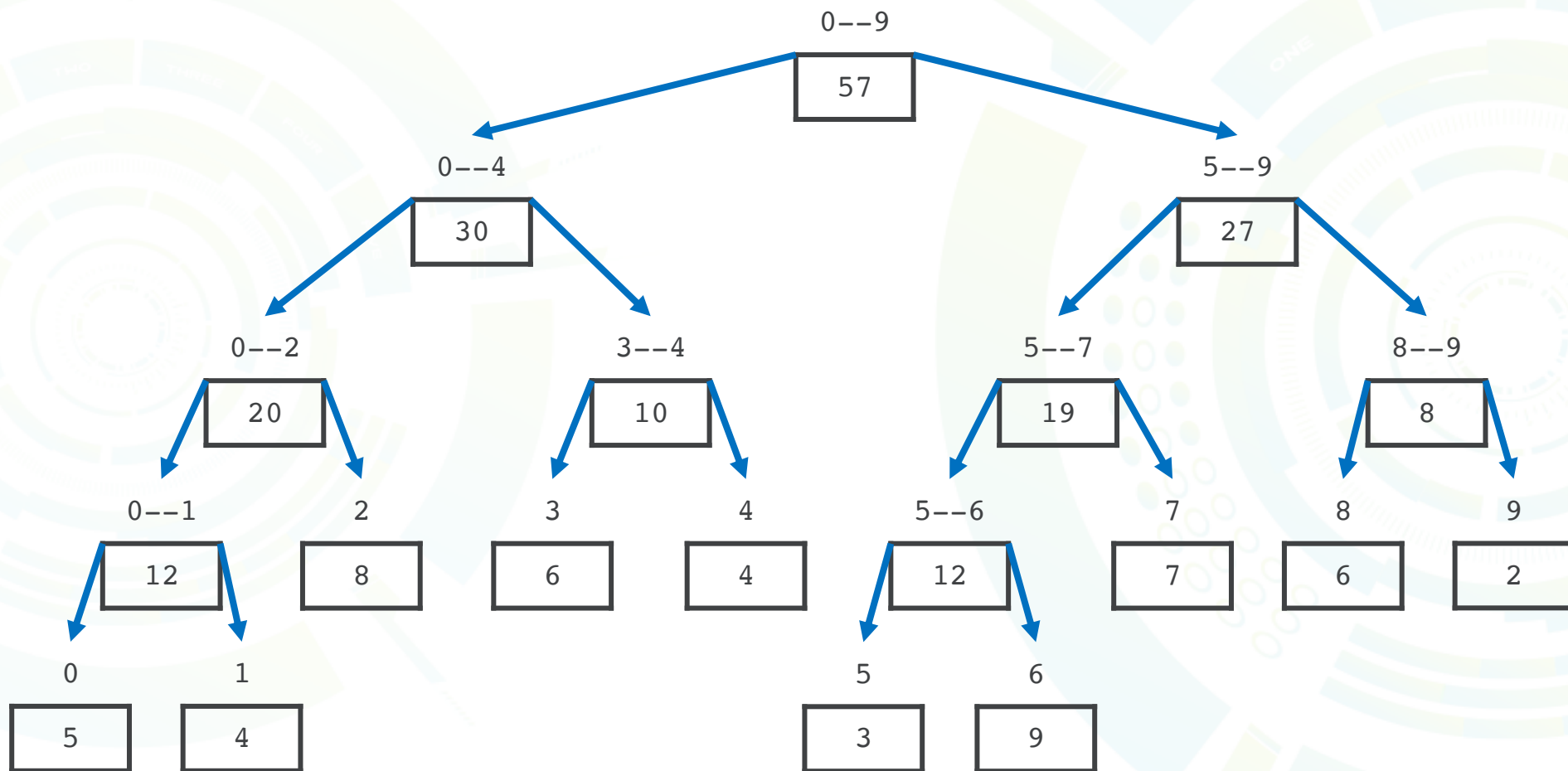
拥有相同父节点的属于一个集合

从父节点根据不同关系分出的子集

线段树

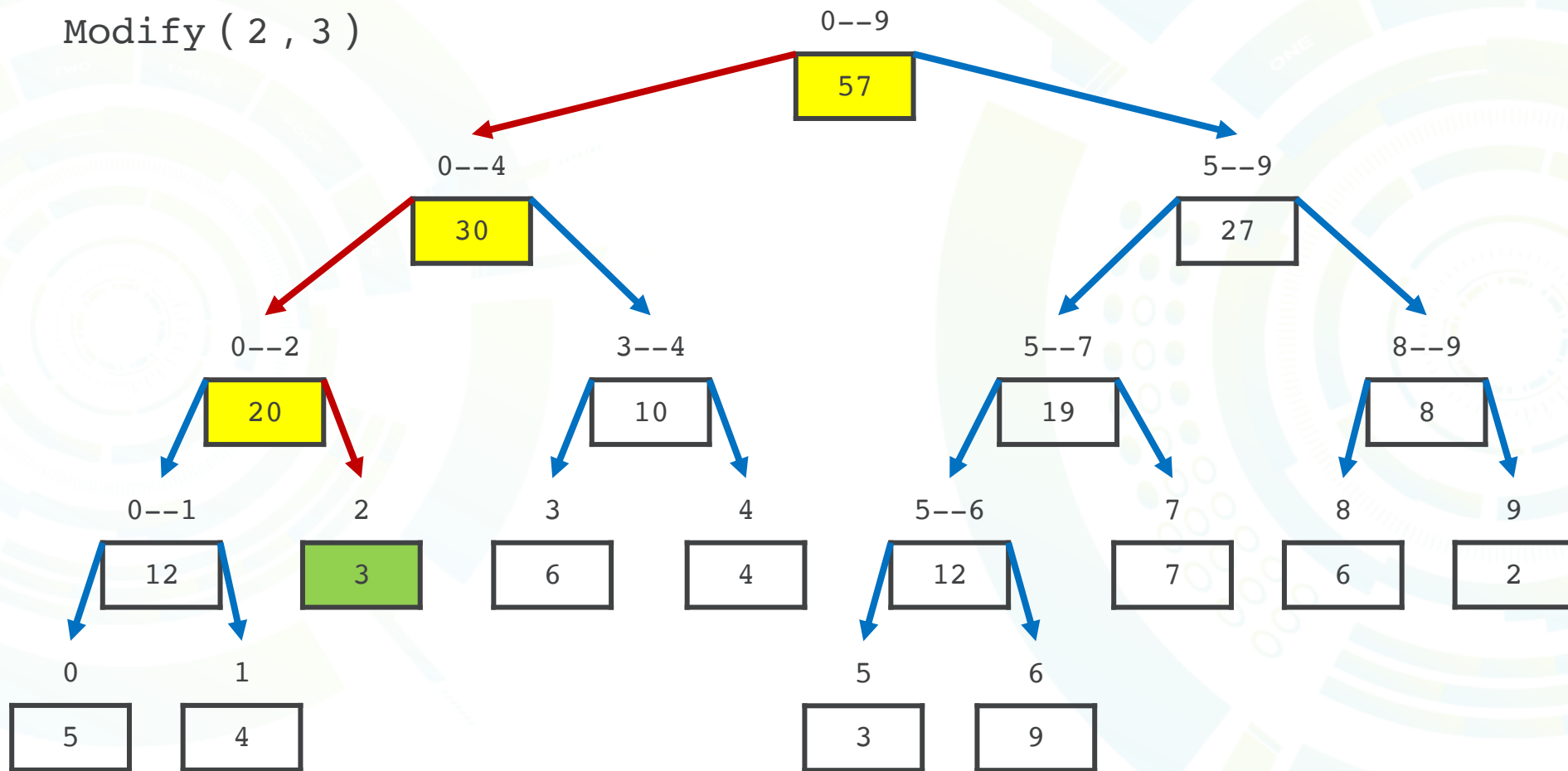


线段树



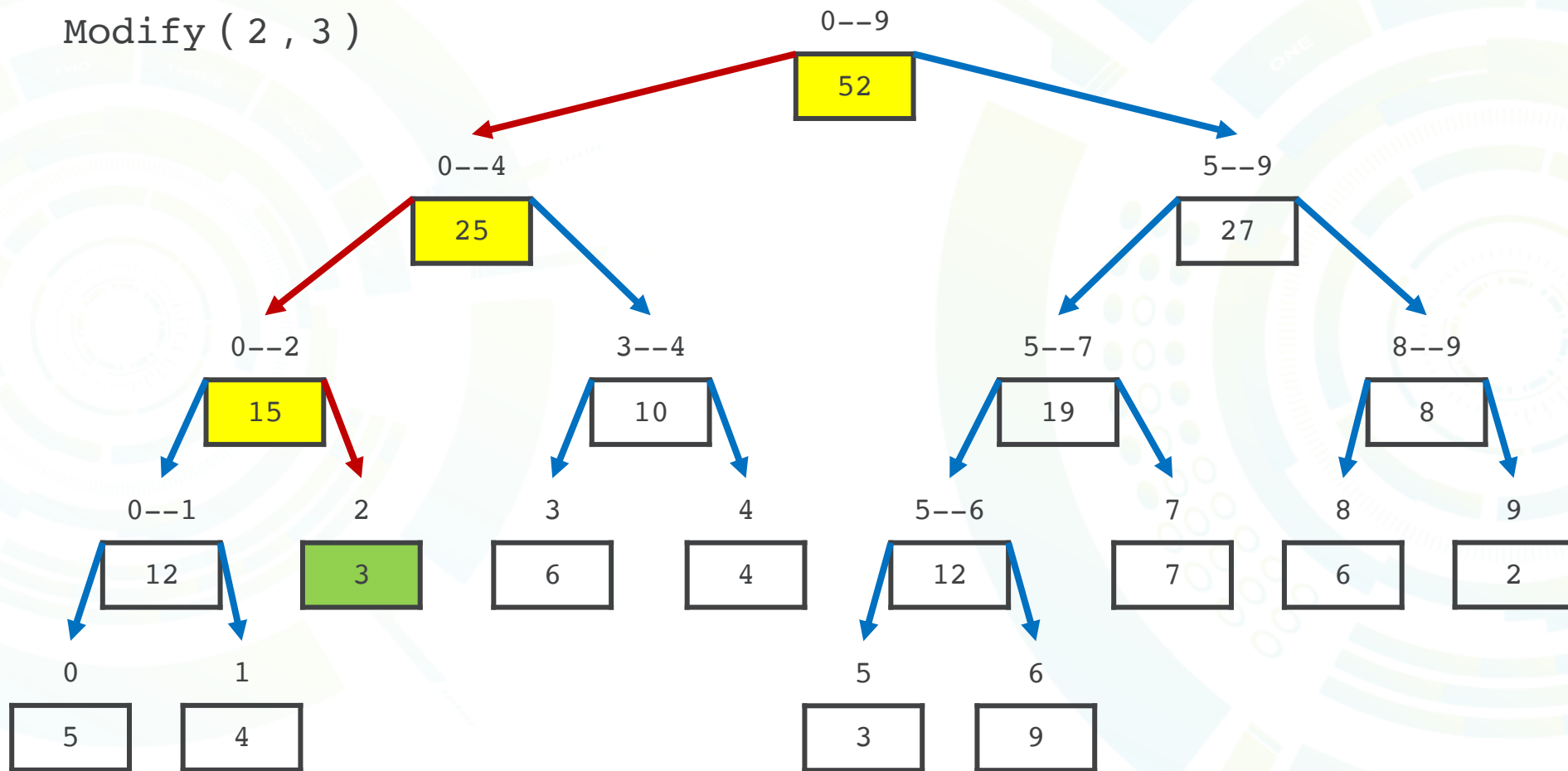
线段树—单点修改

Modify (2 , 3)



