

慕课网《玩转算法面试》

玩儿转算法面试

讲师：liuyubobobo

版权所有，侵权必究

liuyubobobo

慕课网《玩转算法面试》

动态规划

讲师：lilyupobobo

版权所有，侵权必究

慕课网《玩转算法面试》

什么是动态规划?

讲师: liuyubobobo

版权所有, 侵权必究

斐波那契数列 Fibonacci Sequence

$$F(0)=1, F(1)=1, F(n)=F(n-1)+F(n-2)$$

```
int fib( int n ){  
    if( n == 0 )  
        return 0;  
  
    if( n == 1 )  
        return 1;  
  
    return fib(n-1) + fib(n-2);  
}
```

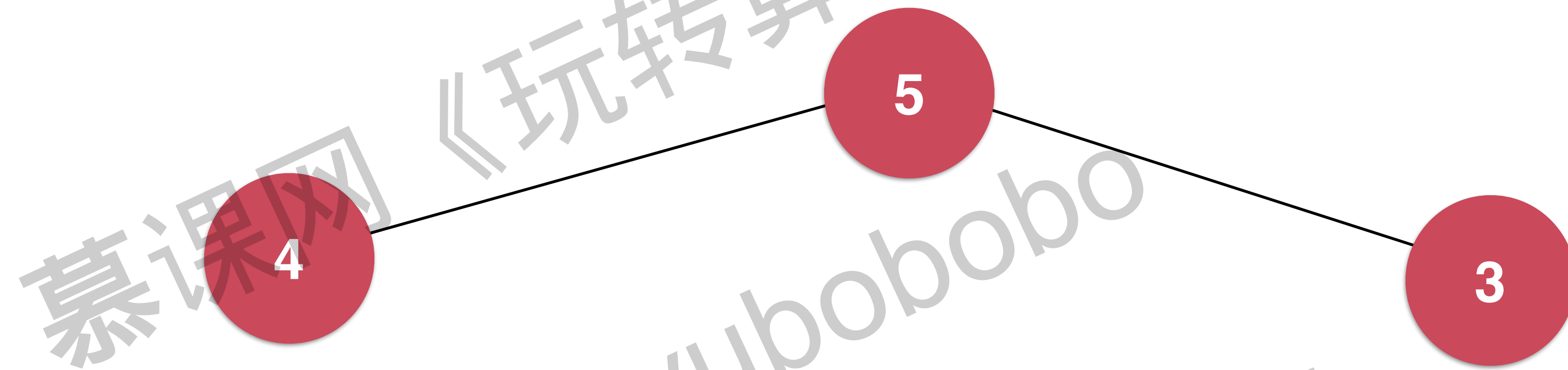
实践：递归实现fib的时间效率

慕课网《玩转算法面试》

讲师：iluyubobobo

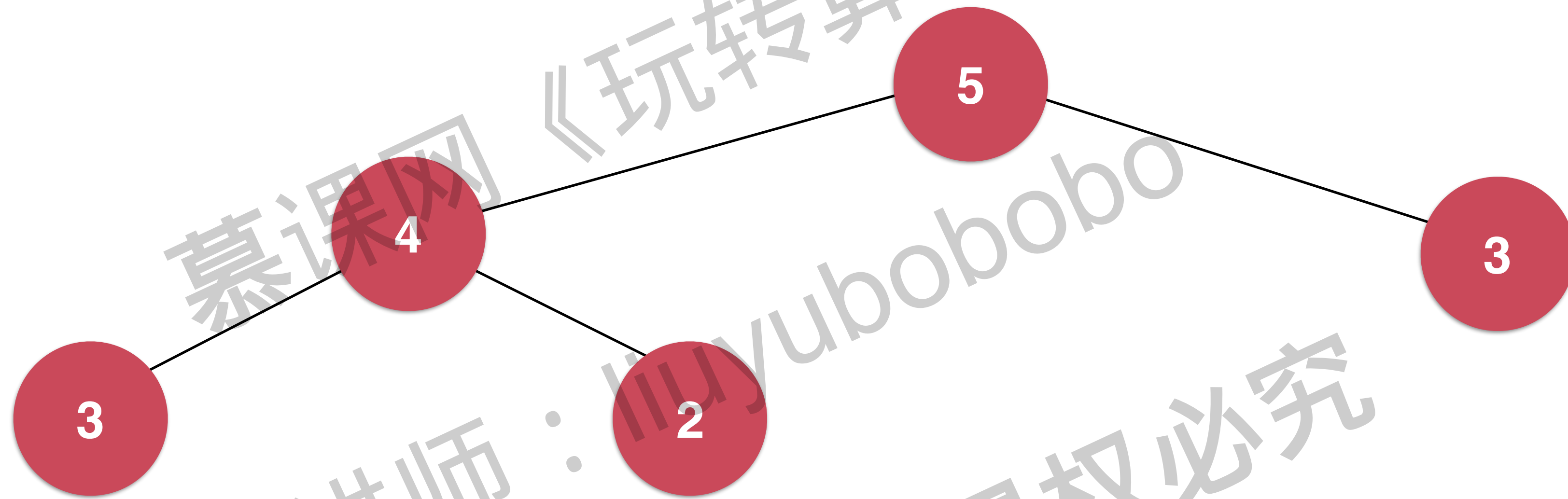
版权所有，侵权必究

斐波那契数列 Fibonacci Sequence

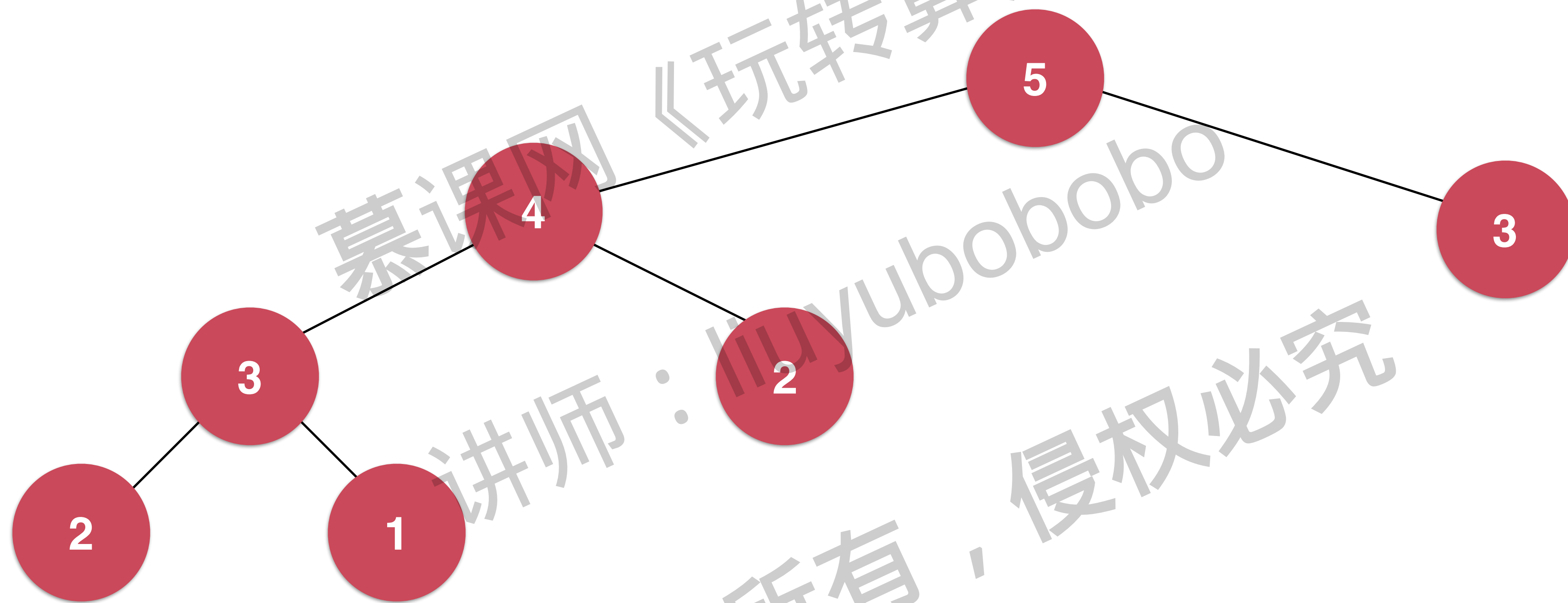


讲师：liuyubobobo
版权所有，侵权必究

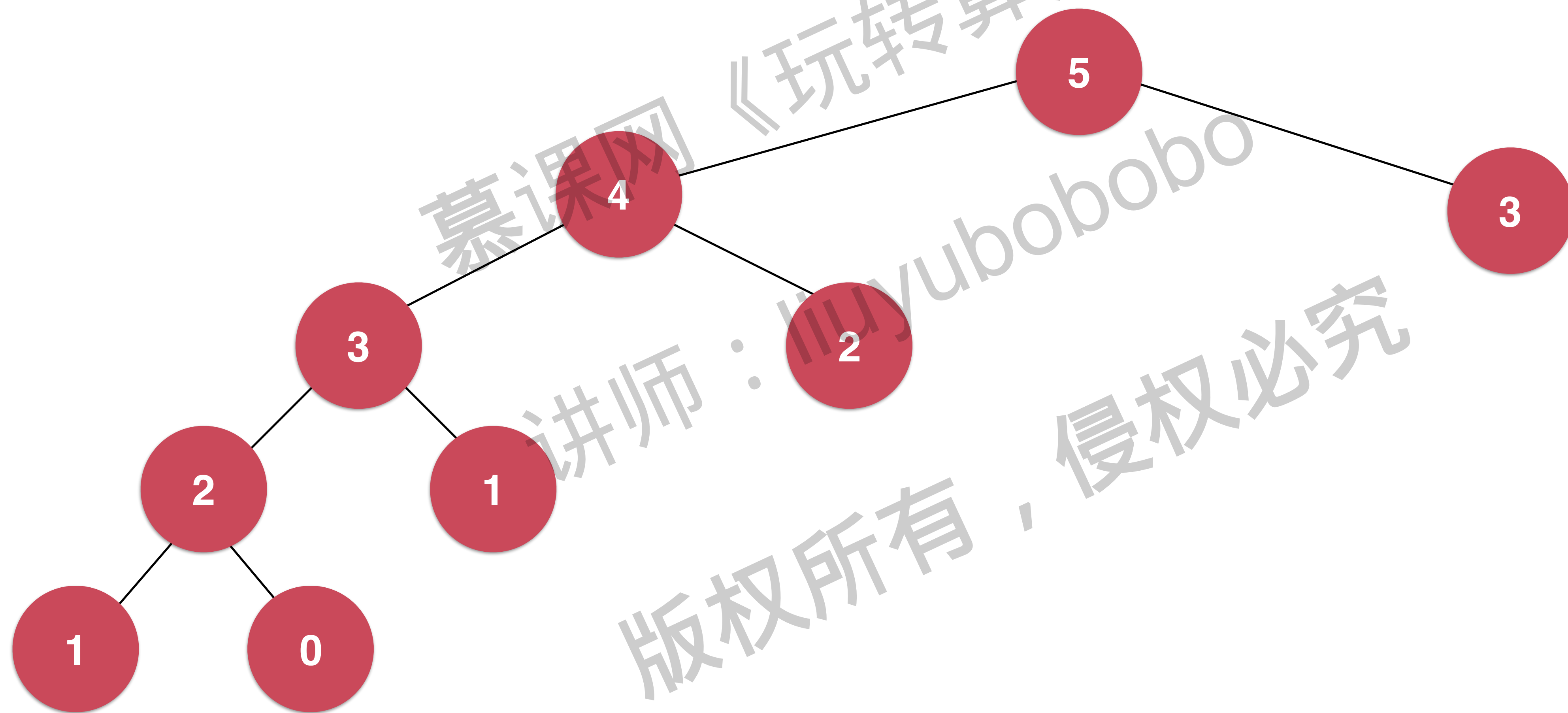
斐波那契数列 Fibonacci Sequence



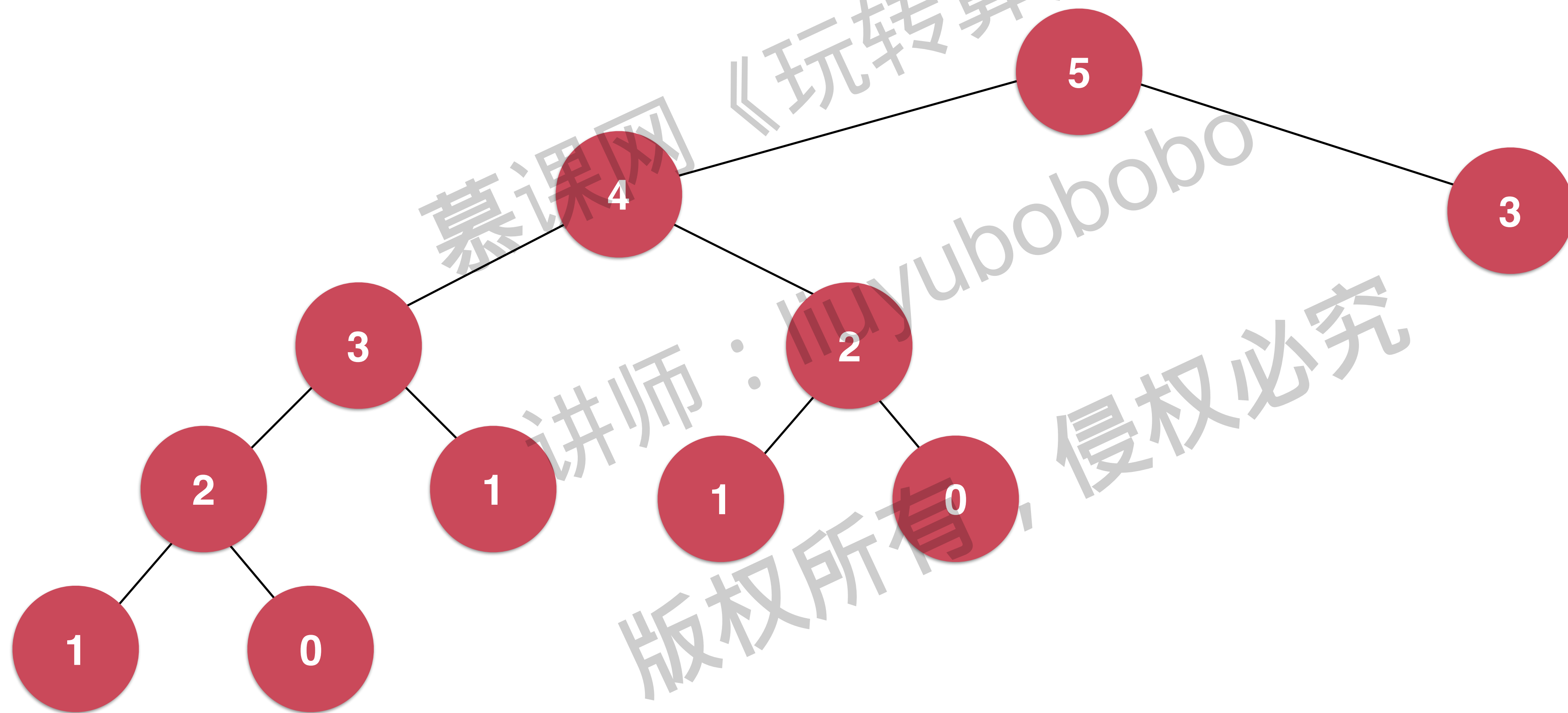
斐波那契数列 Fibonacci Sequence



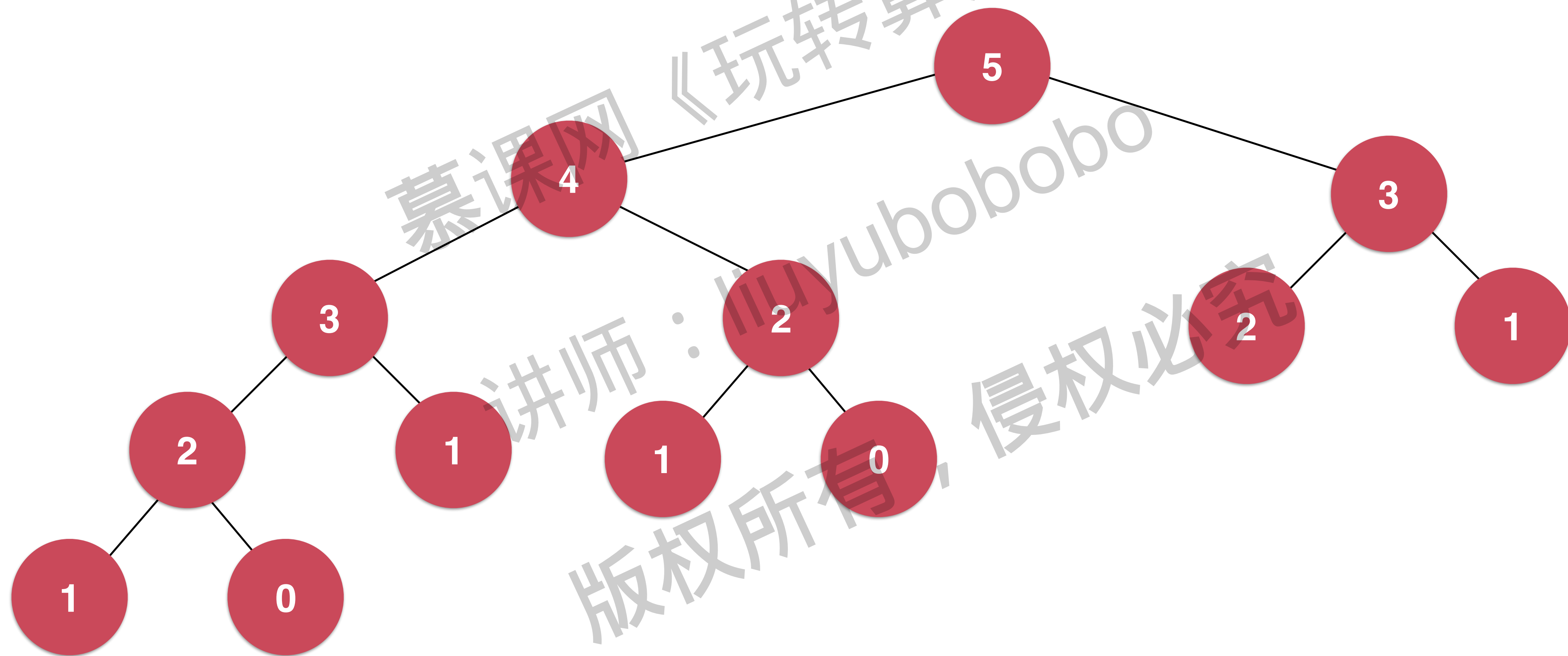
斐波那契数列 Fibonacci Sequence



斐波那契数列 Fibonacci Sequence



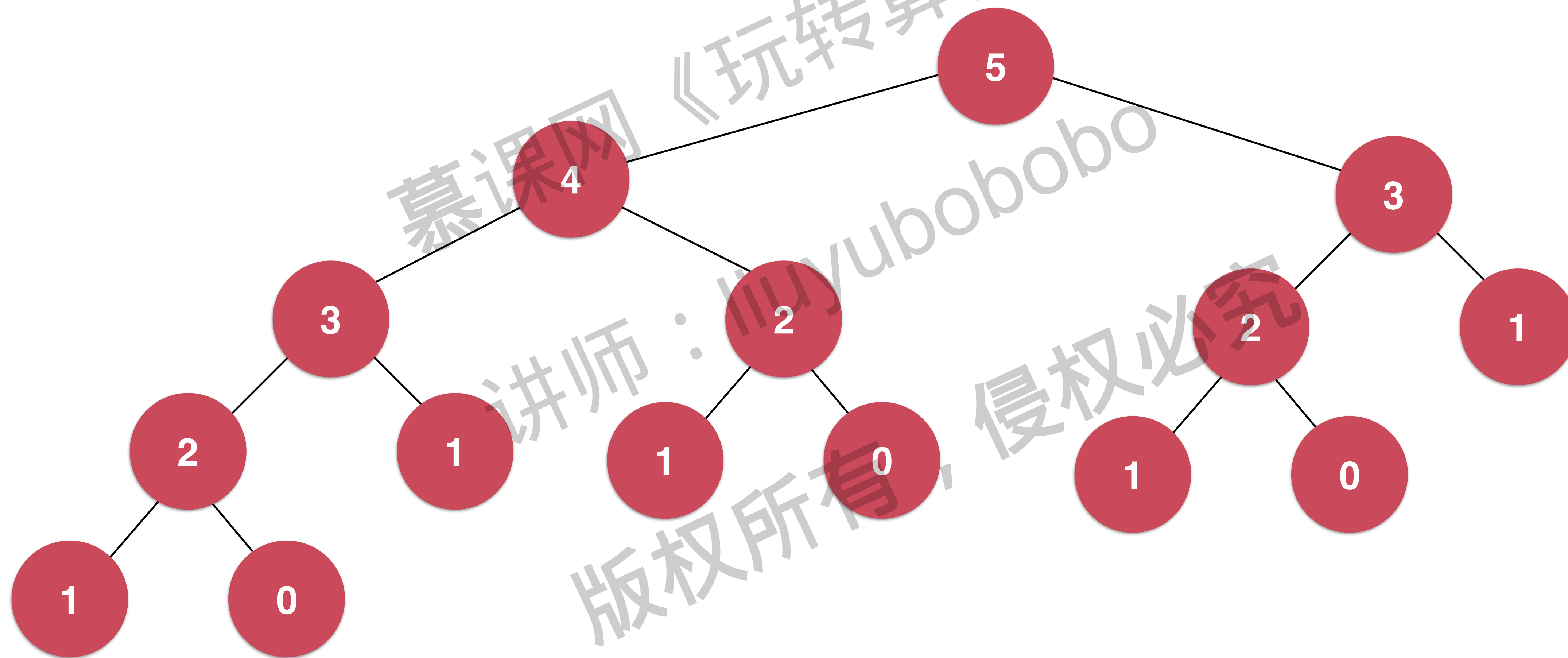
斐波那契数列 Fibonacci Sequence



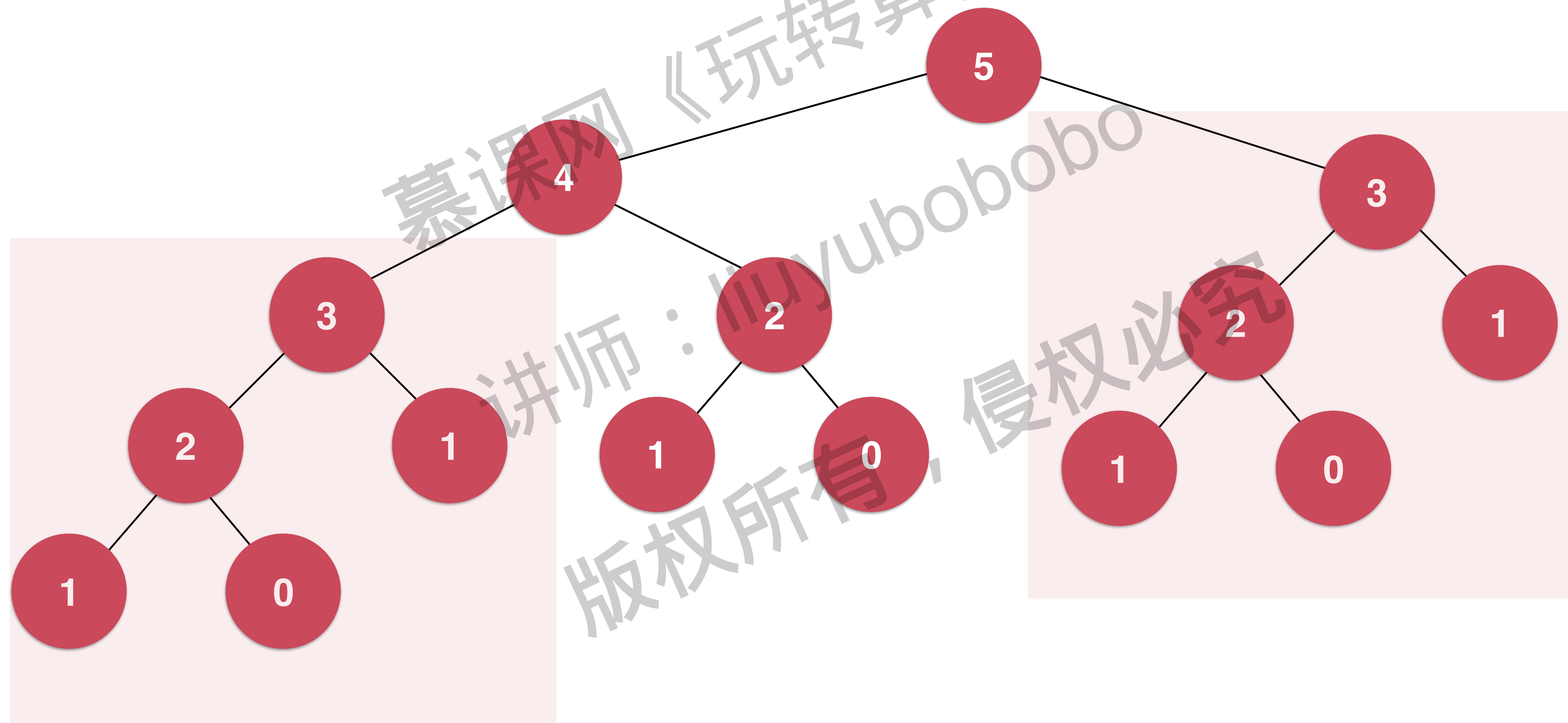
```

graph TD
    5((5)) --- 4((4))
    5 --- 2_1((2))
    4 --- 1_1((1))
    4 --- 2_2((2))
    2_1 --- 1_2((1))
    2_1 --- 0((0))
    2_2 --- 2_3((2))
  
```

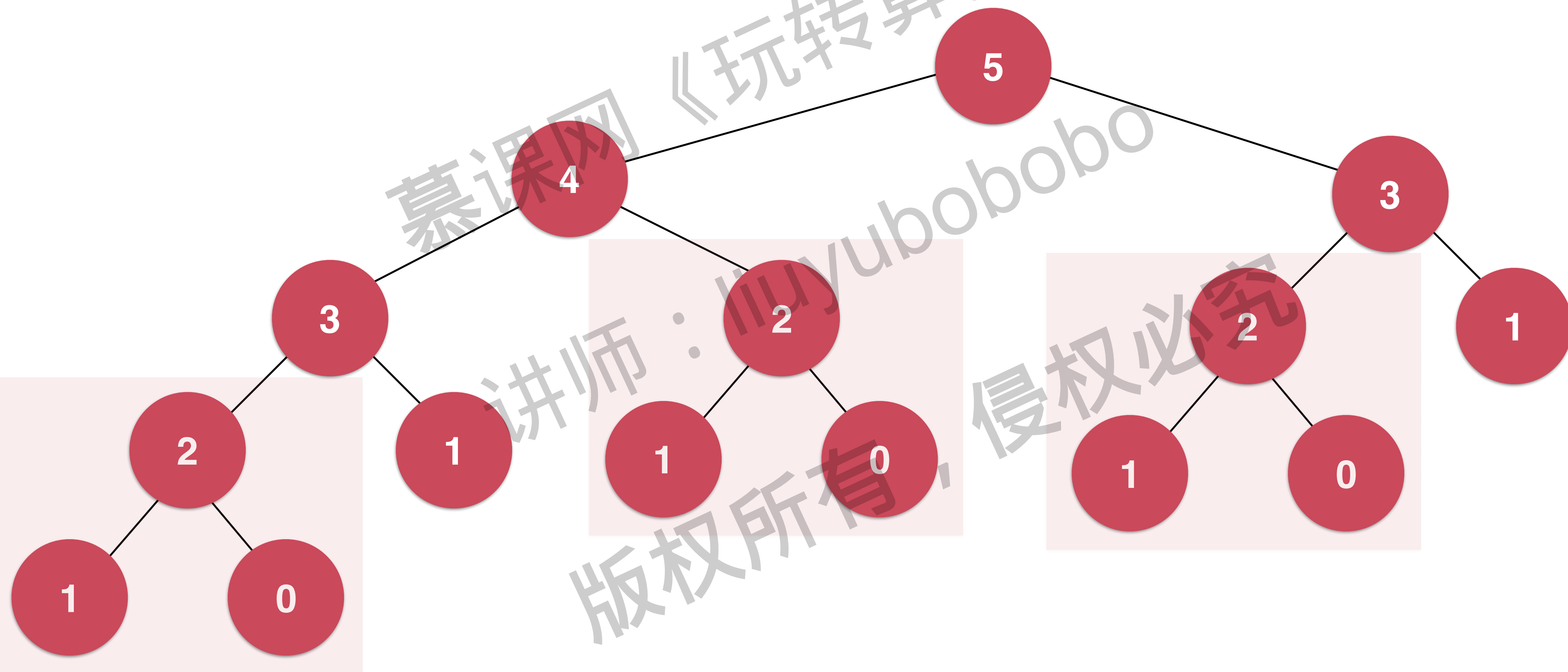
斐波那契数列 Fibonacci Sequence



斐波那契数列 Fibonacci Sequence



斐波那契数列 Fibonacci Sequence



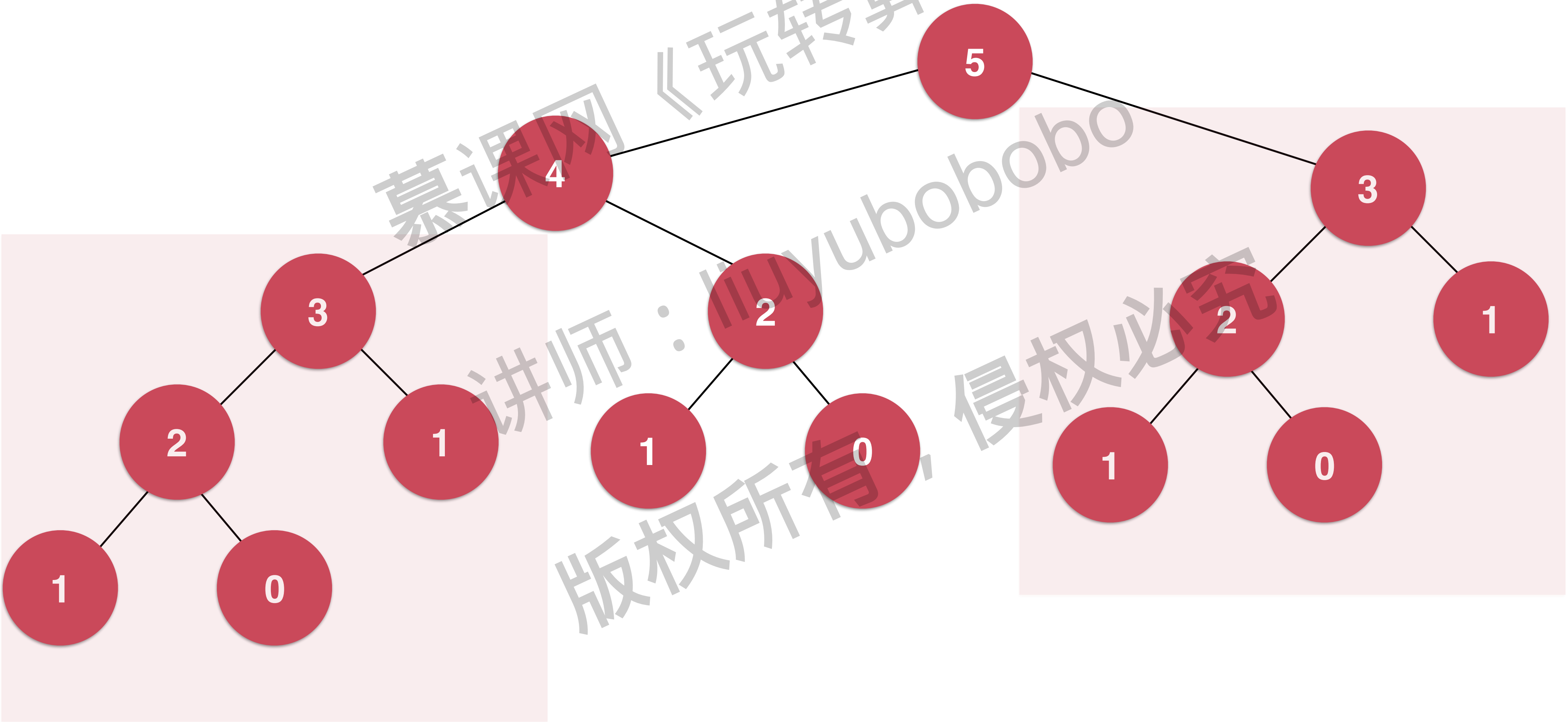