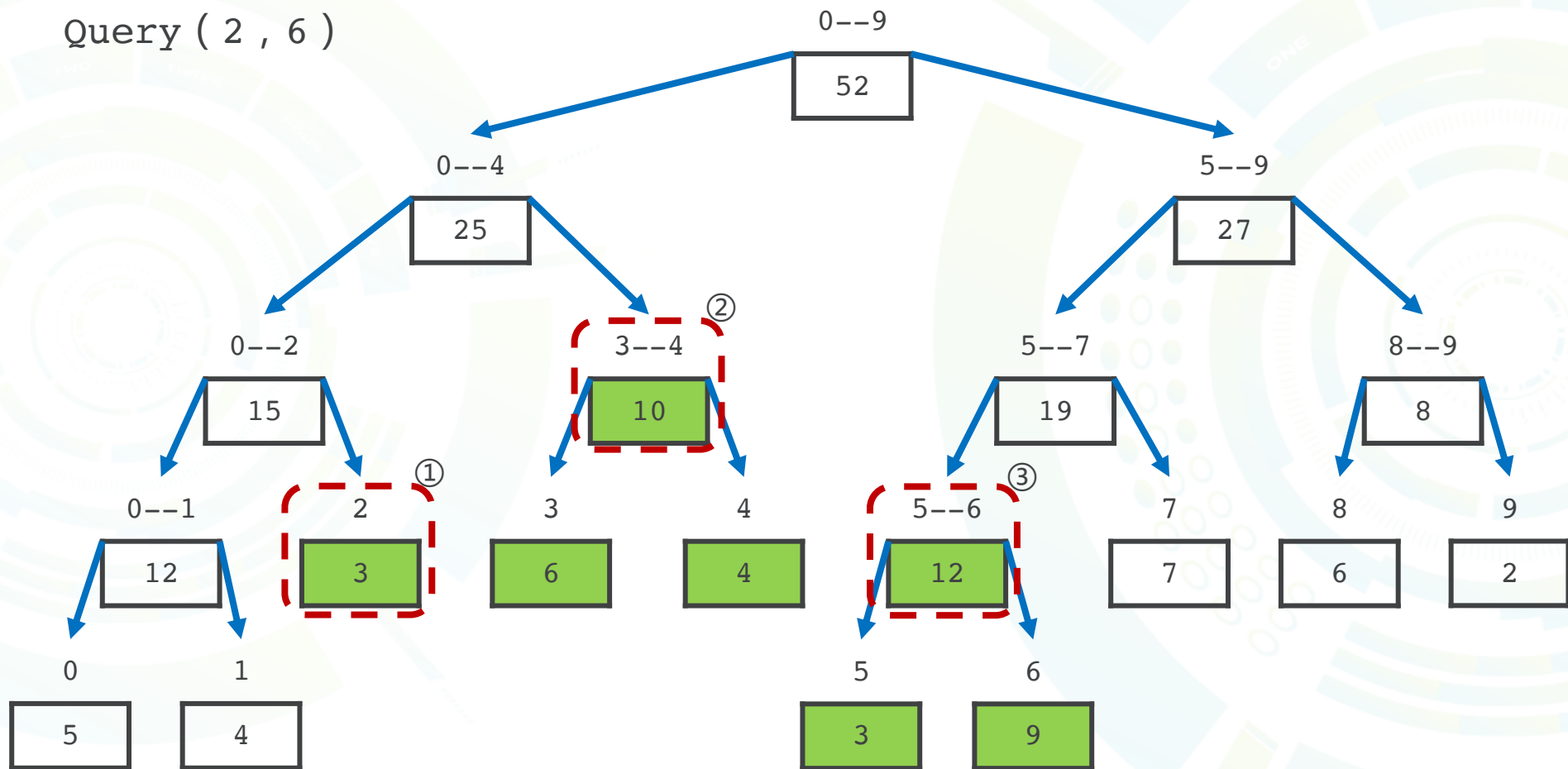
线段树—单点修改

Modify (2 , 3)



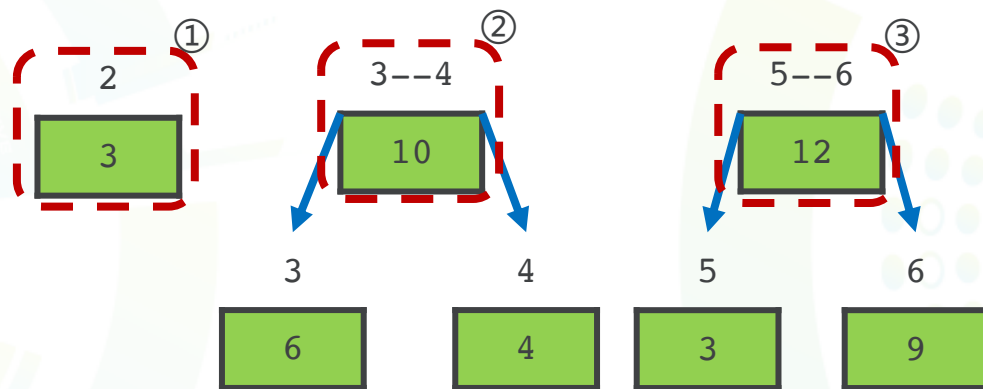
线段树—区间查询

Query (2 , 6)



线段树—区间查询

Query (2 , 6)

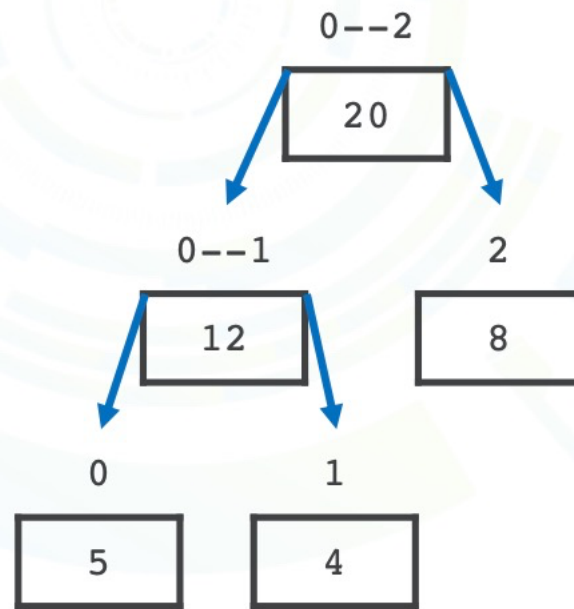


线段树总结

- 1、单点修改: $\log(n)$
- 2、区间查询: $\log(n)$

问题思考:

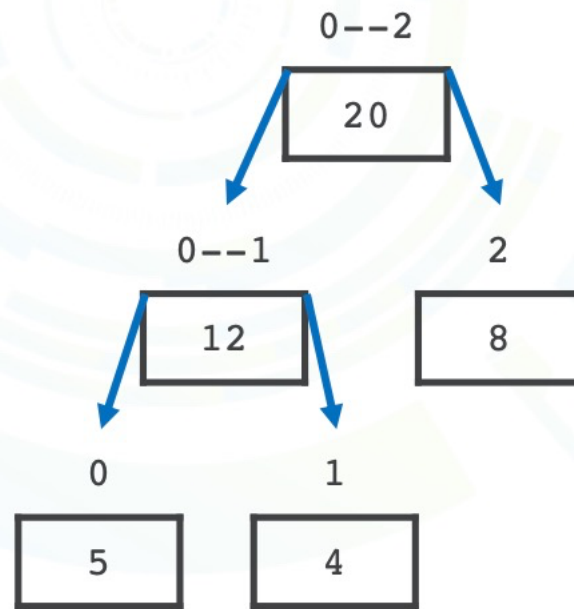
- 1、若采用完全二叉树的存储方式, n 个点的线段树, 最多需要多少个节点空间?
- 2、如何做区间修改?



线段树总结

- 1、叶节点(度为0)有 n 个
- 2、度为 2 的节点有 $n-1$ 个
- 3、最后一层最多需要 $2*n$ 的存储空间

最多需要节点空间: $n + n - 1 + 2n = 4n - 1$

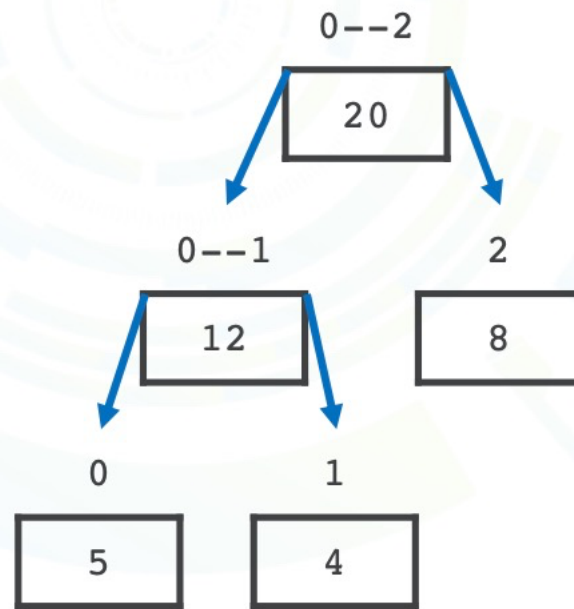


线段树总结

深入思考：

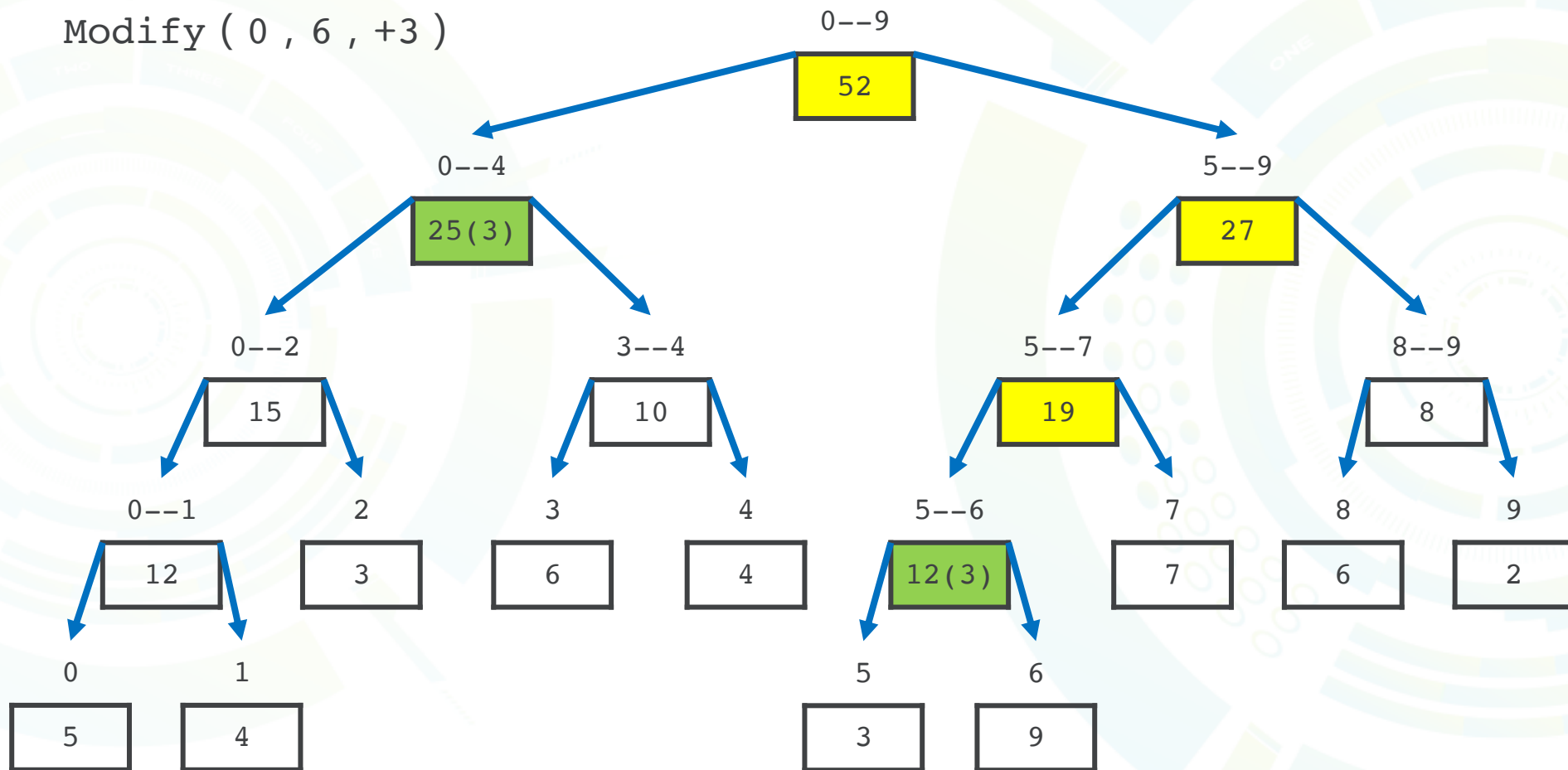
对于区间修改任务，最笨的想法，就是拆解成 m 次单点修改，那么区间修改的时间复杂度： $O(m \log n)$

反观暴力式区间修改时间复杂度： $O(m)$



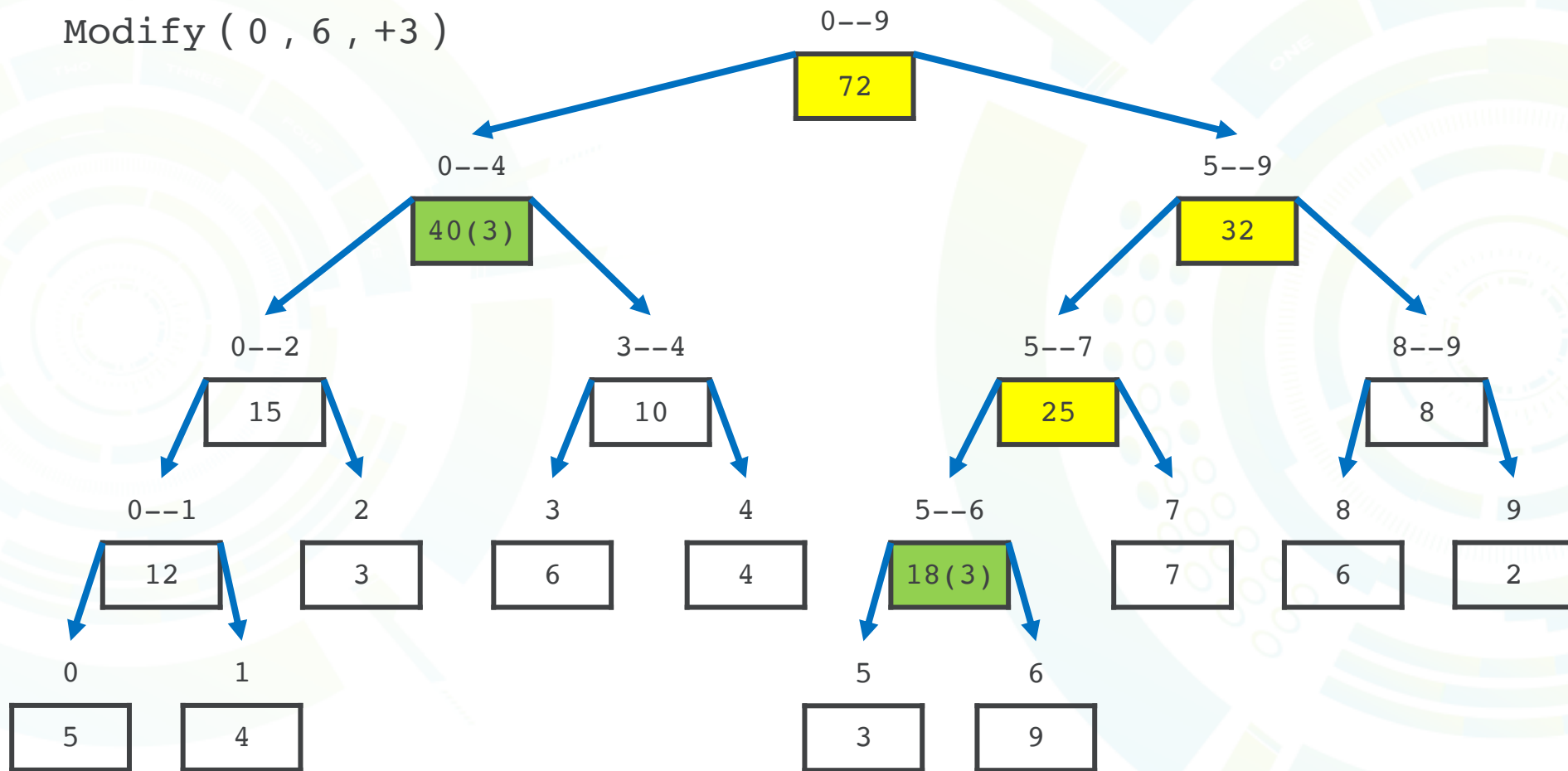
线段树—区间修改

Modify (0 , 6 , +3)



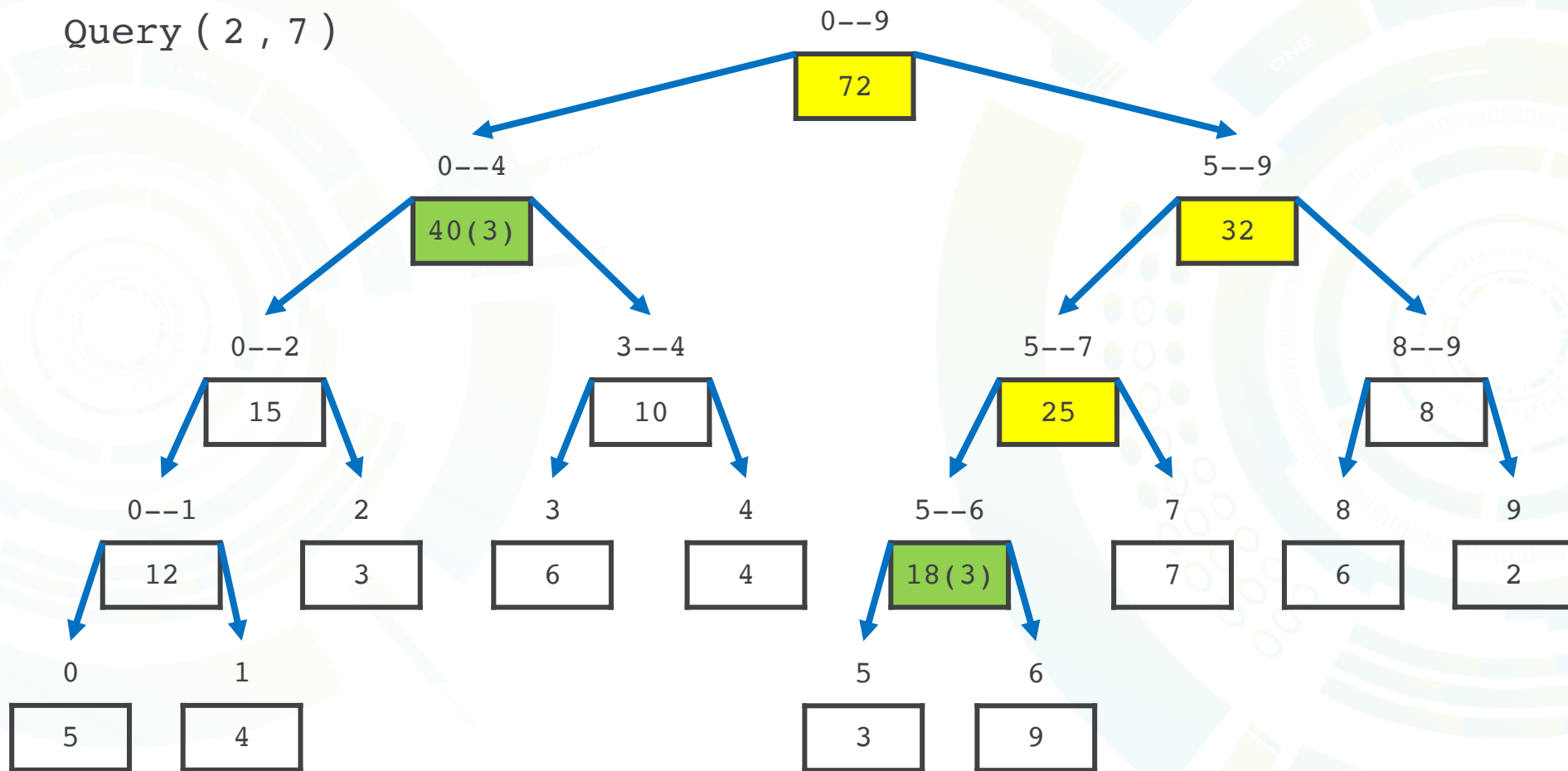
线段树—区间修改

Modify (0 , 6 , +3)