实践：递归实现fib的调用次数

慕课网《玩转算法面试》

讲师：iluyubobobo

版权所有，侵权必究

重叠子问题



记忆化搜索

```
vector<int> memo;  
  
int fib( int n ){  
    if( n == 0 )  
        return 0;  
  
    if( n == 1 )  
        return 1;  
  
    if( memo[n] == -1 )  
        memo[n] = fib(n-1) + fib(n-2);  
  
    return memo[n];  
}
```

实践：记忆化搜索实现fib的时间效率

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

实践：记忆花搜索实现fib的调用次数

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

记忆化搜索 - 自上向下的解决问题

```
vector<int> memo;  
  
int fib( int n ){  
    if( n == 0 )  
        return 0;  
  
    if( n == 1 )  
        return 1;  
  
    if( memo[n] == -1 )  
        memo[n] = fib(n-1) + fib(n-2);  
  
    return memo[n];  
}
```

动态规划 - 自下向上的解决问题

```
int fib( int n ){  
    vector<int> memo( n+1, -1 );  
  
    memo[0] = 0;  
    memo[1] = 1;  
    for( int i = 2 ; i <= n ; i ++ )  
        memo[i] = memo[i-1] + memo[i-2];  
  
    return memo[n];  
}
```

实践：动态规划实现fib的时间效率

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

动态规划

dynamic programming (also known as dynamic optimization) is a method for solving a complex problem by breaking it down into a collection of simpler subproblems, solving each of those subproblems just once, and storing their solutions – ideally, using a memory-based data structure.