线段树—区间修改

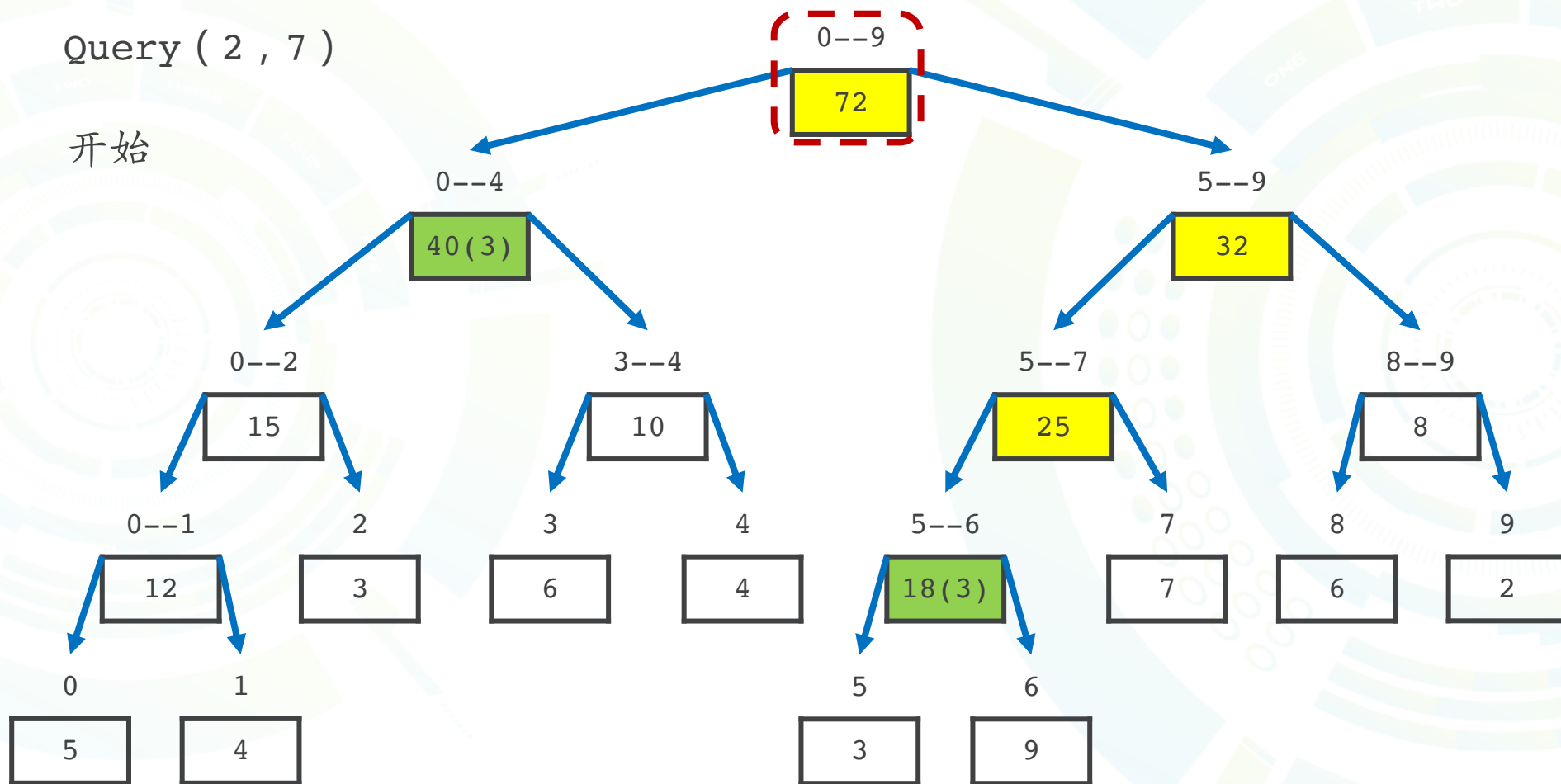
Query (2 , 7)



线段树—区间修改

Query (2 , 7)

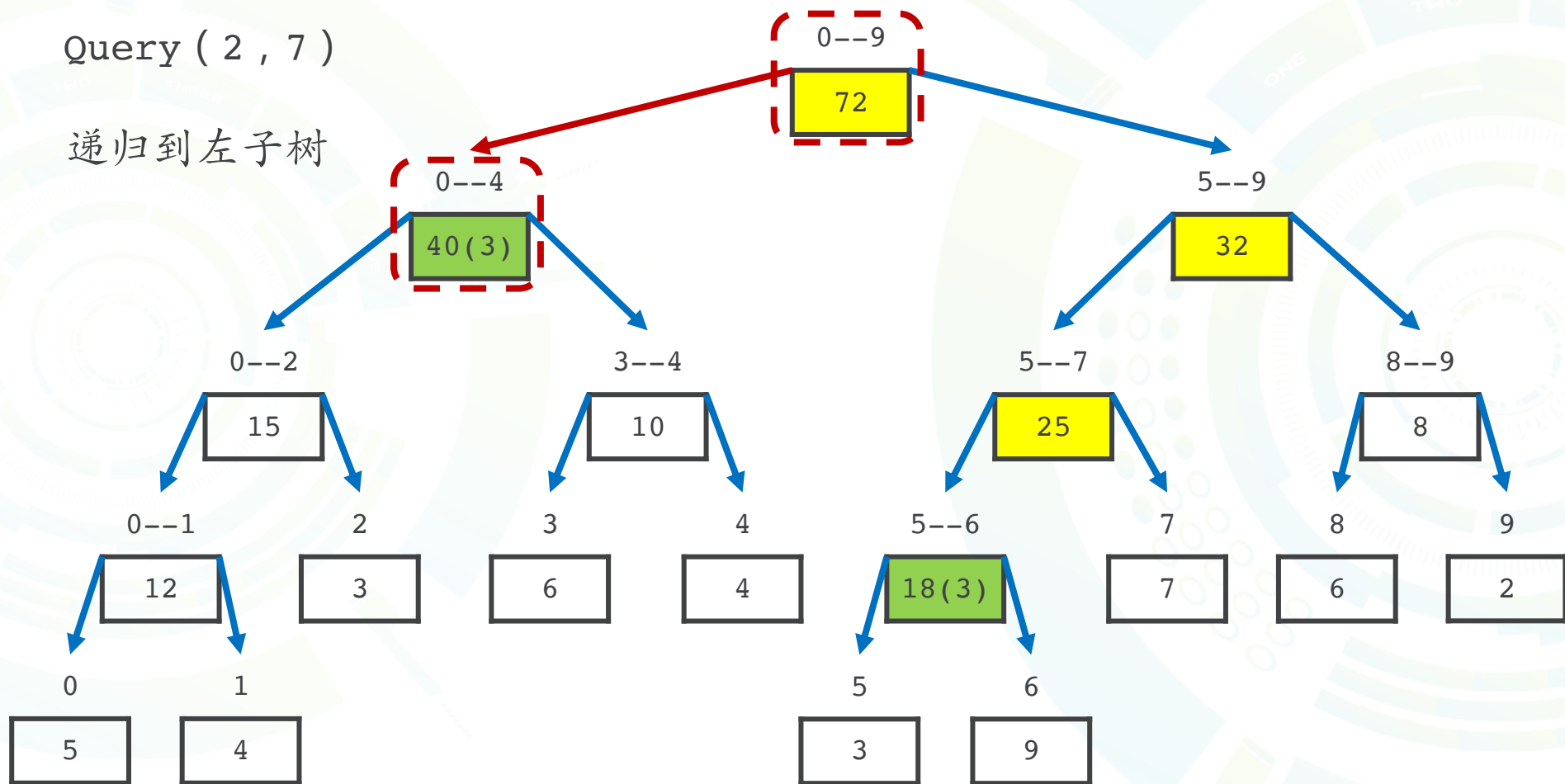
开始



线段树—区间修改

Query (2 , 7)

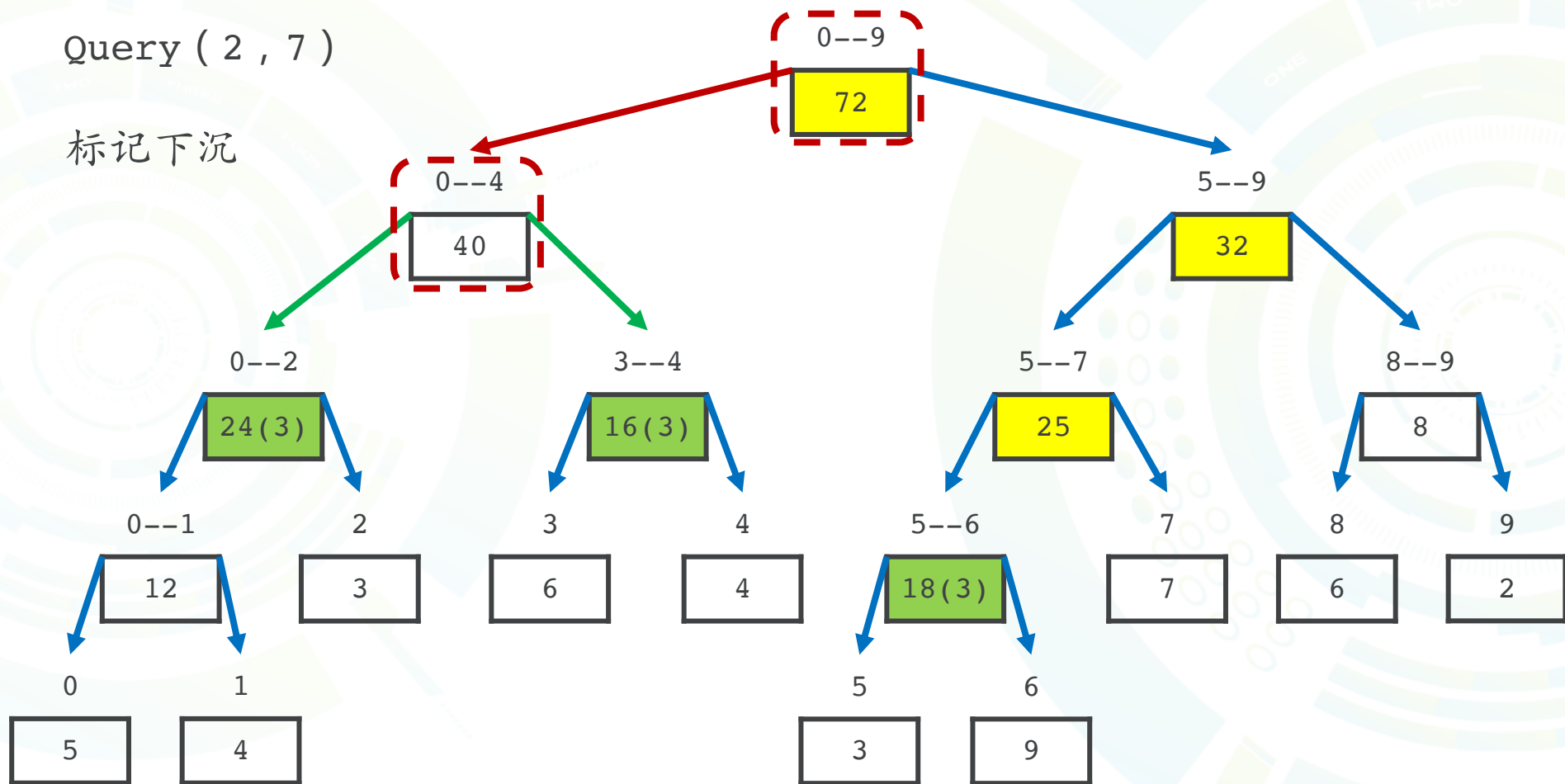
递归到左子树



线段树—区间修改

Query (2 , 7)

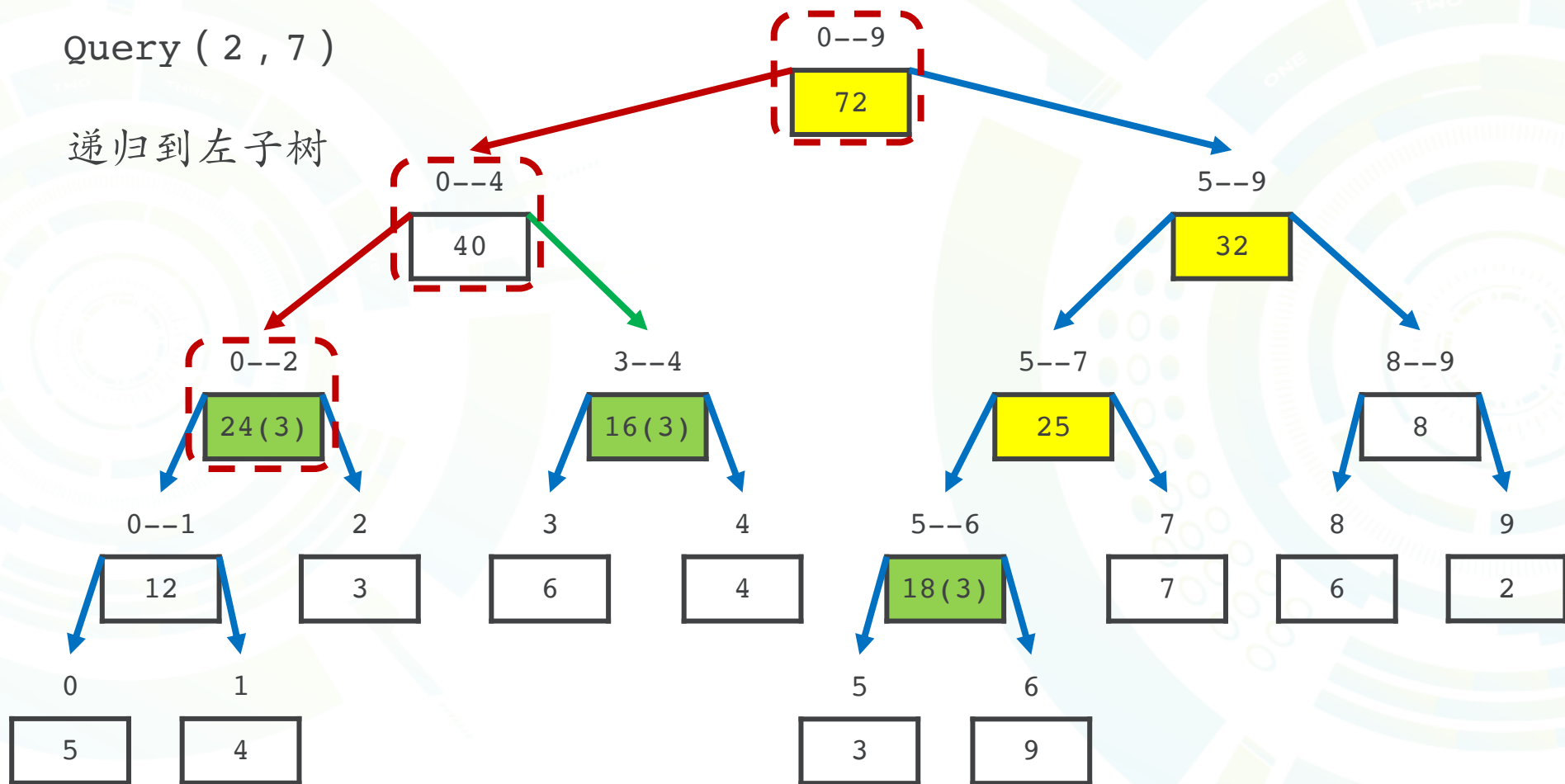
标记下沉



线段树—区间修改

Query (2 , 7)

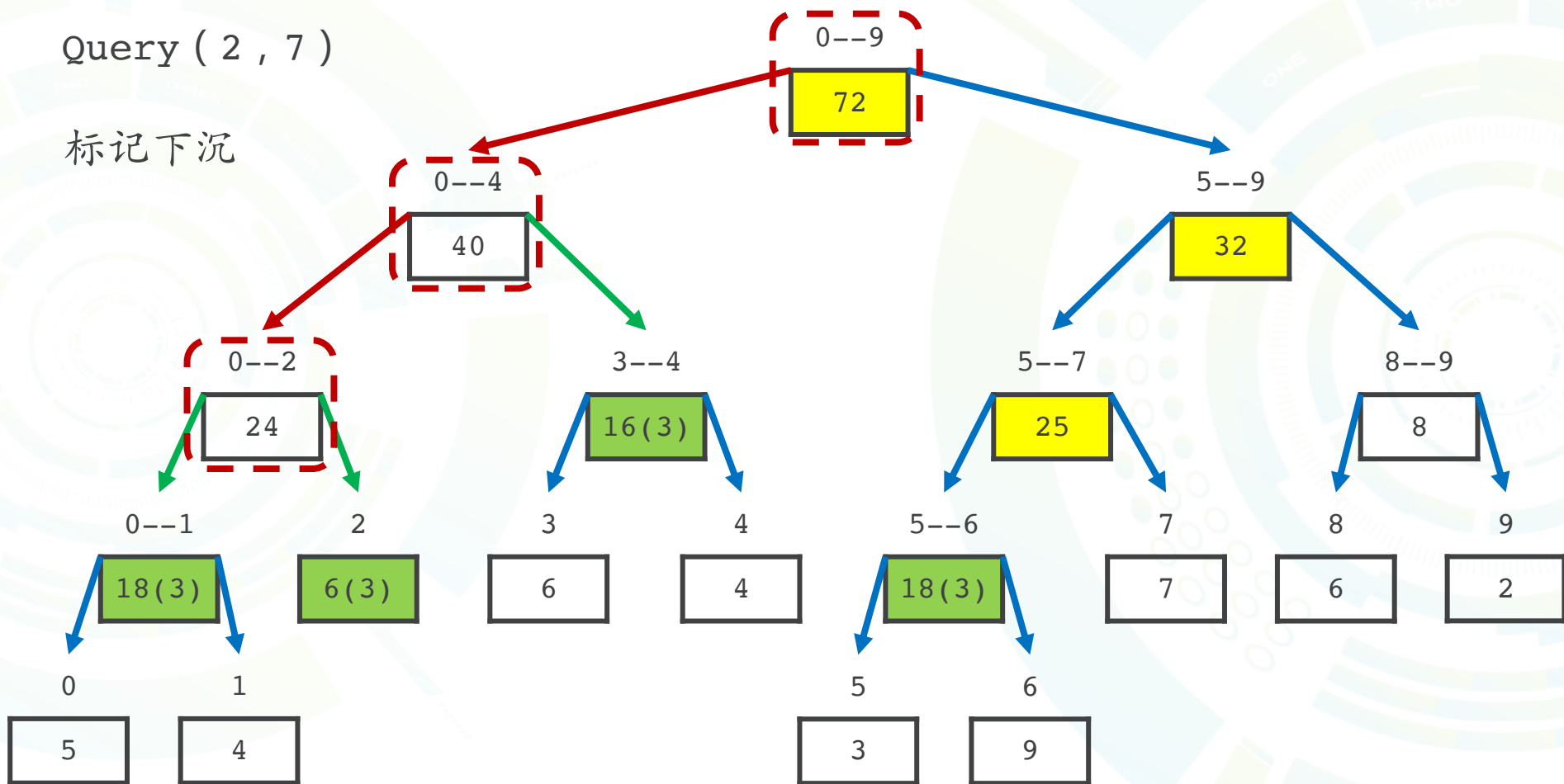
递归到左子树



线段树—区间修改

Query (2 , 7)