将原问题拆解成若干子问题，同时保存子问题的答案，使得每个子问题只求解一次，最终获得原问题的答案。

动态规划



慕课网《玩转算法面试》

第一个动态规划问题

讲师：liuyubobobo

版权所有，侵权必究

70. Climbing Stairs



有一个楼梯，总共有 n 阶台阶。每一次，可以上一个台阶，也可以上两个台阶。问，爬上这样的一个楼梯，一共有多少不同的方法？

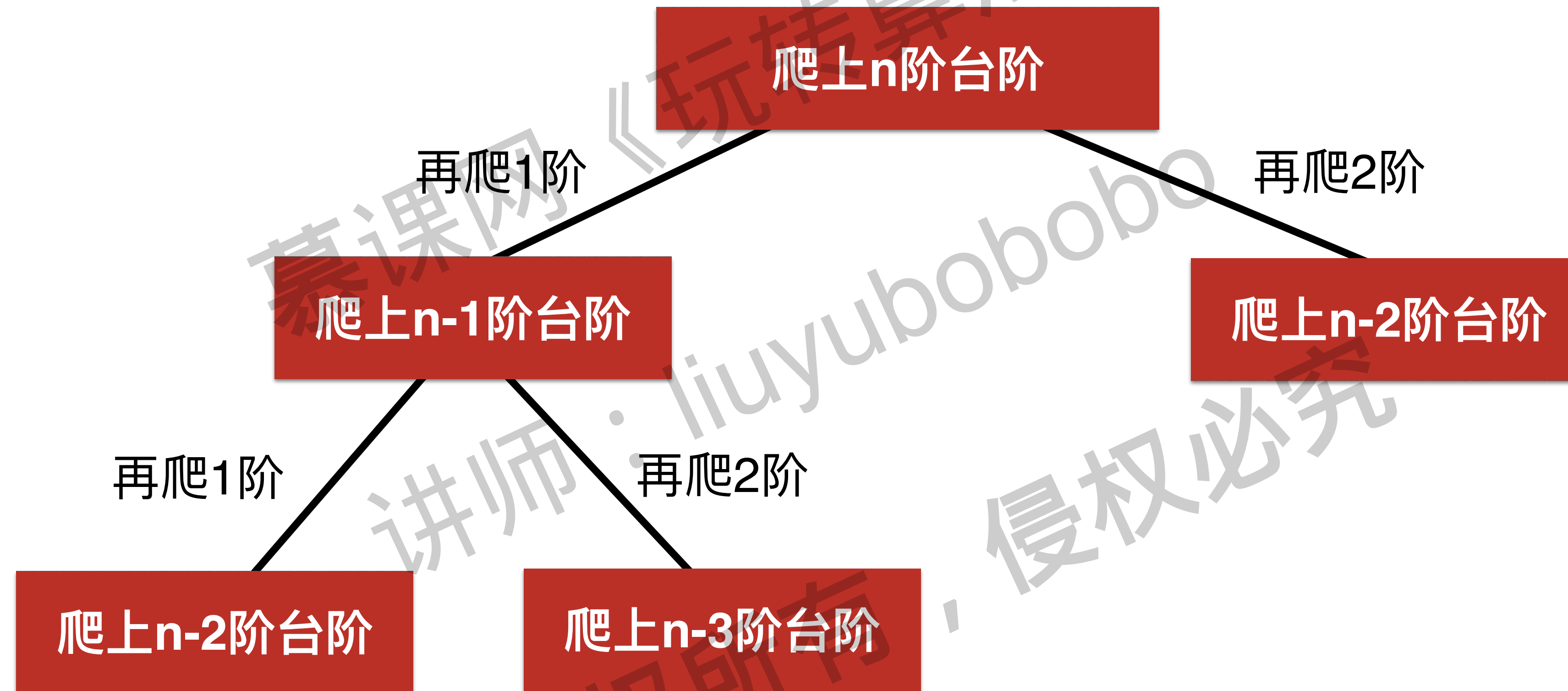
- 如 $n = 3$ ，可以爬上这个楼梯的方法有： $[1,1,1]$ ， $[1,2]$ ， $[2,1]$
- 所以答案为3

70. Climbing Stairs

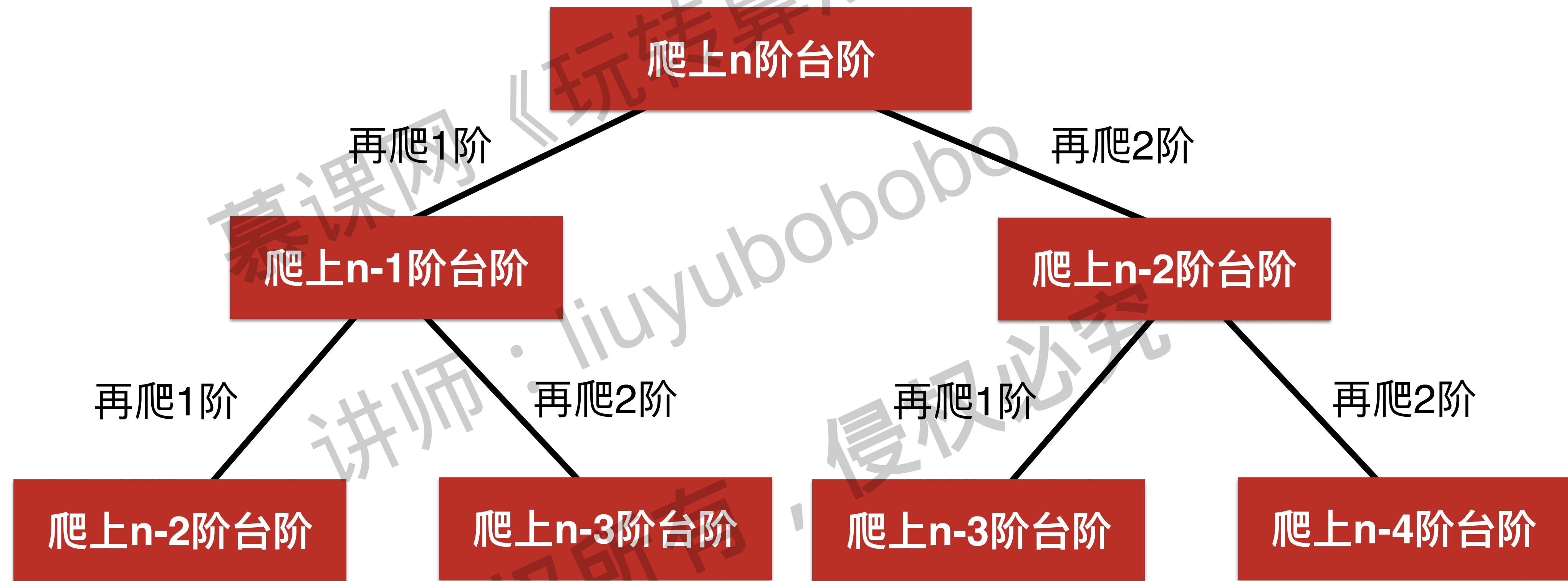


讲师：liuyubobobo
版权所有，侵权必究

70. Climbing Stairs



70. Climbing Stairs



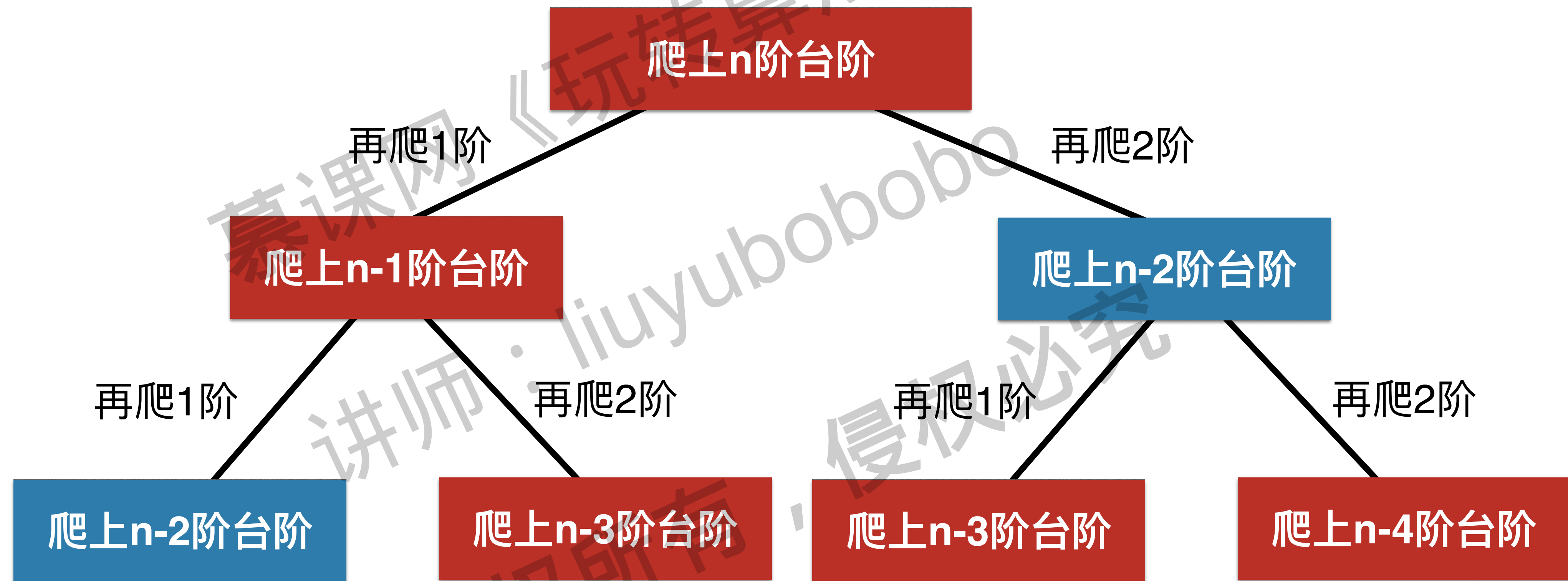
实践：使用递归解决问题

慕课网《玩转算法面试》

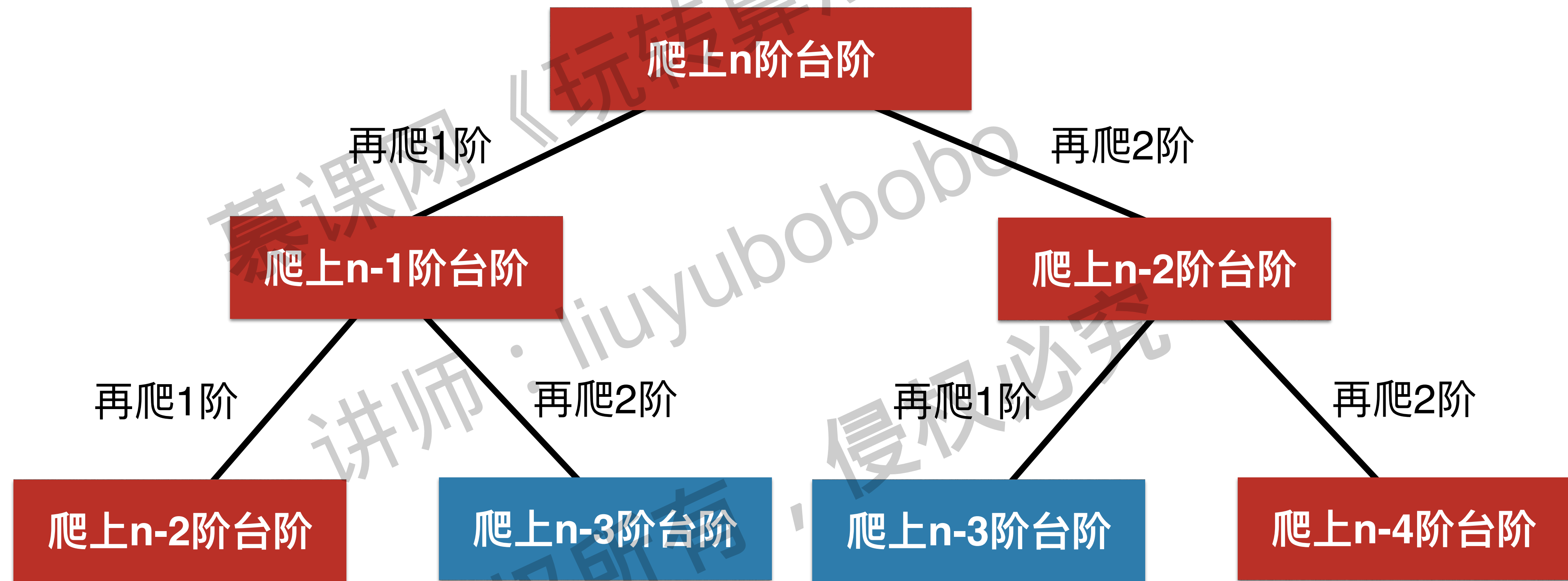
讲师：liuyubobobo

版权所有，侵权必究

70. Climbing Stairs



70. Climbing Stairs



实践：使用记忆化搜索

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

慕课网《玩转算法面试》

实践：使用动态规划

讲师：liuyubobobo

版权所有，侵权必究

120. Triangle

给定一个三角形的数字阵列，选择一条自顶向下的路径，使得沿途的所有数字之和最小。(每一步只能移动到相邻的格子中。)

- 如右图所示的三角形阵列

- 其最小路径和为11

- $(2 + 3 + 5 + 1 = 11)$

[2],

[3, 4],

[6, 5, 7],

[4, 1, 8, 3]

]

64. Minimum Path Sum

给出一个 $m * n$ 的矩阵，其中每一个格子包含一个非负整数。寻找一条从左上角到右下角的路径，使得沿路的数字和最小。

- 每一步只能右移或者下移。

慕课网《玩转算法面试》

发现重叠子问题

讲师：lilyubobobo

版权所有，侵权必究

343. Integer Break

给定一个正数 n ，可以将其分割成多个数字的和，若要让这些数字的乘积最大，求分割的方法（至少要分成两个数）。算法返回这个最大的乘积。

- 如 $n = 2$ ，则返回1 （ $2 = 1 + 1$ ）
- 如 $n = 10$ ，则返回36 （ $10 = 3 + 3 + 4$ ）

343. Integer Break

给定一个正数 n ，可以将其分割成多个数字的和，若要让这些数字的乘机最大，求分割的方法（至少要分成两个数）。算法返回这个最大的乘积。

暴力解法：回溯遍历将一个数做分割的所有可能性。 $O(2^n)$

343. Integer Break

分割4获得最大乘积

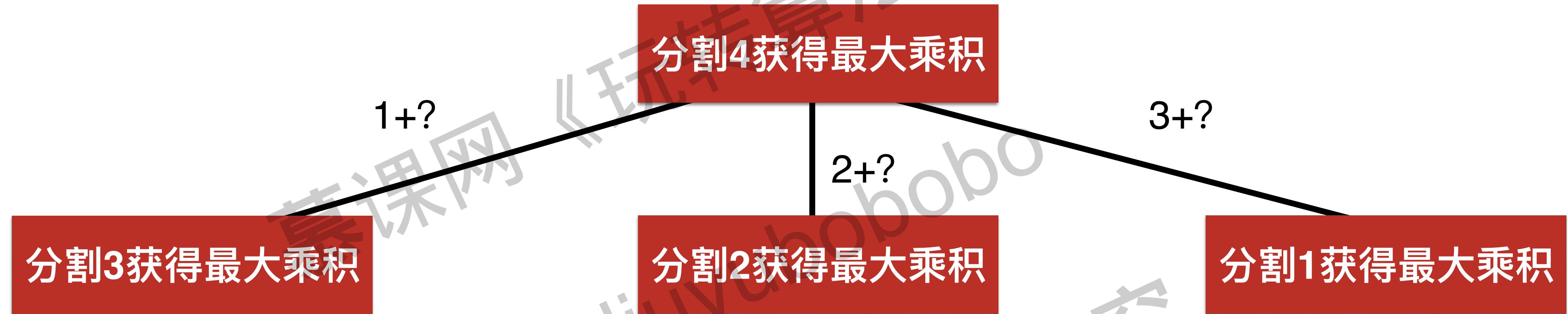
慕课网

《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

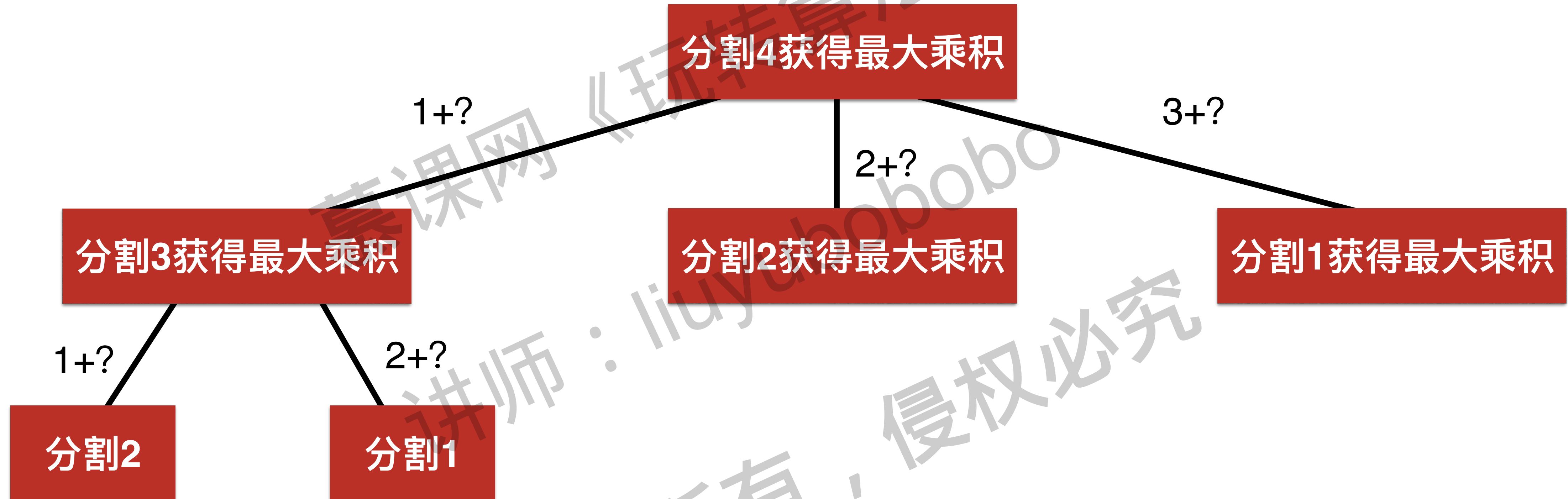
343. Integer Break



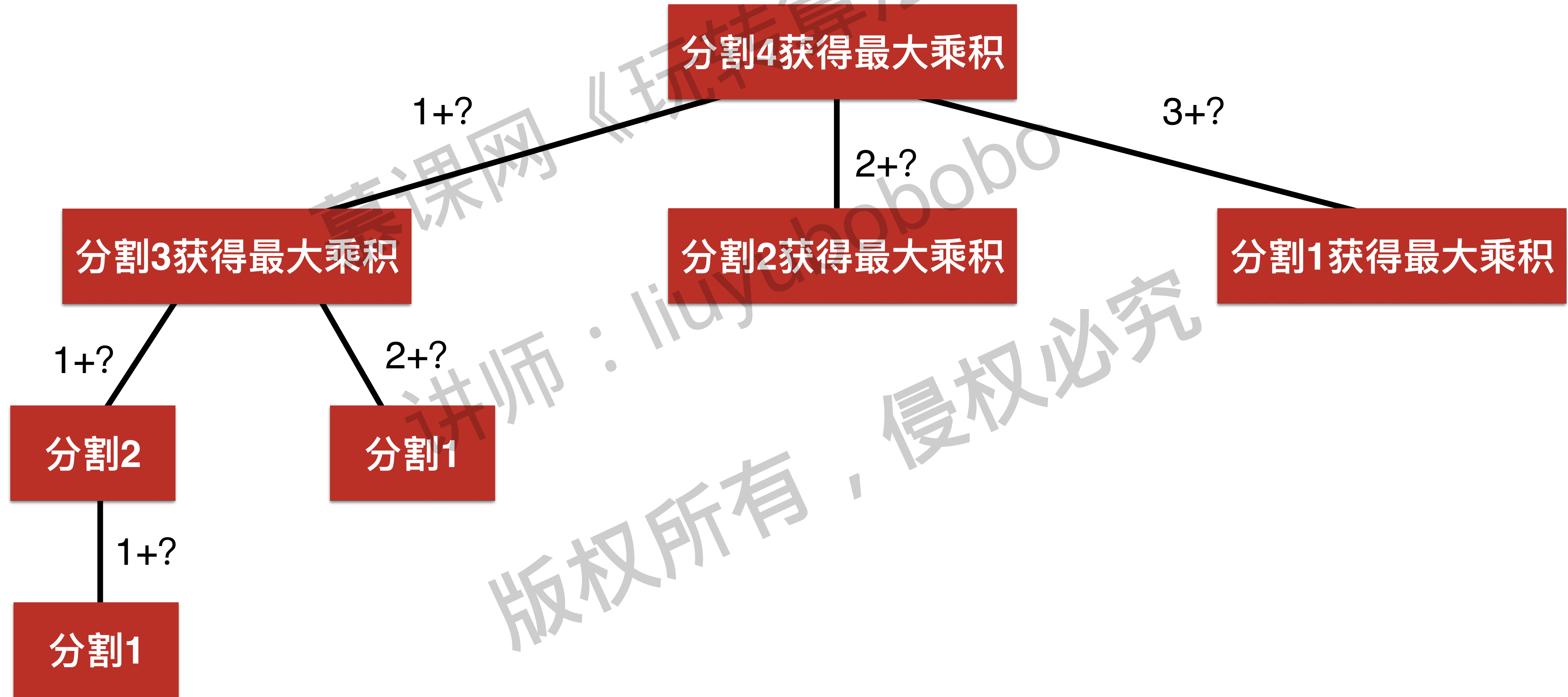
讲师：liuyuboboo

版权所有，侵权必究

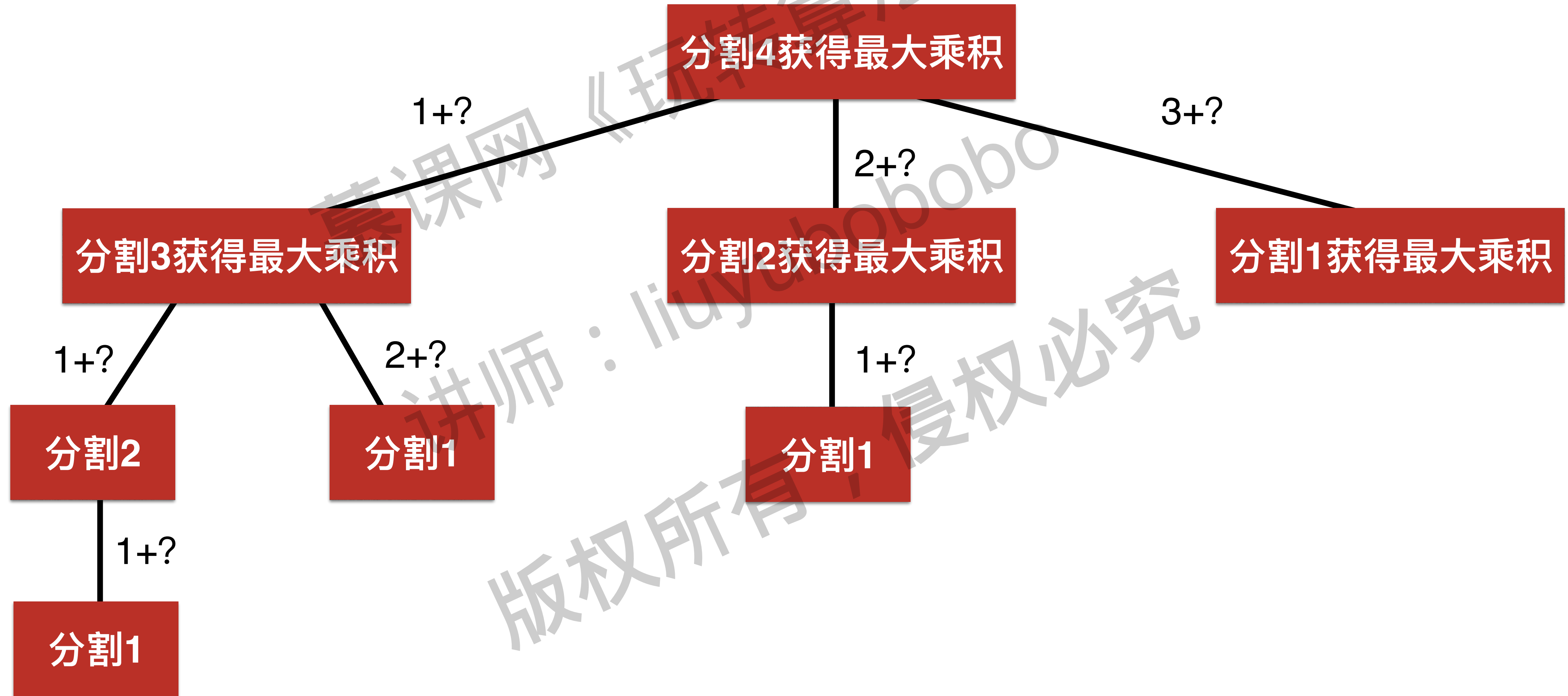
343. Integer Break



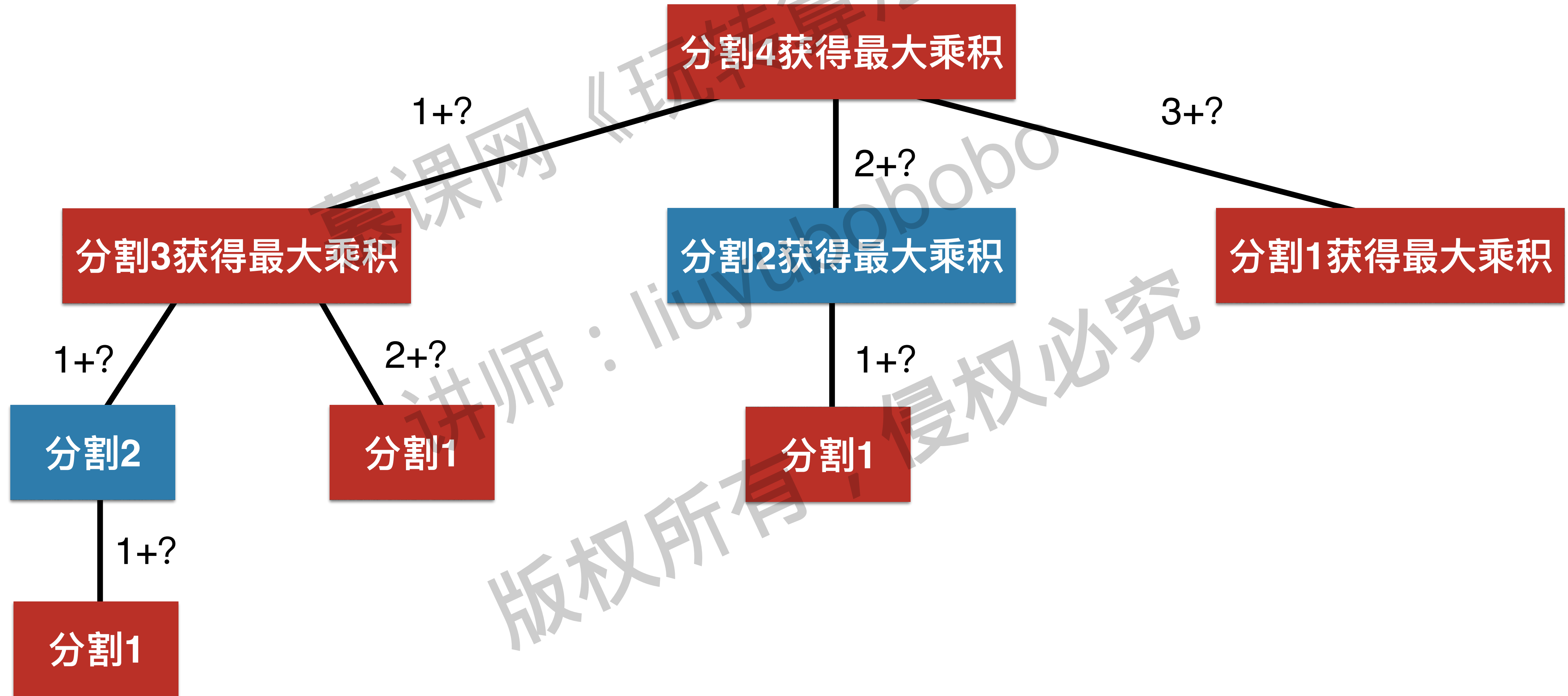
343. Integer Break



343. Integer Break



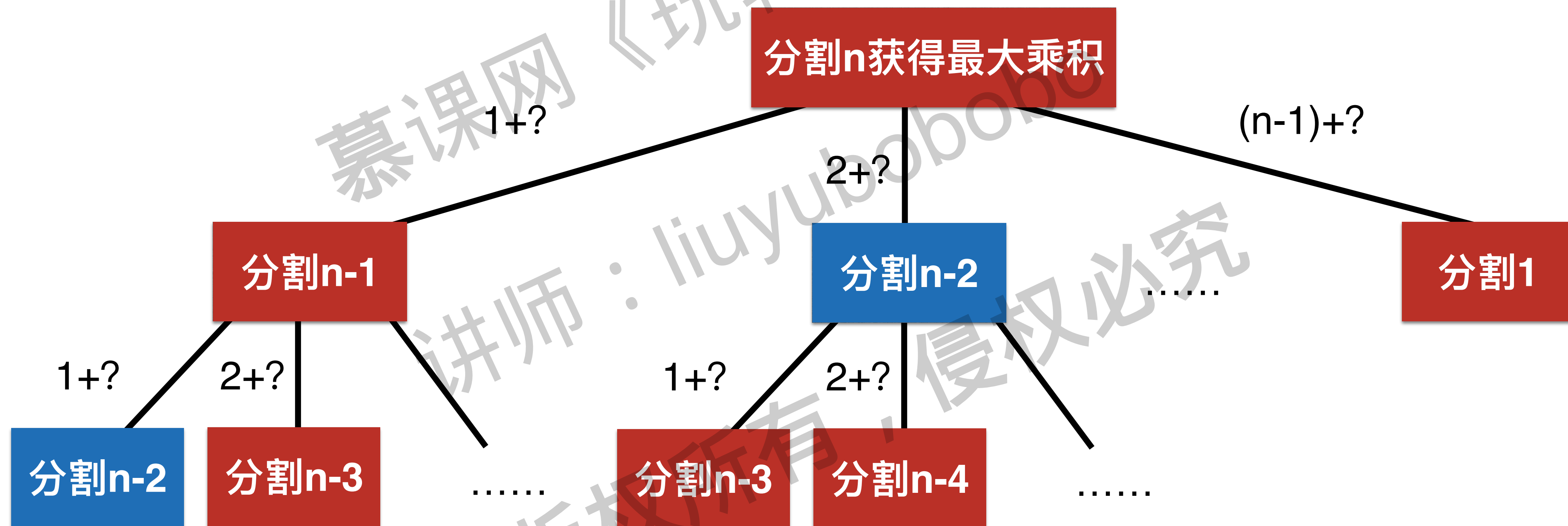
343. Integer Break



343. Integer Break



343. Integer Break



343. Integer Break



最优子结构

通过求子问题的最优解，可以获得原问题的最优解。

343. Integer Break

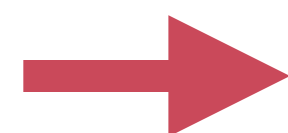


动态规划



动态规划

递归问题



重叠子问题
最优子结构



记忆化搜索

自顶向下的解决问题



动态规划

自底向上的解决问题

慕课网《玩转算法面试》

讲师: wuyubobo
版权所有, 侵权必究

实践：使用递归解决343

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

实践：使用记忆化搜索解决343

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

实践：使用动态规划解决343

慕课网《玩转算法面试》
讲师：luyibobobo

版权所有，侵权必究

慕课网《玩转算法面试》

时间复杂度分析

讲师：liuyubobobo

版权所有，侵权必究

279. Perfect Squares



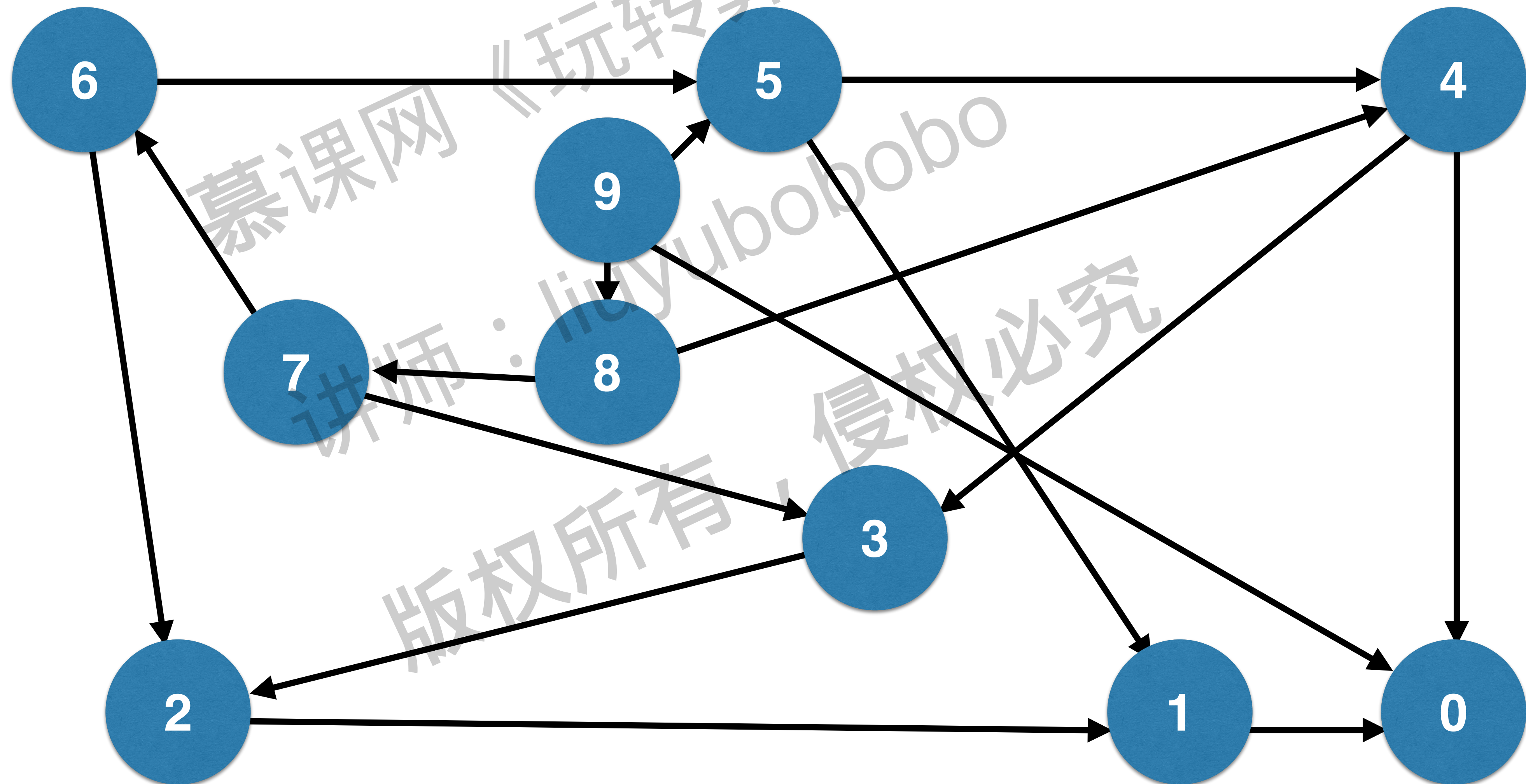
给出一个正整数 n ，寻找最少的完全平方数，使他们的和为 n 。

- 完全平方数：1, 4, 9, 16...