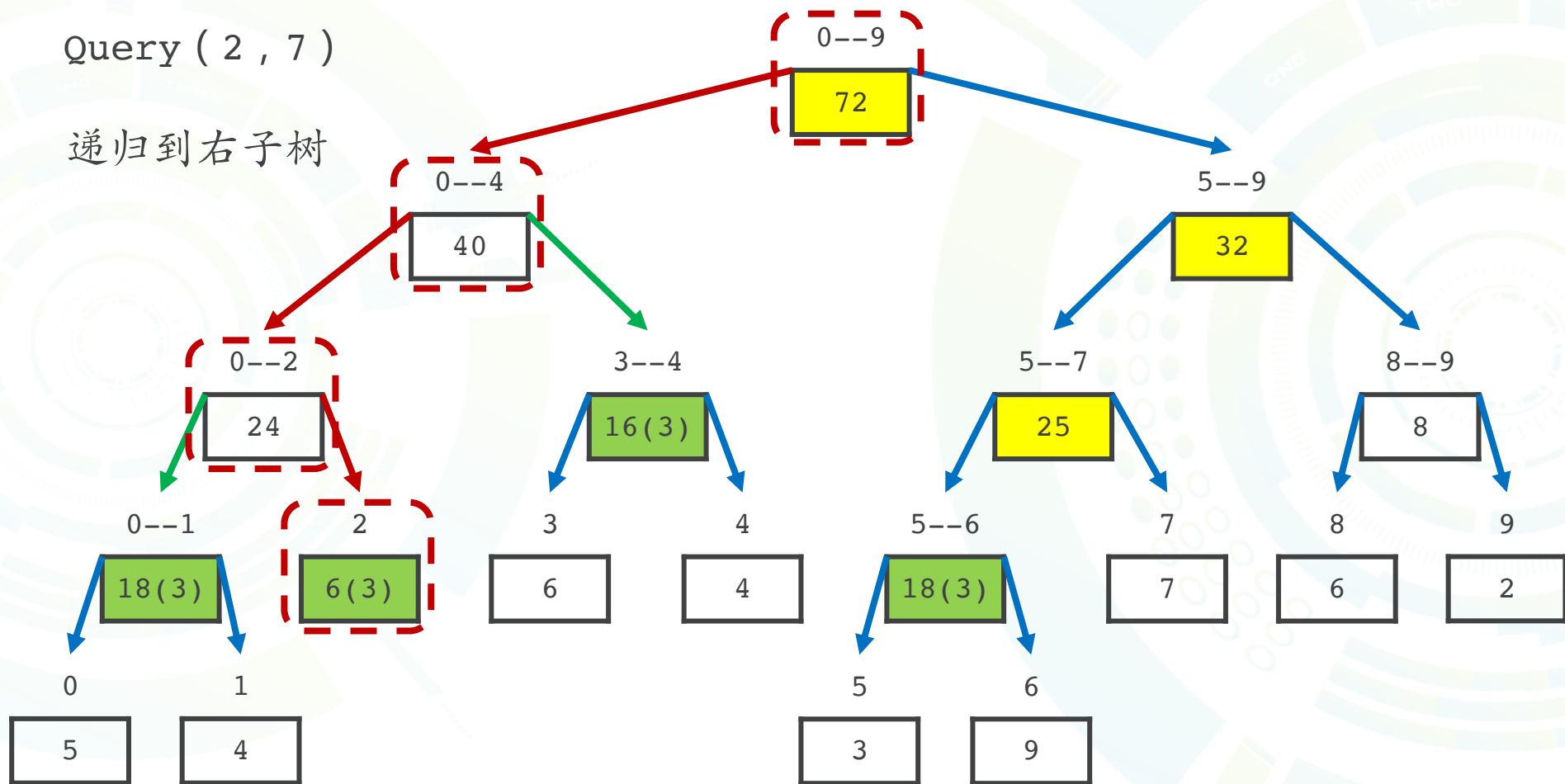
标记下沉



线段树—区间修改

Query (2 , 7)

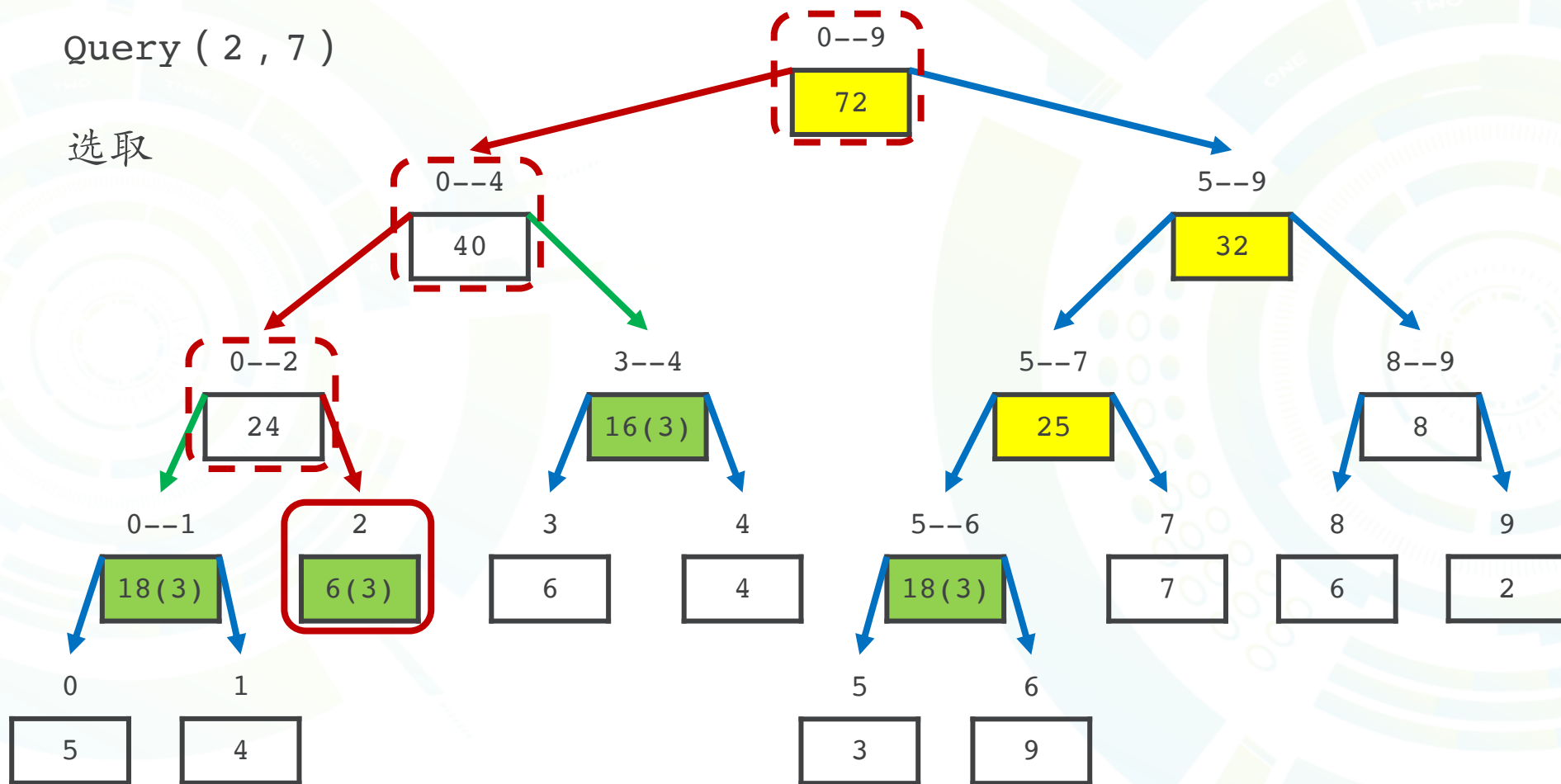
递归到右子树



线段树—区间修改

Query (2 , 7)

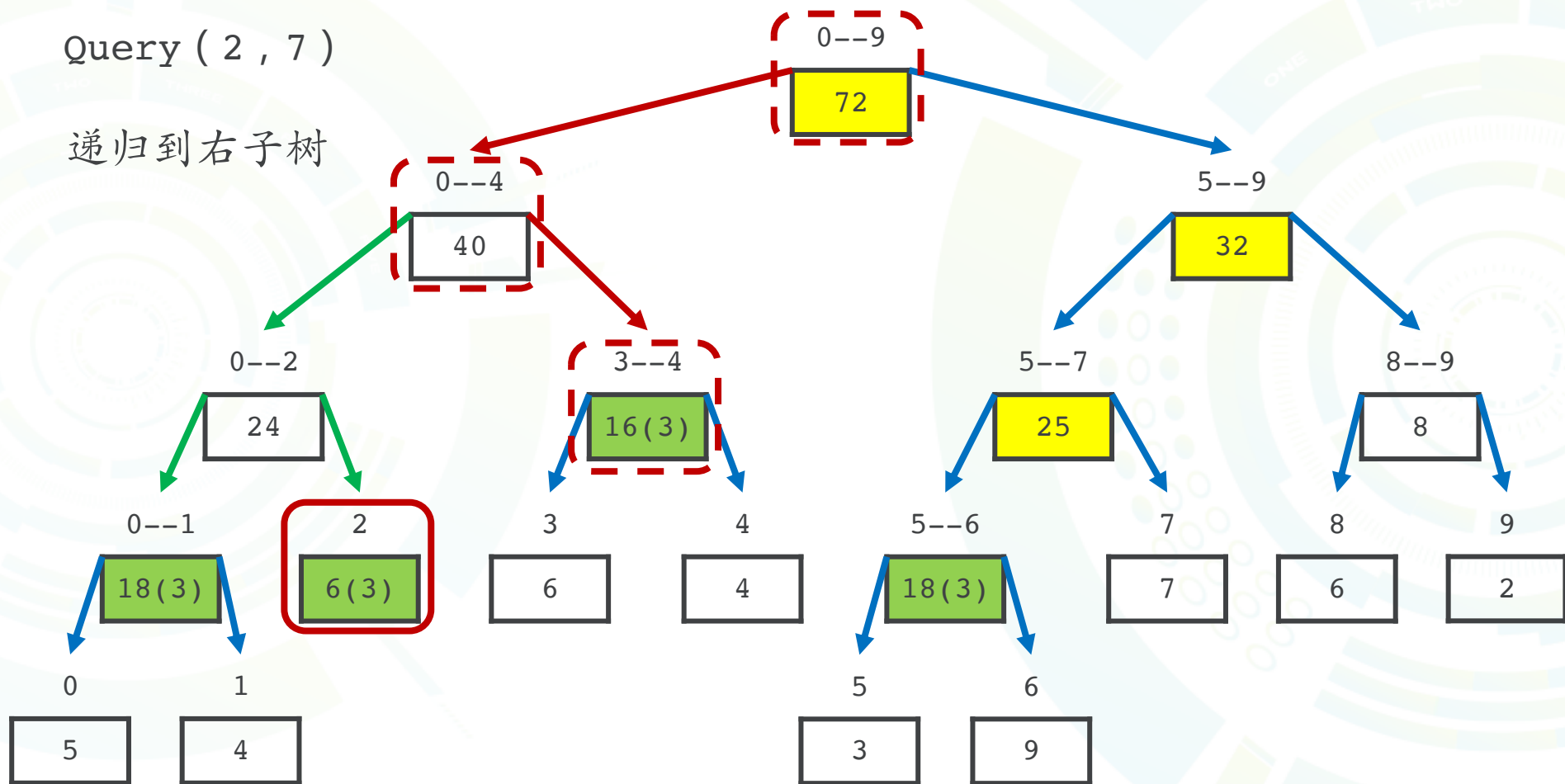
选取



线段树—区间修改

Query (2 , 7)

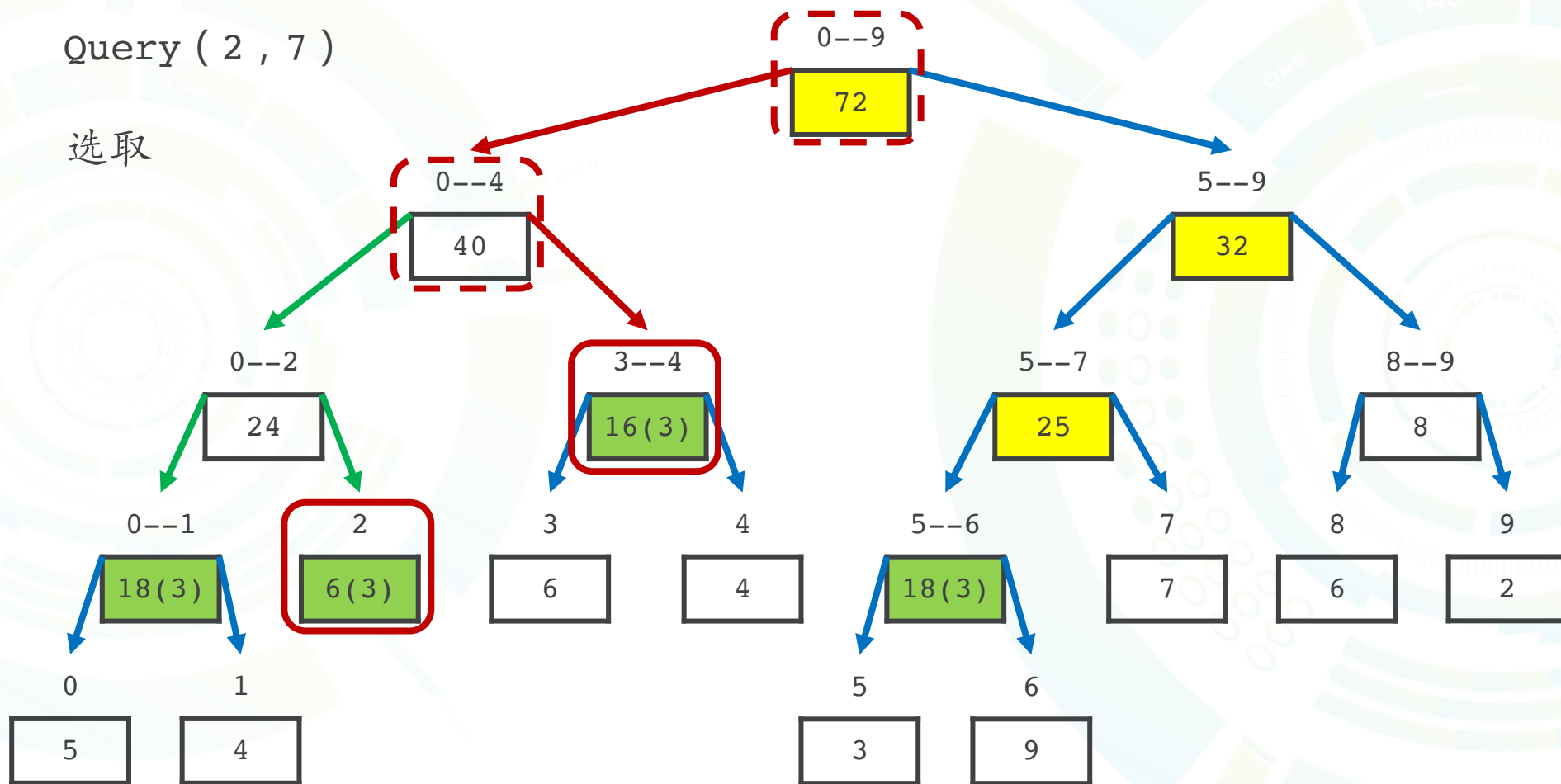
递归到右子树



线段树—区间修改

Query (2 , 7)

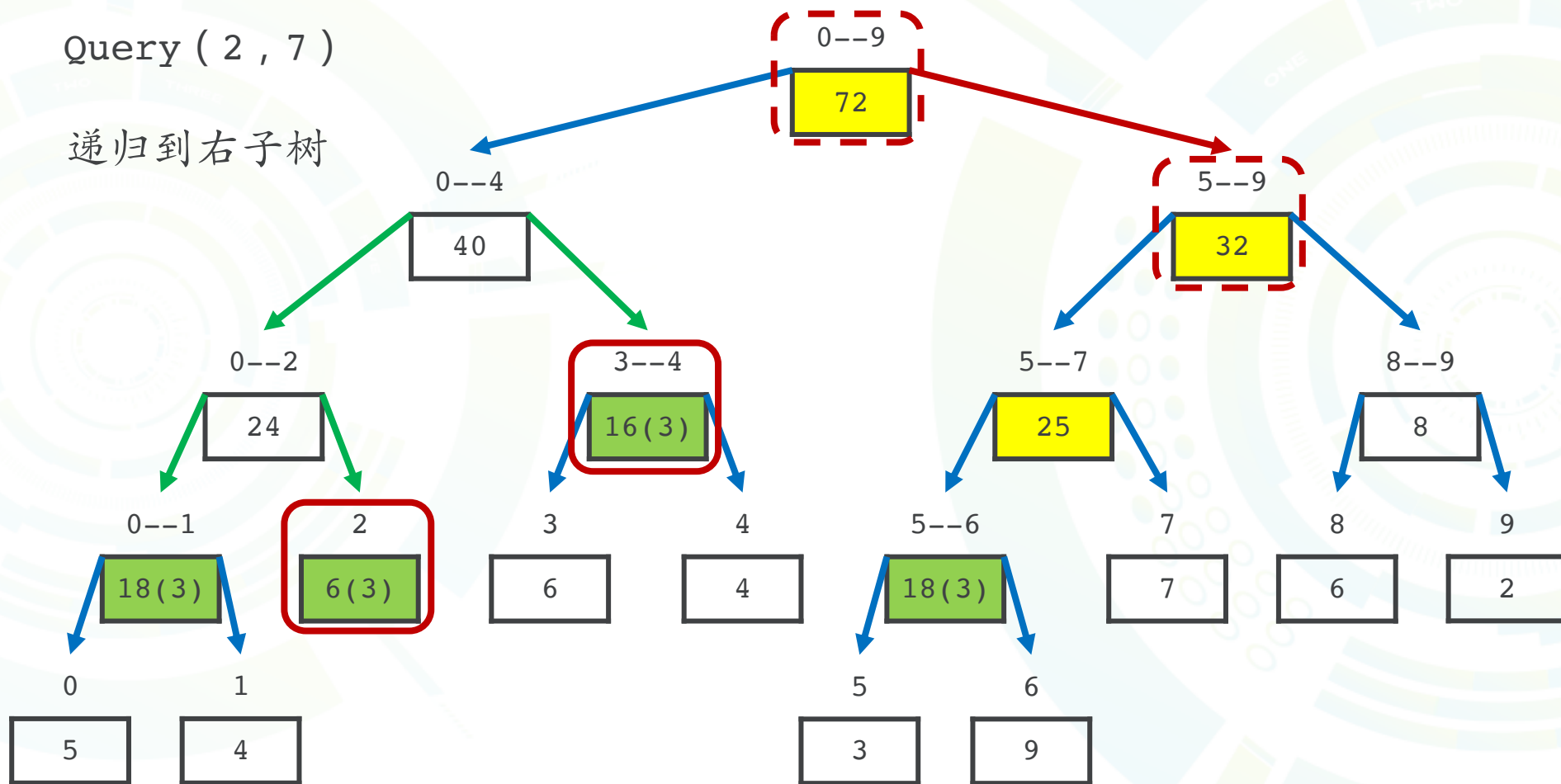
选取



线段树—区间修改

Query (2 , 7)

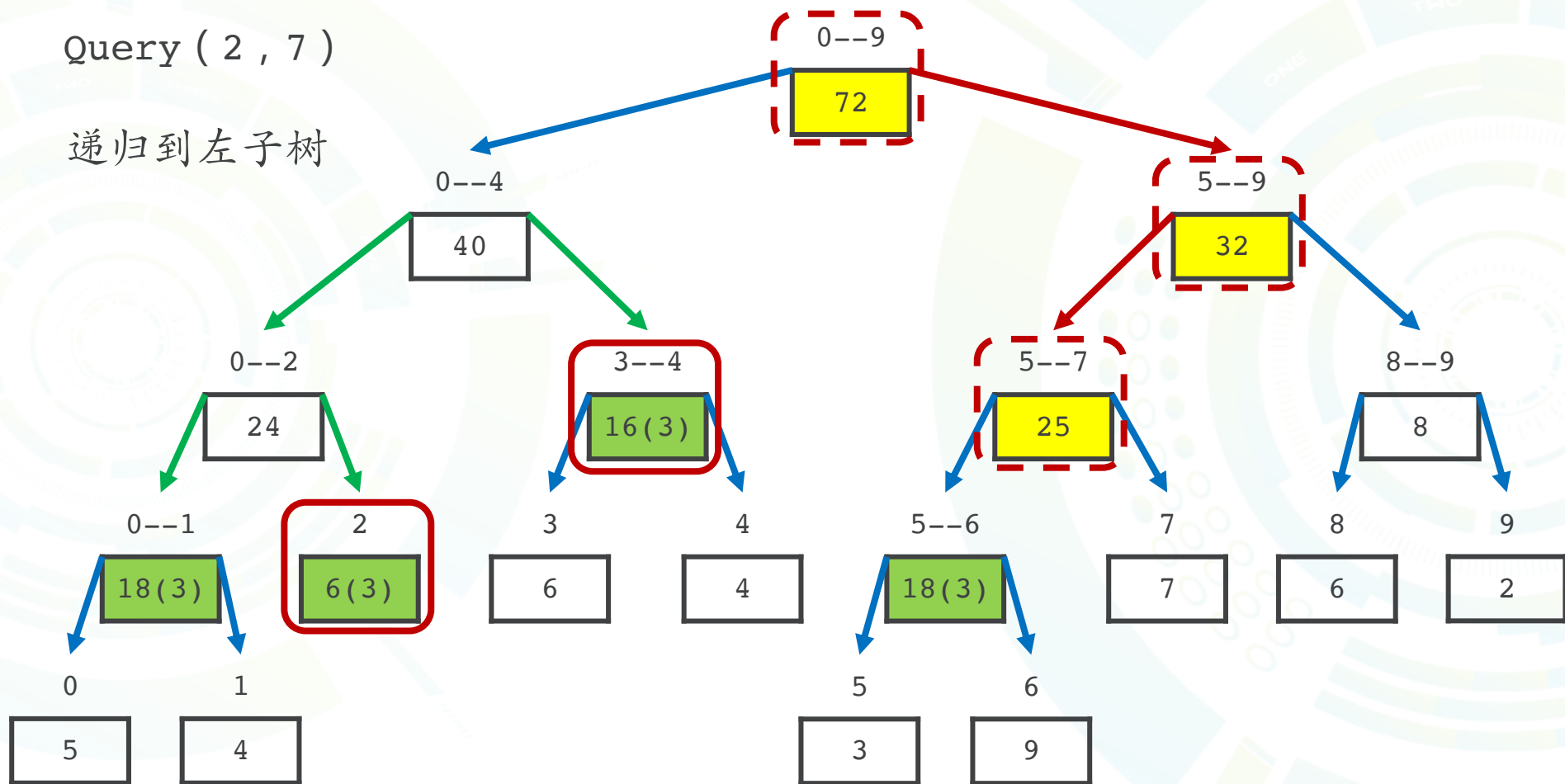
递归到右子树



线段树—区间修改

Query (2 , 7)