- $12 = 4 + 4 + 4$

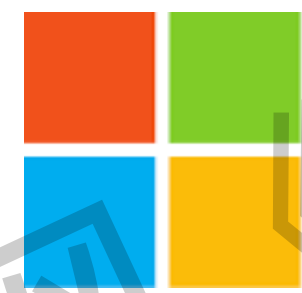
- $13 = 4 + 9$

279. Perfect Squares



91. Decode Ways

facebook



Microsoft



U B E R

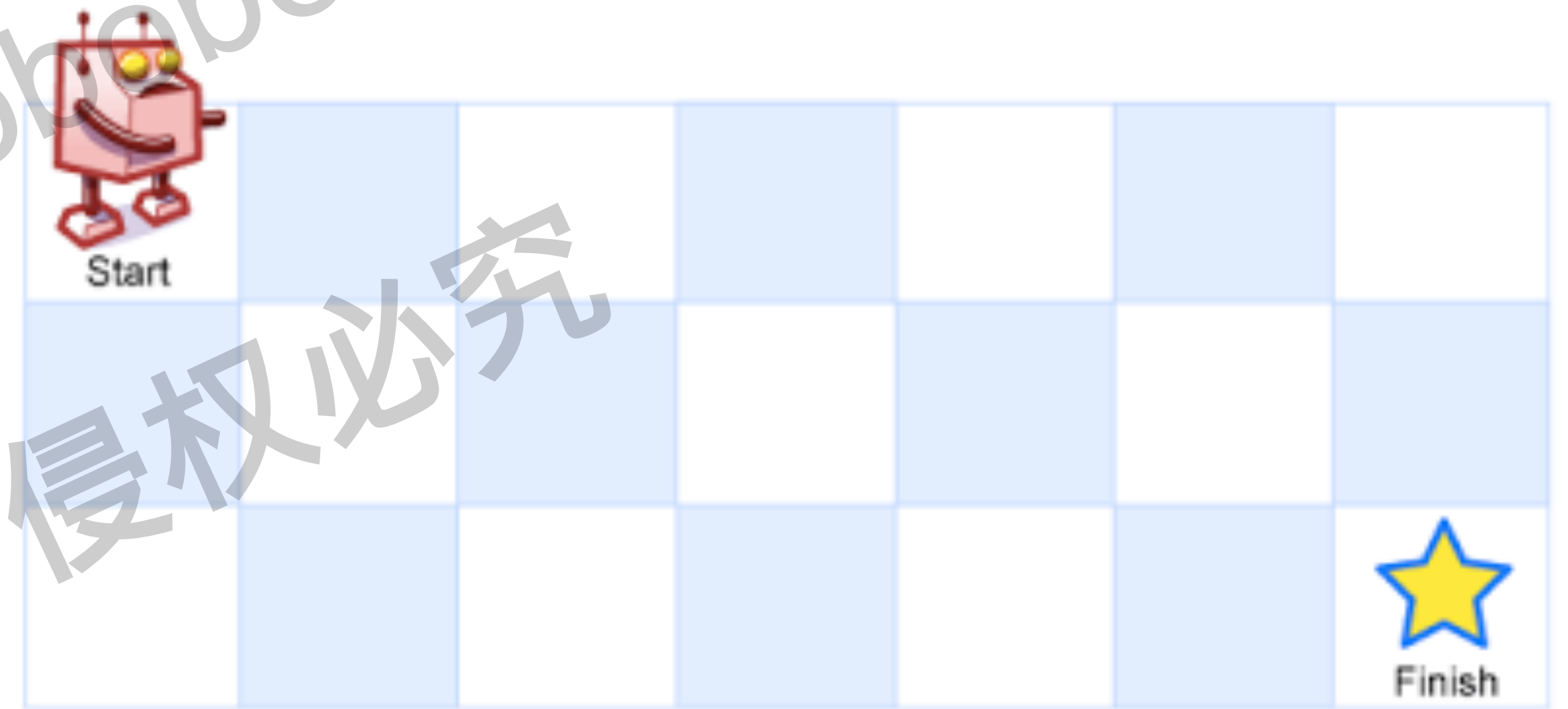
一个字符串，包含A-Z的字母。每一个字符可以和1-26的数字对应，如A->1； B->2； ...； Z->26 ...给出一个数字字符串，问我们有多少种方法可以解析这个数字字符串？

- 如给定12，我们可以将它解析成AB(1,2)或者L (12)
- 最终返回2

62. Unique Paths

Bloomberg

有一个机器人，从一个 $m \times n$ 的矩阵的左上角出发，要达到这个矩阵的右下角。机器人每次只能向右或者向下行进。问一共有多少种不同的路径？



63. Unique Paths II

Bloomberg

有一个机器人，从一个 $m \times n$ 的矩阵的左上角出发，矩阵中存在一些格子里有障碍物。现在机器人要达到这个矩阵的右下角。机器人每次只能向右或者向下行进。问一共有多少种不同的路径？

```
[  
  [0,0,0],  
  [0,1,0],  
  [0,0,0]  
]
```

答案为2

状态的定义 和 状态的转移

慕课网 《玩转算法面试》
讲师：lidyubobobo

版权所有，侵权必究

198. House Robber



你是一个专业的小偷，打算洗劫一条街的所有房子。每一个房子里都有不同价值的宝物，但是，如果你选择偷窃连续的两栋房子，就会触发报警系统。编程求出你最多可以偷窃价值多少的宝物？

- 如 $[3, 4, 1, 2]$ ，则返回 $6 [3, (4), 1, (2)]$
- 如 $[4, 3, 1, 2]$ ，则返回 $6 [(4), 3, 1, (2)]$

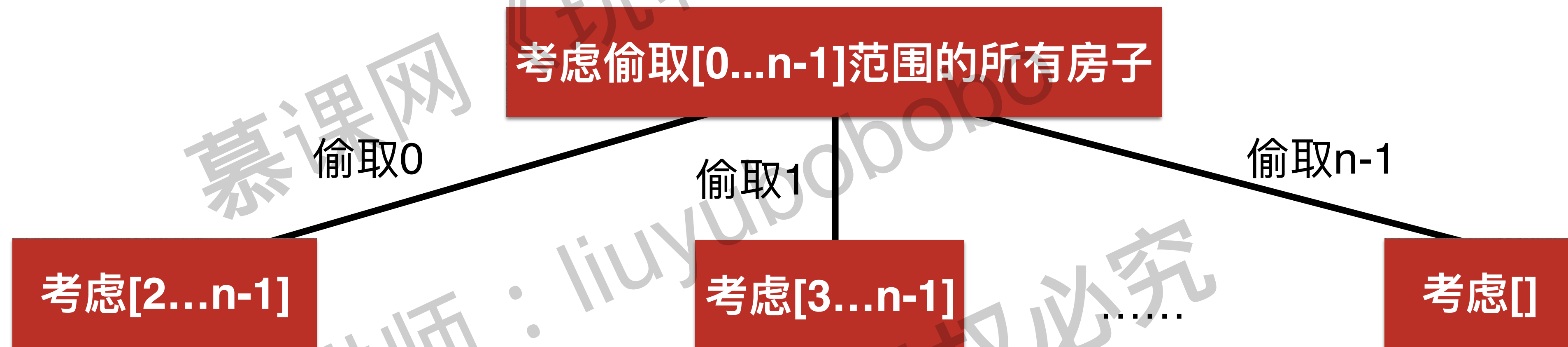
198. House Robber



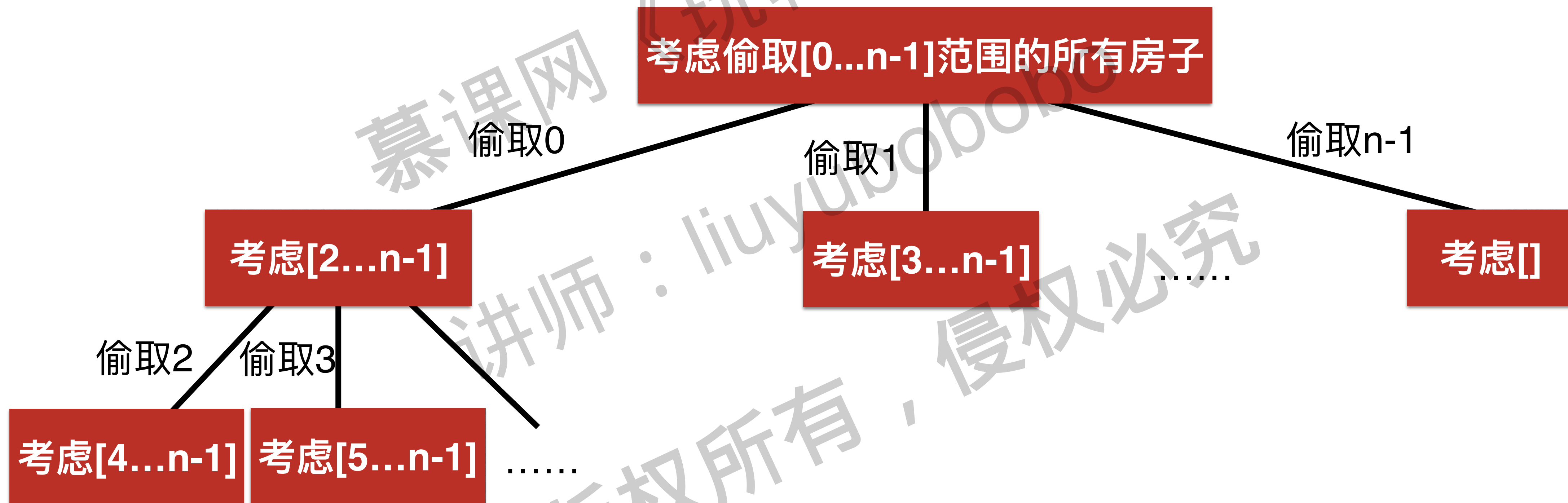
你是一个专业的小偷，打算洗劫一条街的所有房子。每一个房子里都有不同价值的宝物，但是，如果你选择偷窃连续的两栋房子，就会触发报警系统。编程求出你最多可以偷窃价值多少的宝物？

暴力解法：检查所有房子的组合，对每一个组合，检查是否有相邻的房子，如果没有，记录其价值。找最大值。 $O((2^n) * n)$

198. House Robber



198. House Robber



198. House Robber



198. House Robber



198. House Robber

注意其中对状态的定义：

考虑偷取 $[x \dots n-1]$ 范围里的房子 (函数的定义)

根据对状态的定义，决定状态的转移：

$$f(0) = \max\{ v(0) + f(2), v(1) + f(3), v(2) + f(4), \dots,$$

$$v(n-3) + f(n-1), v(n-2), v(n-1) \} \text{ (状态转移方程)}$$

实践：使用递归解决198

慕课网《玩转算法面试》

讲师：huayu99bobo

版权所有，侵权必究

实践：使用记忆化搜索解决198

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

实践：使用动态规划解决198

慕课网《玩转算法面试》
讲师：liuyubobobo

版权所有，侵权必究

慕课网《玩转算法面试》

时间复杂度分析

讲师：liuyubobobo

版权所有，侵权必究

198. House Robber

对状态的定义：

考虑偷取 $[x \dots n-1]$ 范围里的房子 (函数的定义)

改变对状态的定义：

考虑偷取 $[0 \dots x]$ 范围里的房子 (函数的定义)

练习：使用新的状态定义，完成问题

213. House Robber II



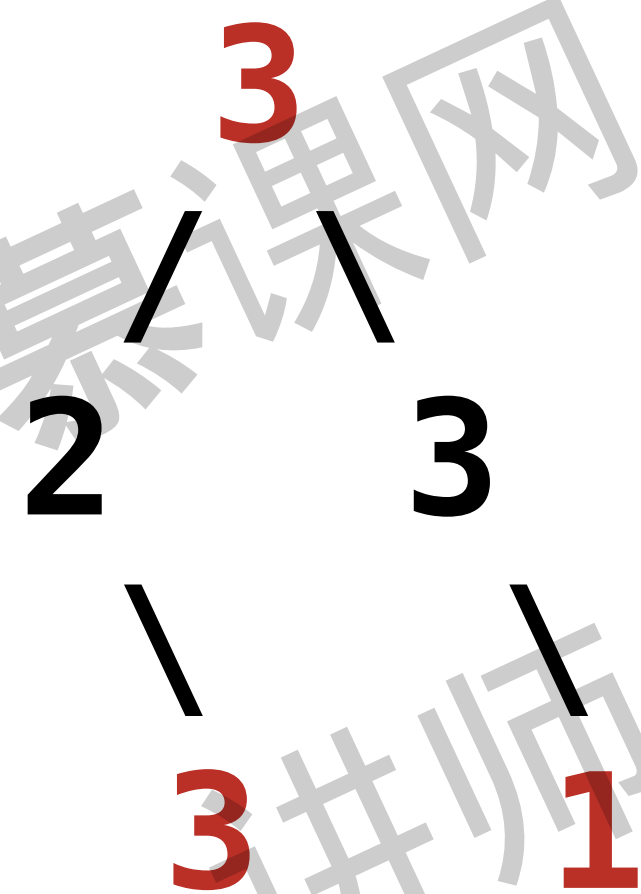
和 House Robber 一样，不过这次是在一个环形街道中。也就是说给定的数组中，最后一个元素和第一个元素为邻居。在不触碰警报的情况下，问能够窃取的财产的最大值是多少？

337. House Robber III

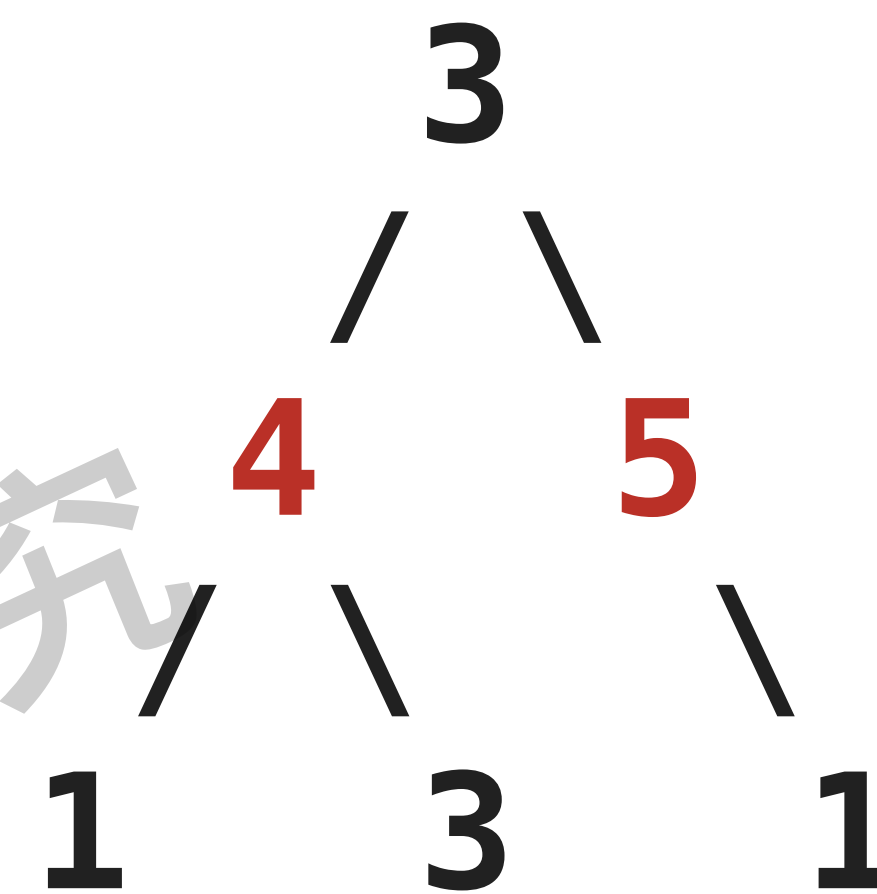
 U B E R

和 House Robber 一样，不过这次是在一个小区中，整个小区成二叉树的结构。在不触碰警报的情况下，问能够窃取的财产的最大值是多少？

337. House Robber III



最大值为 $3+3+1=7$



最大值为 $4+5=9$

309. Best Time to Buy and Sell Stock with Cooldown



给定一个数组，表示一支股票在每一天的价格。设计一个交易算法，在这些天进行自动交易，要求：每一天只能进行一次操作；在买完股票后，必须卖了股票，才能再次买入；每次卖了股票以后，在下一天是不能购买的。问如何交易，能让利润最大？

- 如 `prices = [1, 2, 3, 0, 2]`;
- 最佳交易方式: `[buy, sell, cooldown, buy, sell]` , 利润为3, 算法返回3

慕课网《玩转算法面试》

0-1背包问题

讲师：liuyubobobo

版权所有，侵权必究

0-1 背包问题

有一个背包，它的容量为C (Capacity)，。现在有n种不同的物品，编号为0...n-1，其中每一件物品的重量为w(i)，价值为v(i)。问可以向这个背包中盛放哪些物品，使得在不超过背包容量的基础上，物品的总价值最大。

暴力解法：每一件物品都可以放进背包，也可以不放进背包。

$O((2^n)*n)$

0-1 背包问题

贪心算法？ 优先放入平均价值最高的物品？

id	0	1	2
weight	1	2	3
value	6	10	12
v / w	6	5	4

有一个容量为5的背包



$$6 + 10 = 16$$

0-1 背包问题

贪心算法？ 优先放入平均价值最高的物品？

id	0	1	2
weight	1	2	3
value	6	10	12
v / w	6	5	4


有一个容量为5的背包



$$10 + 12 = 22$$

0-1 背包问题

$F(n, C)$ 考虑将 n 个物品放进容量为 C 的背包，使得价值最大

$$\begin{aligned} F(i, c) &= F(i-1, c) \\ &= v(i) + F(i-1, c - w(i)) \end{aligned}$$


max

$$F(i, c) = \max(F(i-1, c), v(i) + F(i-1, c - w(i)))$$

实践：使用递归解决0-1背包问题

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

实践：使用记忆化搜索解决0-1背包问题

慕课网《玩转算法面试》

讲师：luyubobo

版权所有，侵权必究

0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

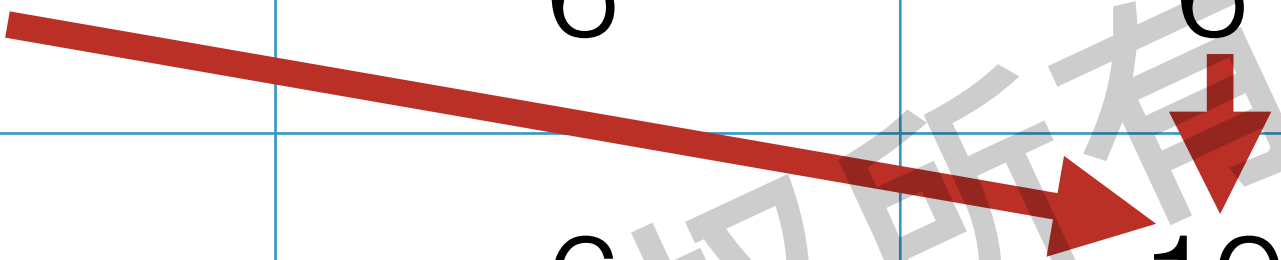
	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6				
2						

0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10			
2						



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16		
2						

0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

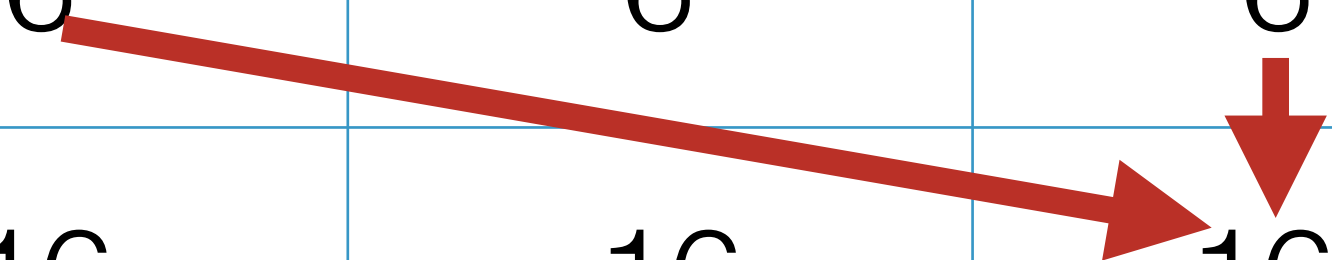
	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	
2						

0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2						



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0					

0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6				

0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

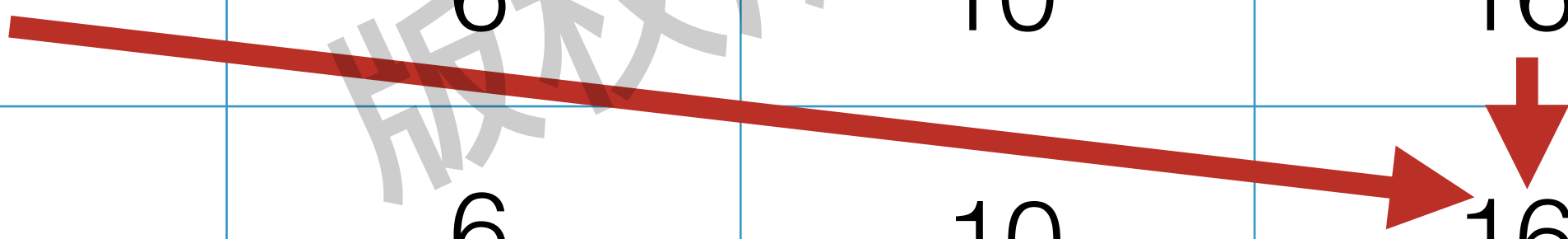
	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10			

0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10	16		

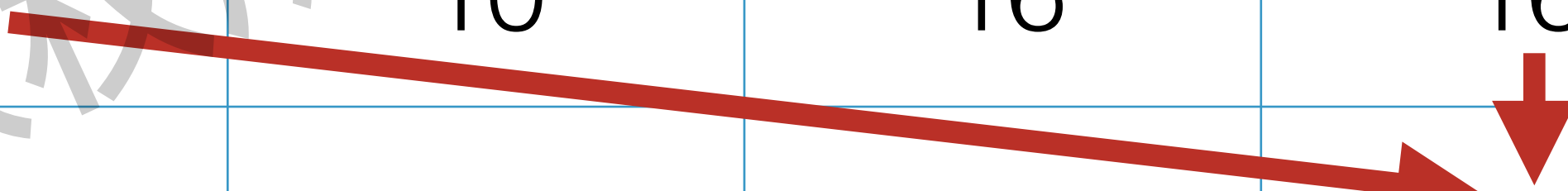


0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10	16	18	

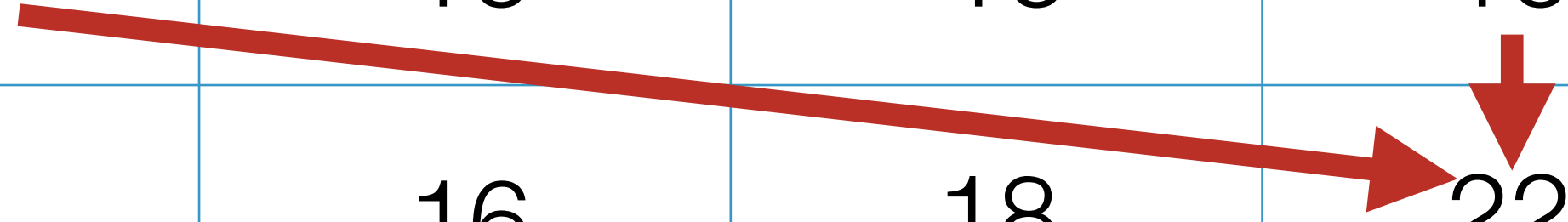


0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10	16	18	22



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10	16	18	22

实践：使用动态规划解决0-1背包问题

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

0-1 背包问题

时间复杂度: $O(n * C)$

空间复杂度: $O(n * C)$

0-1 背包问题的空间优化与变种

慕课网《玩转算法面试》
讲师：liuyubobobo
版权所有，侵权必究

慕课网《玩转算法面试》

0-1背包问题的优化

讲师：liuyubobobo

版权所有，侵权必究

0-1 背包问题

$F(n, C)$ 考虑将 n 个物品放进容量为 C 的背包，使得价值最大

$$F(i, c) = \max(F(i-1, c), v(i) + F(i-1, c - w(i)))$$

第 i 行元素只依赖于第 $i-1$ 行元素。理论上，只需要保持两行元素。

空间复杂度： $O(2 * C) = O(C)$

0-1 背包问题



慕课网

《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

0-1 背包问题

$i = 0$

0-1 背包问题

$i = 0$

$i = 1$

0-1 背包问题

$i = 2$

$i = 1$

0-1 背包问题

$i = 2$

$i = 3$

0-1 背包问题

$i = 4$

$i = 3$

0-1 背包问题

$i = 4$ (偶数)

$i = 3$ (奇数)

实践：0-1 背包问题动态规划空间优化1

0-1 背包问题

第*i*行元素只依赖于第*i*-1行元素。理论上，只需要保持两行元素。

空间复杂度： $O(2 * C) = O(C)$

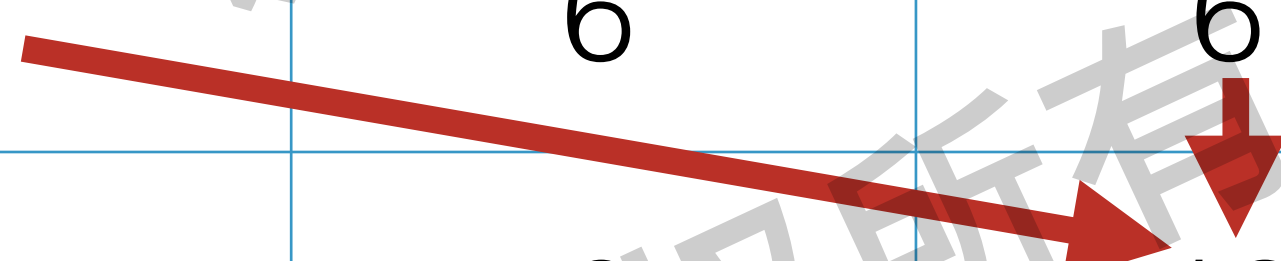
只使用一行大小为C的数组完成动态规划？

0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10			



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

0	1	2	3	4	5
0	6	6	6	6	6



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

0	1	2	3	4	5
0	6	6	6	6	16



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

0	1	2	3	4	5
0	6	6	6	6	16



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

0	1	2	3	4	5
0	6	6	6	16	16



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

0	1	2	3	4	5
0	6	6	6	16	16



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

0	1	2	3	4	5
0	6	6	16	16	16



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

0	1	2	3	4	5
0	6	6	16	16	16



0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

0	1	2	3	4	5
0	6	10	16	16	16



实践：0-1 背包问题动态规划空间优化2

慕课网《玩转算法面试》

0-1 背包问题的变种

讲师：liuyubobobo

版权所有，侵权必究

0-1 背包问题更多变种

多重背包问题：每个物品不止1个，有 $\text{num}(i)$ 个。

完全背包问题：每个物品可以无限使用。

0-1 背包问题更多变种

多维费用背包问题：要考虑物品的体积和重量两个维度？

0-1 背包问题更多变种

物品间加入更多约束

物品间可以互相排斥；也可以互相依赖

慕课网《玩转算法面试》

面试中的0-1背包问题

讲师：lidyubobobo

版权所有，侵权必究

416. Partition Equal Subset Sum



给定一个非空数组，其中所有的数字都是正整数。问是否可以将这个数组的元素分成两部分，使得每部分的数字和相等？

- 如对 $[1, 5, 11, 5]$ ，可以分成 $[1, 5, 5]$ 和 $[11]$ 两部分，元素和相等，返回true
- 如对 $[1, 2, 3, 5]$ ，无法分成元素和相等的两部分，返回false

416. Partition Equal Subset Sum

典型的背包问题，在n个物品中选出一定物品，填满sum/2的背包

$F(n, C)$ 考虑将n个物品填满容量为C的背包

$$F(i, c) = F(i-1, c) \parallel F(i-1, c - w(i))$$

时间复杂度： $O(n * \text{sum}/2) = O(n * \text{sum})$

416. Partition Equal Subset Sum



给定一个非空数组，其中所有的数字都是正整数。问是否可以将这个数组的元素分成两部分，使得每部分的数字和相等？

- 最多有200个数字
- 每个数字最大为100
- 所有数字和为20000；背包最大为10000； $n * \text{sum} / 2 = 100 * 10000 = 100\text{万}$

实践：使用递归解决416

慕课网《玩转算法面试》
讲师：huayu99bobo

版权所有，侵权必究

实践：使用记忆化搜索解决416

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

实践：使用动态规划解决416

慕课网《玩转算法面试》
讲师：luyibobobo

版权所有，侵权必究

322. Coin Change

给定不同面值的硬币。问最少需要多少枚硬币，可以凑成指定的金额？算法返回这个数。如果无法凑成指定金额，则返回-1。（可以无限次使用同种面额的硬币。）

- 如给定硬币的面额为 [1, 2, 5], $\text{amount} = 11$ ，则返回3 （ $5+5+1=11$ ）
- 如给定硬币的面额为 [2], $\text{amount} = 3$ ，则返回-1

377. Combination Sum IV

facebook

Google



snapchat

给定一个整数数组，其中元素没有重复。问有多少种可能，使用这个数组中的数字，凑出一个指定的整数target

- 如 $\text{nums} = [1, 2, 3]$, $\text{target} = 4$
- 可能的组合有 $[1, 1, 1, 1]$, $[1, 1, 2]$, $[1, 2, 1]$, $[1, 3]$, $[2, 1, 1]$, $[2, 2]$, $[3, 1]$
- 算法返回7
- 注意：顺序性

474. Ones and Zeroes



给定一个字符串数组。数组中的每个字符串都是一个01串。问用m个0和n个1，最多可以组成数组中的多少个01串？

- 如[10, 0001, 111001, 1, 0],
- 给定5个0和3个1，最多可以组成其中的4个元素：10, 0001, 1, 0
- 如[10, 0, 1], 给定1个0和1个1，最多可以组成其中2两个元素，0和1

474. Ones and Zeroes



给定一个字符串数组。数组中的每个字符串都是一个01串。问用m个0和n个1，最多可以组成数组中的多少个01串？

- m和n不超过100
- 数组中的元素个数不超过600
- 注意：什么叫组成？必须用完？还是有剩余就可以？

139. Word Break

Google

facebook

amazon

UBER

Bloomberg

Pocket Gems

YAHOO!

给定一个非空字符串s和一个字符串数组wordDict，问能否使用wordDict中的不同字符串首尾连接，组成s。假定wordDict中没有重复的字符串。

- 如 s = "leetcode"，dict = ["leet", "code"]，则返回true

494. Target Sum



facebook

给定一个非0数字序列，在这些数字前加上+或者-的符号，使其计算结果为给定的整数S。问一共有多少种可能。

- 如：nums = [1, 1, 1, 1, 1], S = 3
- 答案为5

$$-1+1+1+1+1 = 3$$

$$+1-1+1+1+1 = 3$$

$$+1+1-1+1+1 = 3$$

$$+1+1+1-1+1 = 3$$

$$+1+1+1+1-1 = 3$$

慕课网《玩转算法面试》

最长上升子序列

讲师：liuyubobobo

版权所有，侵权必究

300. Longest Increasing Subsequence



给定一个整数序列，求其中的最长上升子序列的长度。

- 如 [10, 9, 2, 5, 3, 7, 101, 18]，其最长上升子序列的长度为4。
- 最长上升子序列为 [2, 5, 7, 101]
- 注意1：什么是子序列？
- 注意2：什么是上升？
- 注意3：一个序列可能有多个最长上升子序列；但这个最长的长度只有1个。

300. Longest Increasing Subsequence



给定一个整数序列，求其中的最长上升子序列的长度。

- 如 [10, 9, 2, 5, 3, 7, 101, 18]，其最长上升子序列的长度为4。

暴力解法：选择所有的子序列进行判断。 $O(2^n * n)$

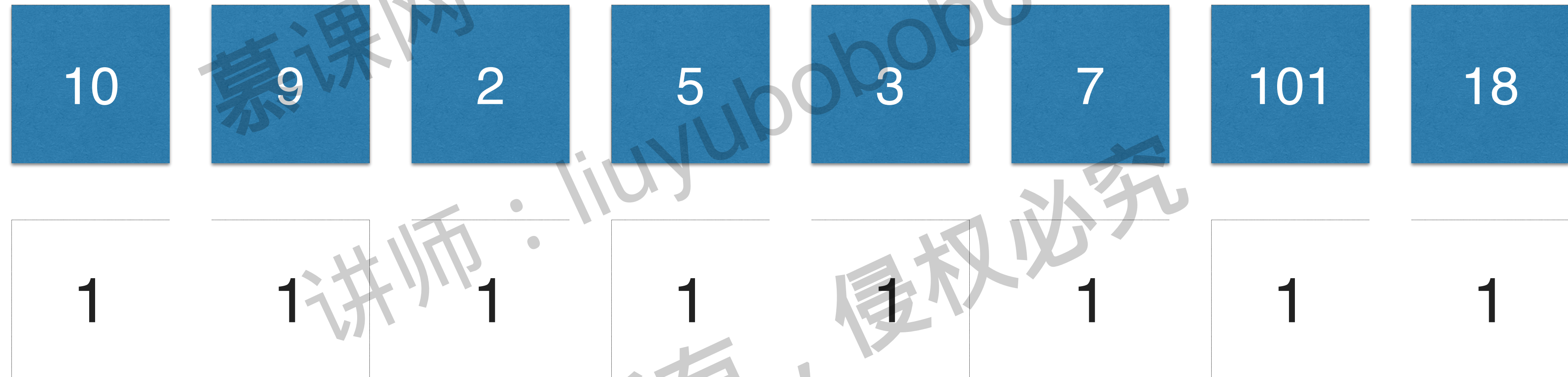
300. Longest Increasing Subsequence

LIS(i) 表示以第 i 个数字为结尾的最长上升子序列的长度

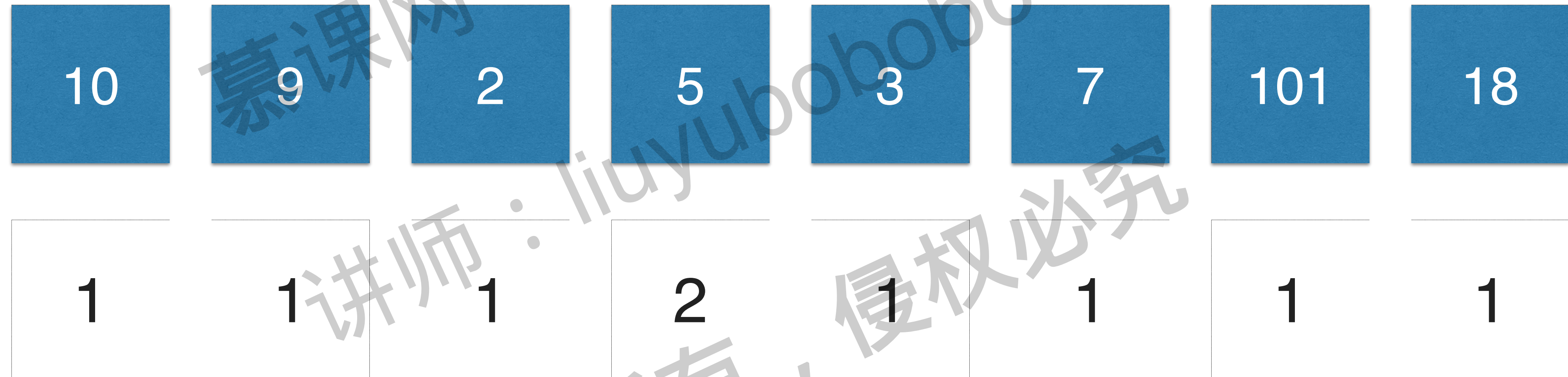
LIS(i) 表示 [0...i] 的范围内，选择数字nums[i]可以获得的最长上升子序列的长度

$$\text{LIS}(i) = \max_{j < i} (1 + \text{LIS}(j) \text{ if } \text{nums}[i] > \text{nums}[j])$$