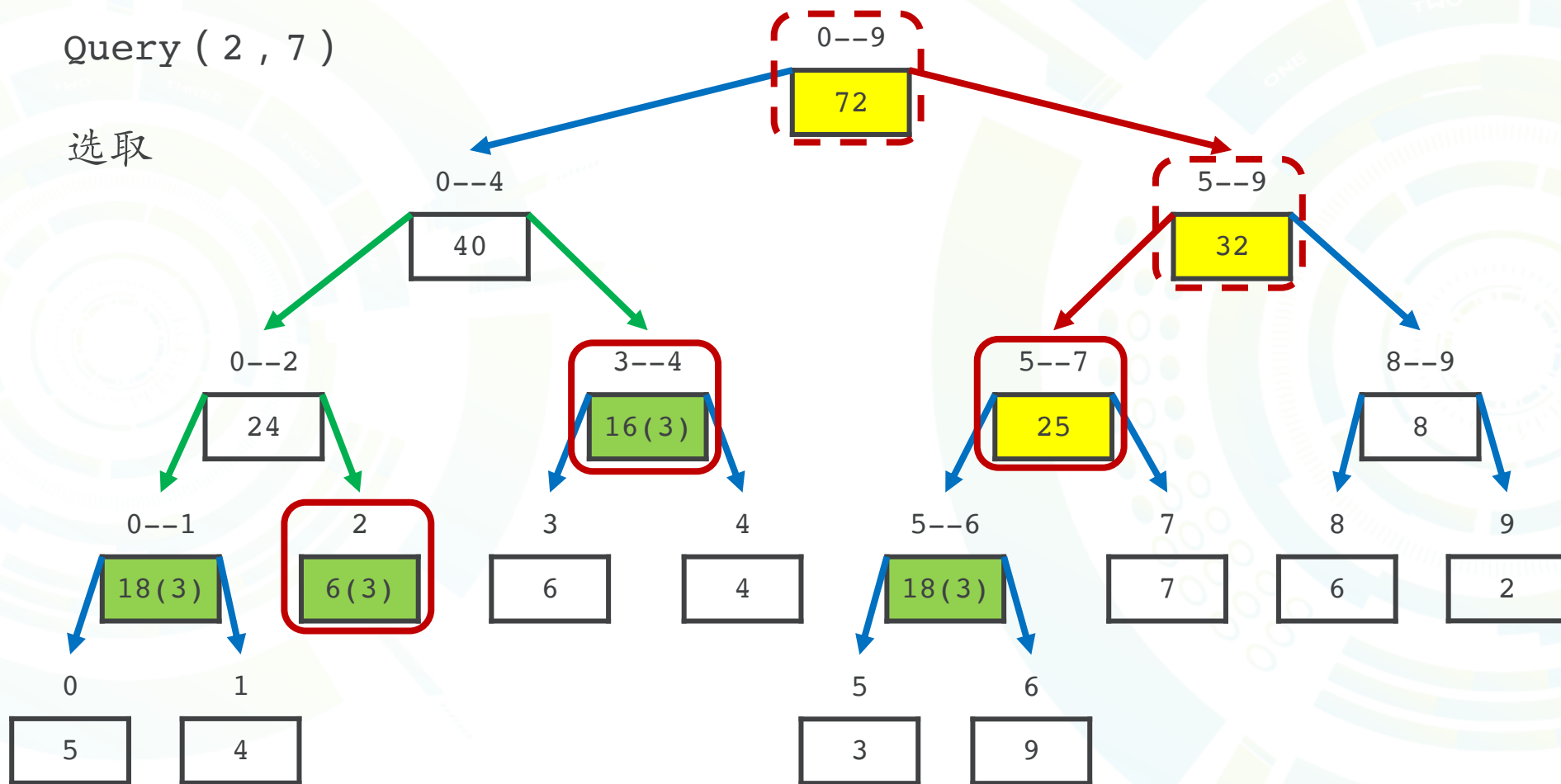
递归到左子树



线段树—区间修改

Query (2 , 7)

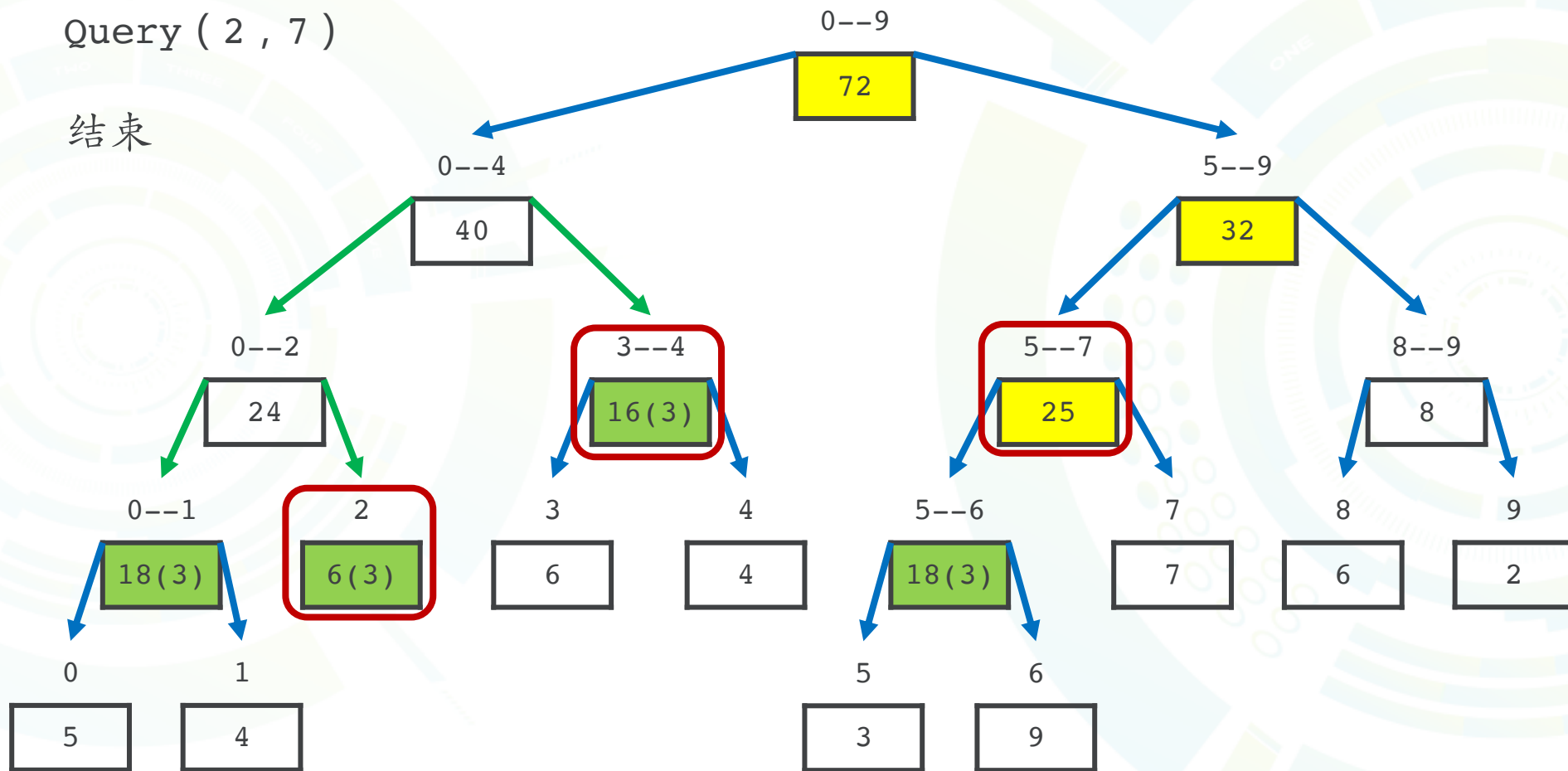
选取



线段树—区间修改

Query (2 , 7)

结束



线段树—关键词

完全二叉树：线段树程序实现时候的实际存储结构

区间：线段树中每个节点所维护的范围

向上更新：用两个子节点的信息更新本节点的信息

下沉标记：将本节点的懒惰标记更新给两个子节点

口诀：下沉发生在递归之前，向上发生在递归之后

练习题1：线段树模板（一）

【编写程序】 给定一个 n 位数组和两种操作，修改某个位置的值，和查询数组中某个区间的最大值。

0	1	2	3	4	5	6	7	8	9
5	4	8	6	4	3	9	7	6	2

Query	0	5	Ans:	8
Modify	2	3		
Query	0	5	Ans:	6
Modify	6	7		
Query	7	9	Ans:	7

练习题2：线段树模板（二）

【编写程序】 给定一个 n 位数组和两种操作，给某个区间的所有数加上某个值，和查询数组中某个区间的和值。

0	1	2	3	4	5	6	7	8	9
5	4	8	6	4	3	9	7	6	2

Query	0	5	Ans:	30
Modify	1	5	3	
Query	2	4	Ans:	27
Modify	4	7	4	
Query	5	9	Ans:	42

练习题 3：复合线段树

【编写程序】

给定一个 n 位数组和三种操作：

1. 给某个区间的所有数加上某个值
2. 给某个区间的所有数乘上某个值
3. 查询数组中某个区间的和值

0	1	2	3	4	5	6	7	8	9
5	4	8	6	4	3	9	7	6	2

Query	0	5	Ans:	30
Multiply	1	5	3	
Query	2	4	Ans:	54
Add	4	7	4	
Query	5	9	Ans:	45

习题作业

Luogu P3372

Luogu P3924

Luogu P1168

Luogu P1276

Luogu P1442

Luogu P1531

Luogu P1558

Luogu P1816

Luogu P3373

Luogu P1047

Luogu P1204

Luogu P1438

Luogu P1471

Luogu P1533

Luogu P1637

Luogu P2574