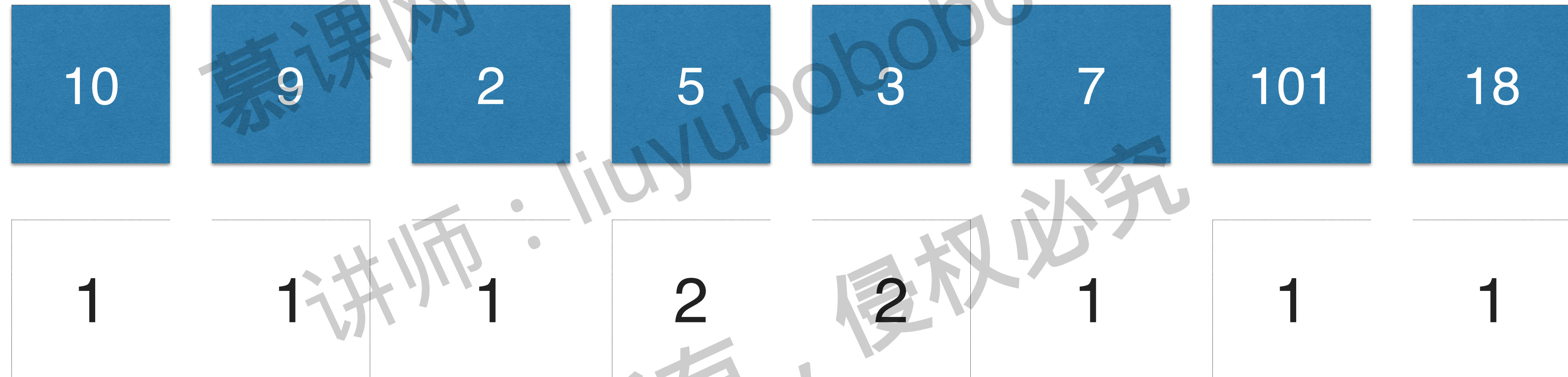
300. Longest Increasing Subsequence



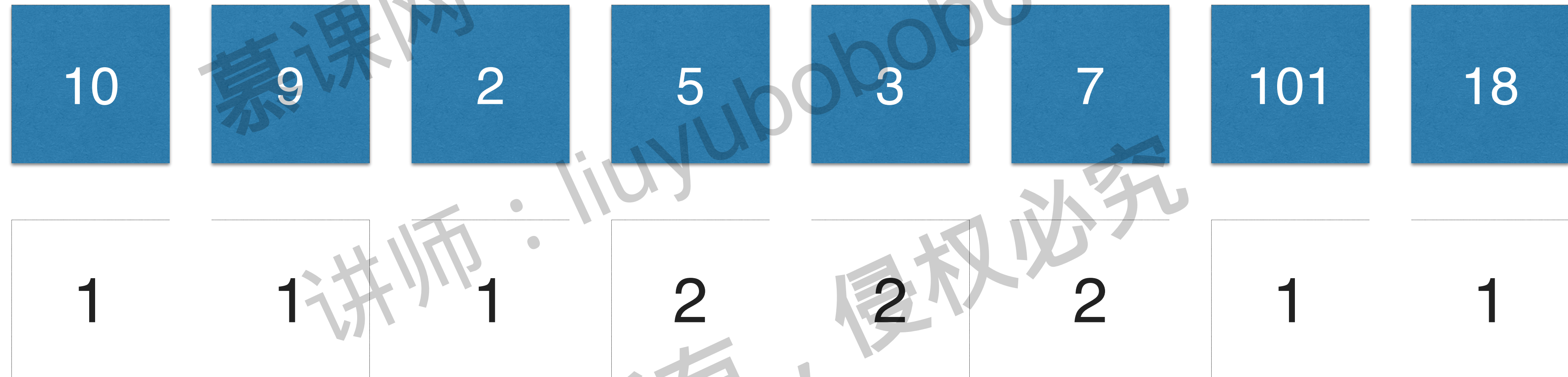
300. Longest Increasing Subsequence



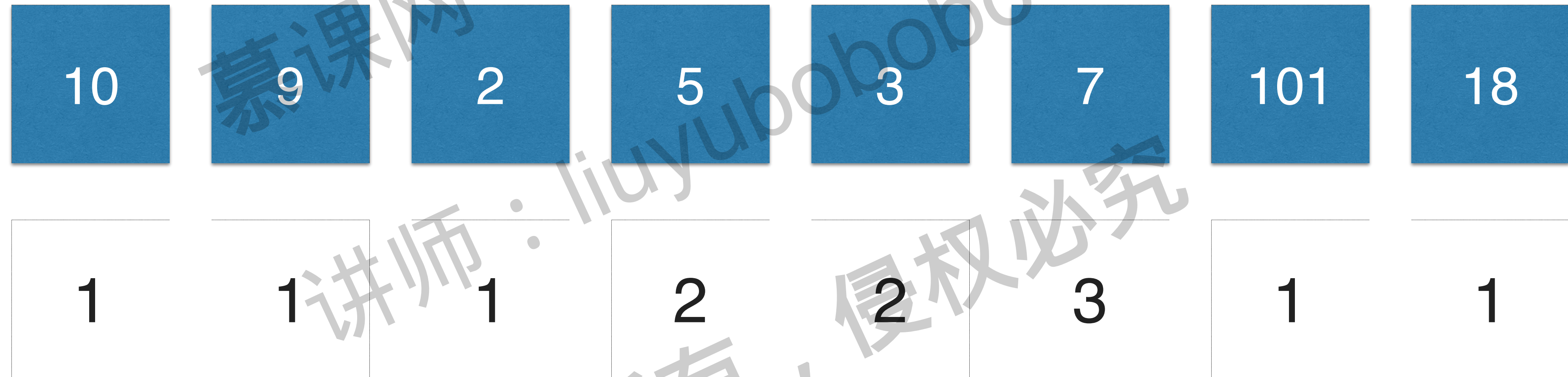
300. Longest Increasing Subsequence



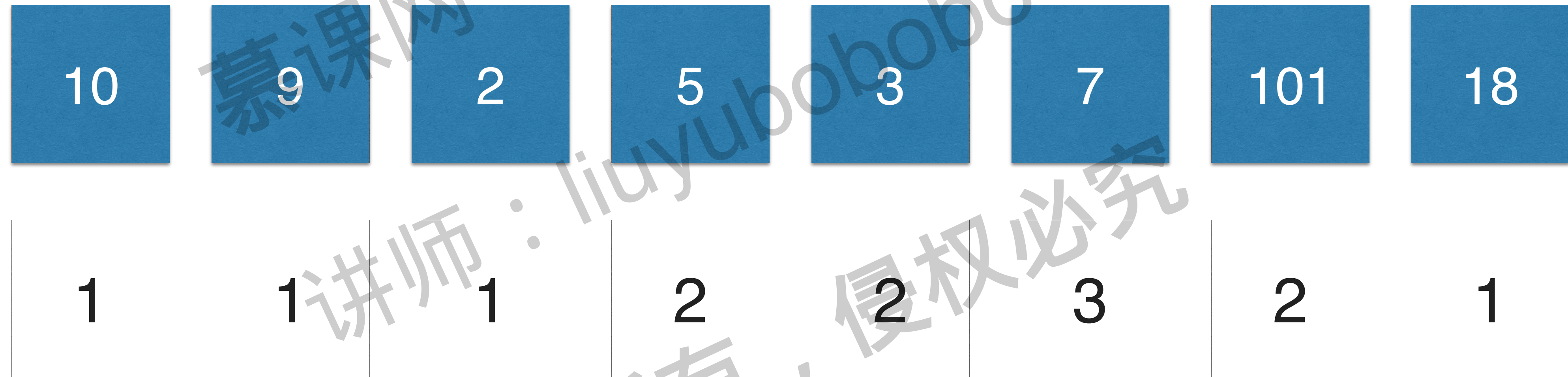
300. Longest Increasing Subsequence



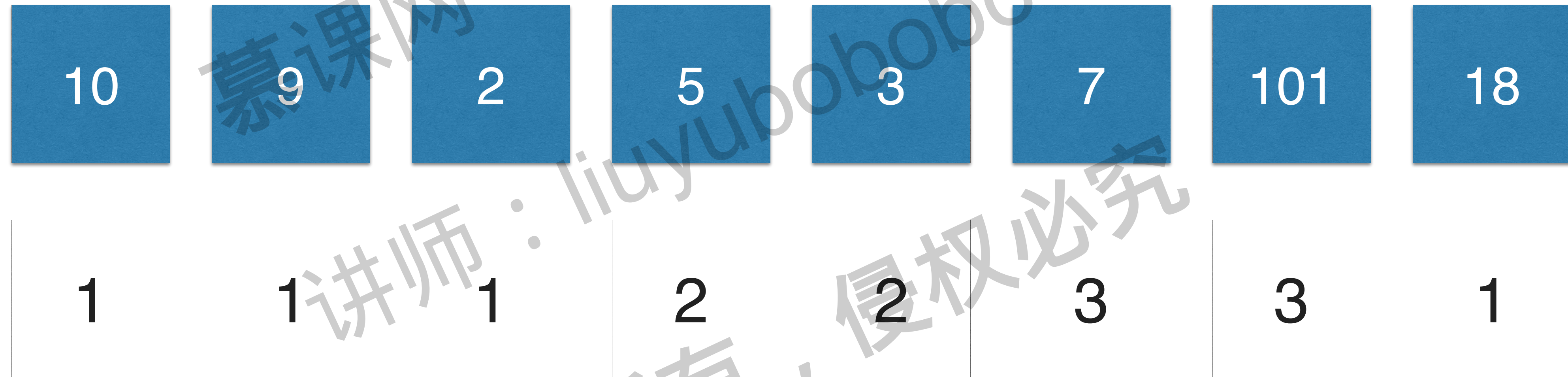
300. Longest Increasing Subsequence



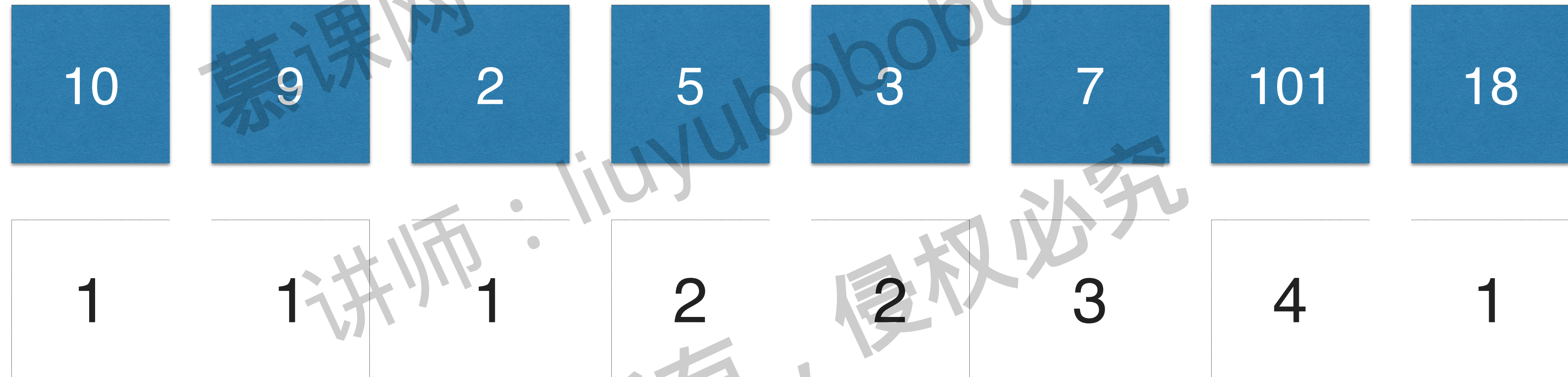
300. Longest Increasing Subsequence



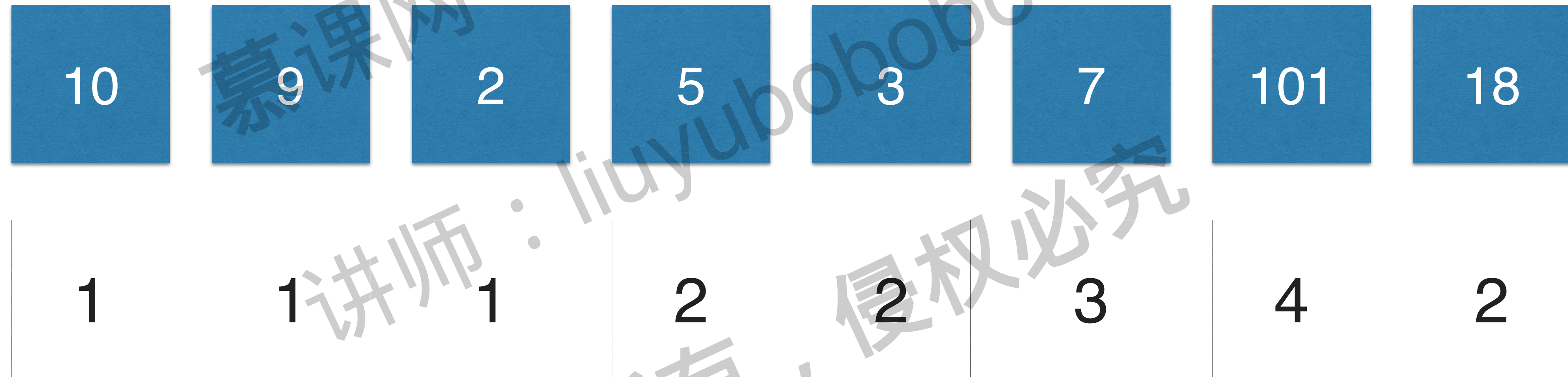
300. Longest Increasing Subsequence



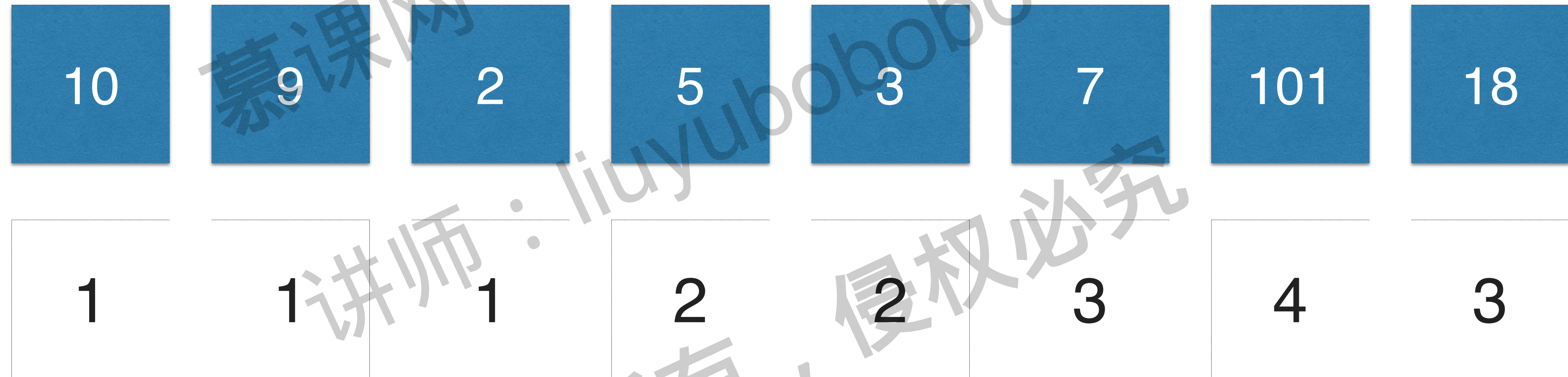
300. Longest Increasing Subsequence



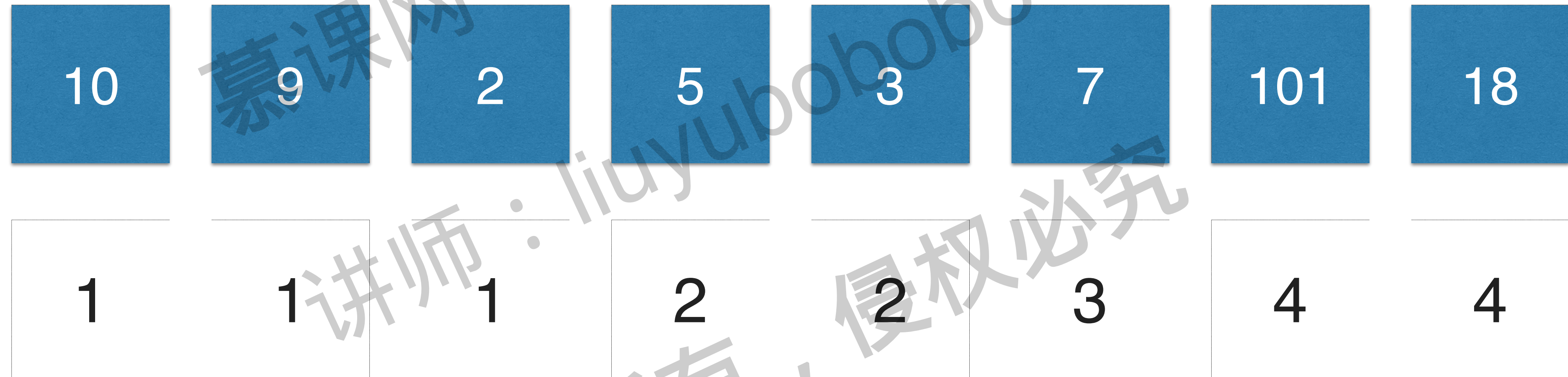
300. Longest Increasing Subsequence



300. Longest Increasing Subsequence



300. Longest Increasing Subsequence



慕课网《玩转算法面试》

注意

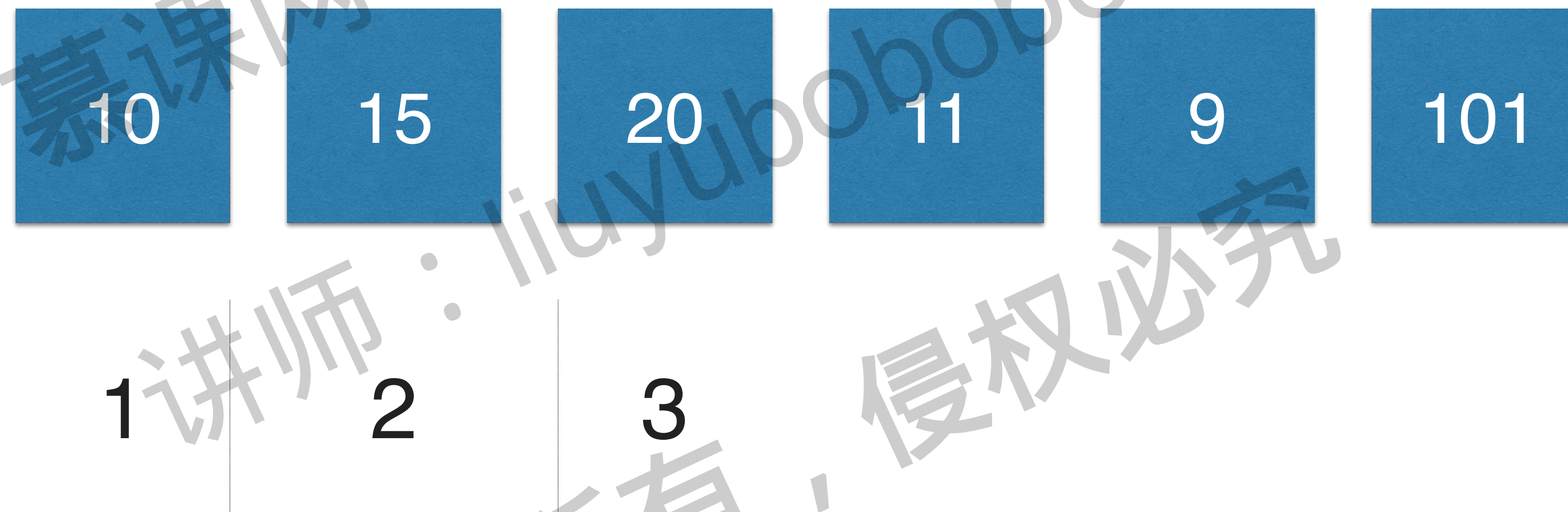
讲师：liuyubobobo

版权所有，侵权必究

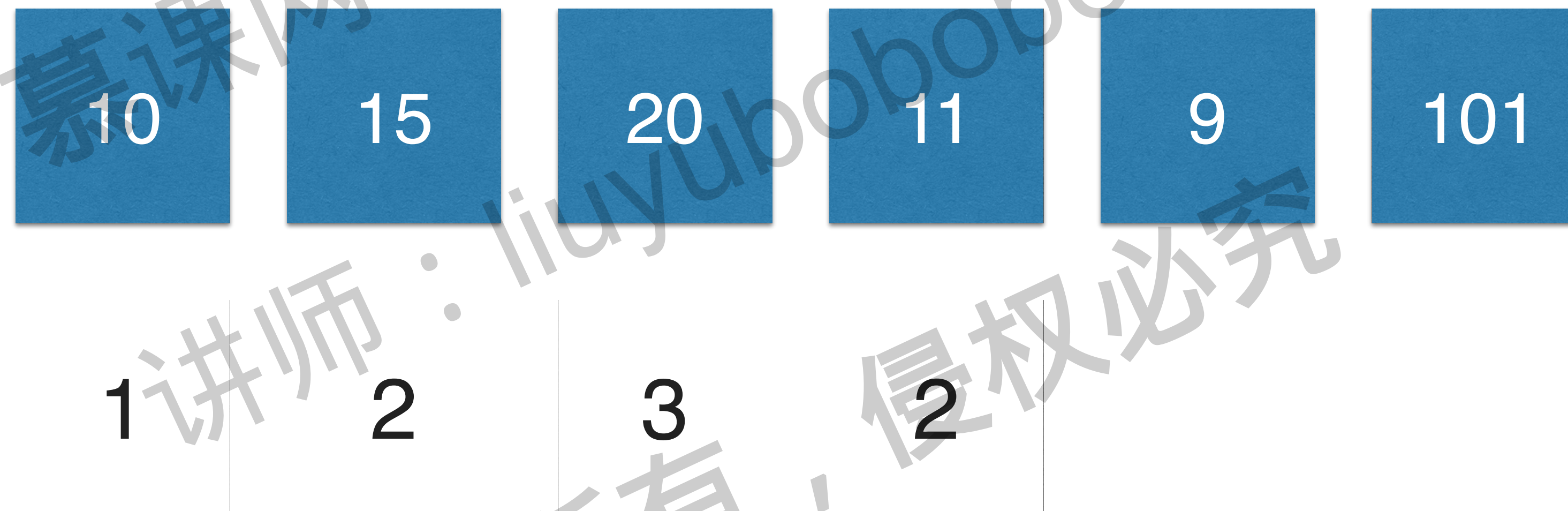
300. Longest Increasing Subsequence



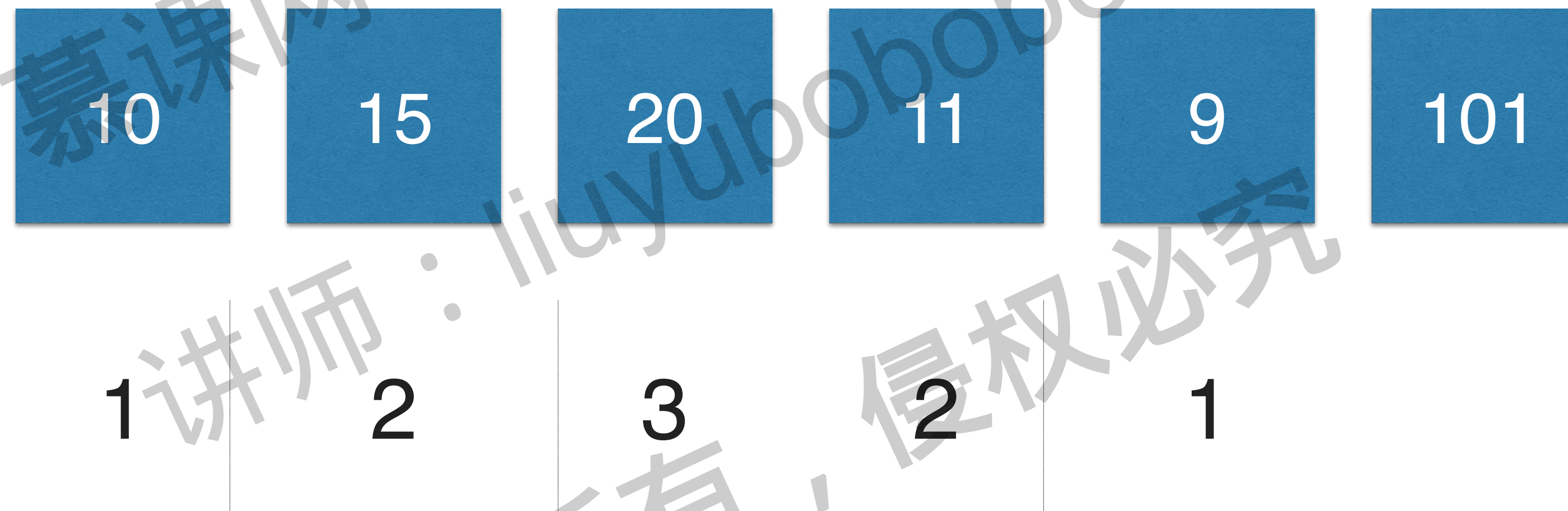
300. Longest Increasing Subsequence



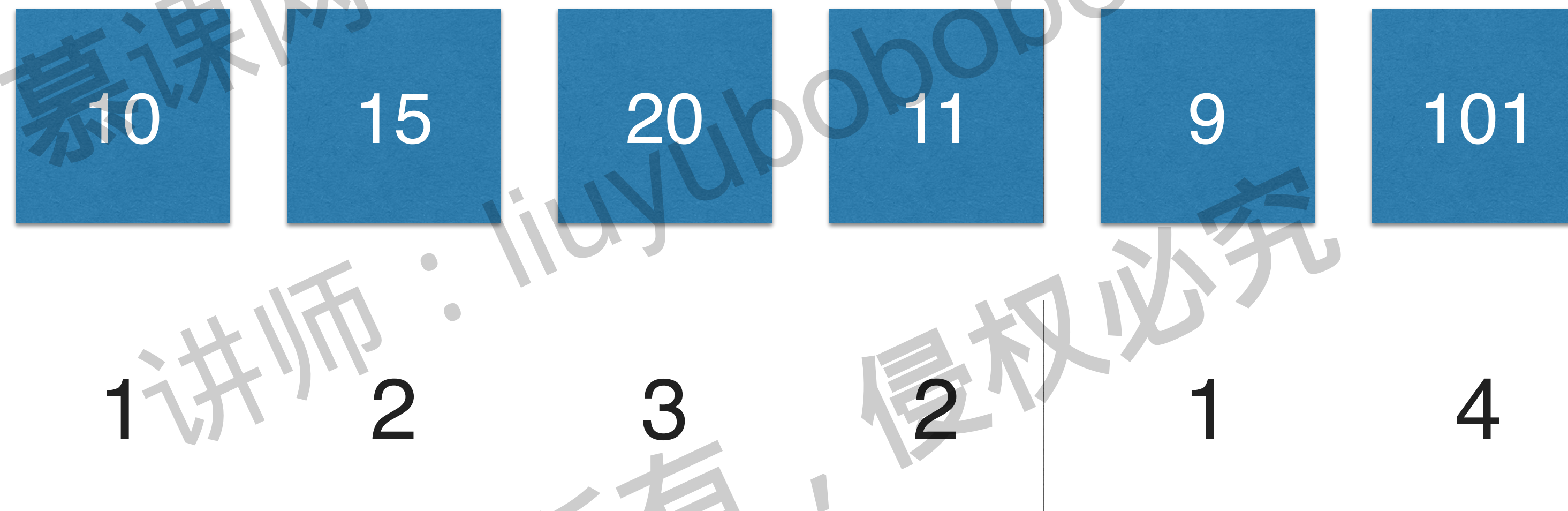
300. Longest Increasing Subsequence



300. Longest Increasing Subsequence



300. Longest Increasing Subsequence



实践：动态规划实现 300

讲师：lucyubobobo

版权所有，侵权必究

LIS问题的 $O(n\log n)$ 解法

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

376. Wiggle Subsequence

一个序列，它的相邻数字的大小关系是升序降序轮流交替的（最初可以是升序，也可以是降序），就称为wiggle sequence。比如 [1, 7, 4, 9, 2, 5] 就是一个wiggle sequence。但是 [1, 4, 7, 2, 5] 和 [1, 7, 4, 5, 5] 就不是。给出一个数组，求出他的最长 wiggle sequence 子序列。

慕课网《玩转算法面试》

更多关于动态规划

讲师：liuyubobobo

版权所有，侵权必究

最长公共子序列

Longest Common Sequence (LCS)

最长公共子序列

给出两个字符串S1和S2，求这两个字符串的最长公共子序列的长度

S1 = AAACCGTGAGTTATTCGTTCTAGAA

S2 = CACCCCTAAGGTACCTTTGGTTC

S1 = ABCD

S2 = AEBD

最长公共子序列

LCS(m , n) S1[0...m] 和 S2[0...n] 的最长公共子序列的长度

$S1[m] == S2[n] :$

$LCS(m, n) = 1 + LCS(m-1, n-1)$

$S1[m] != S2[n] :$

$LCS(m, n) = \max(LCS(m-1, n) , LCS(m, n-1))$

最长公共子序列

ABCD | AEBD

慕课网

《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

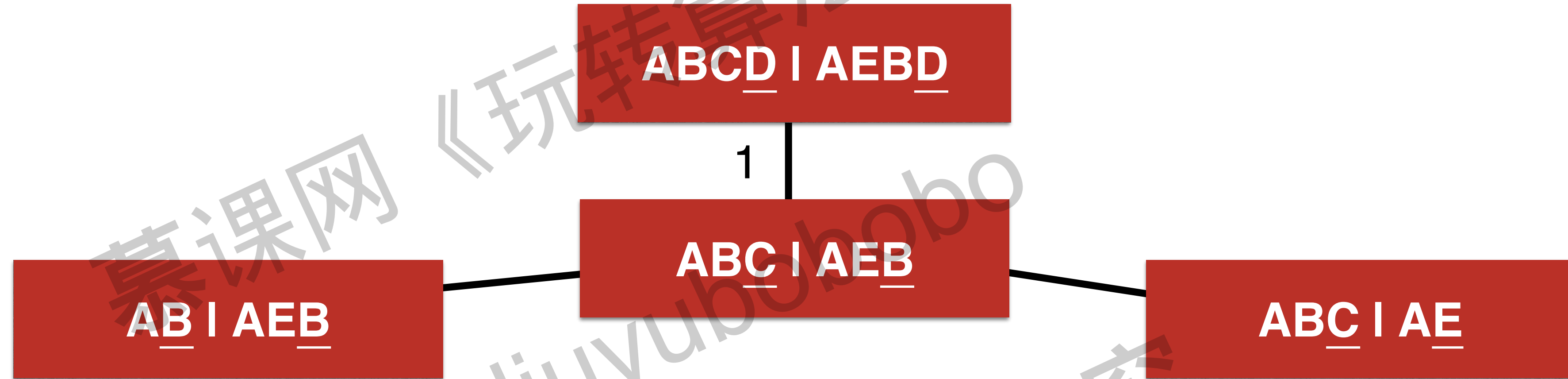
最长公共子序列

ABCD | AEBD

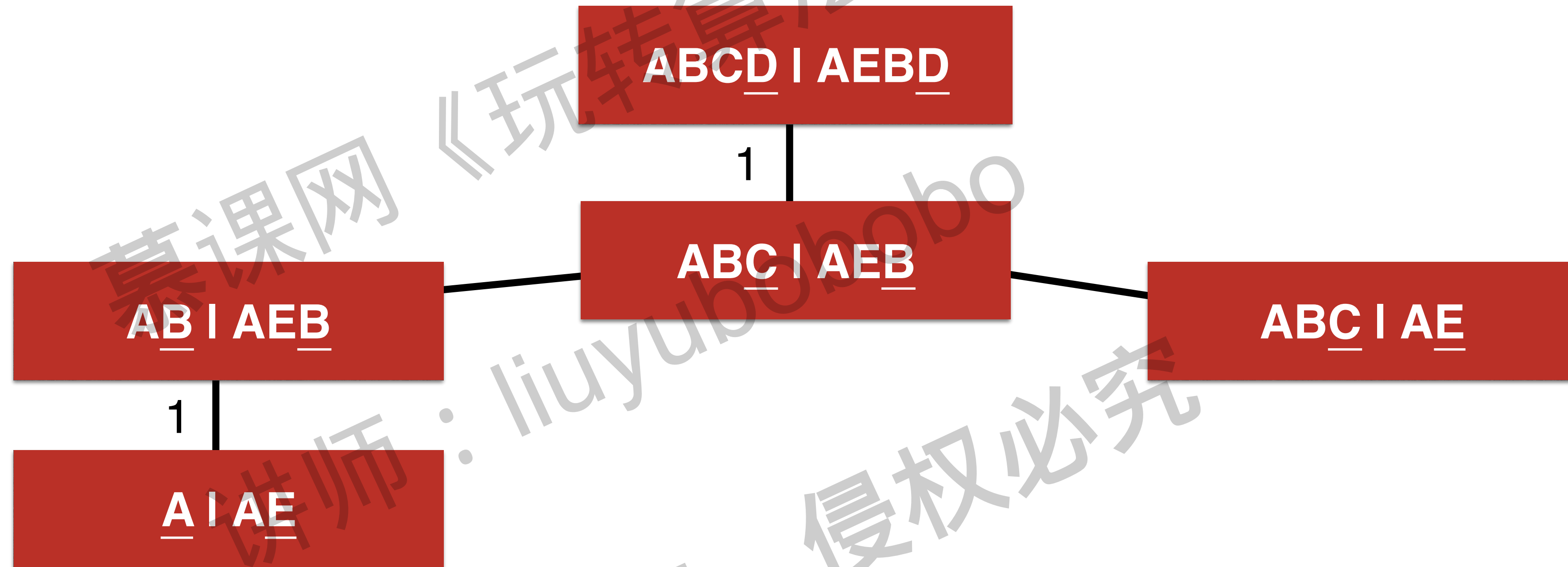
1

ABC | AEB

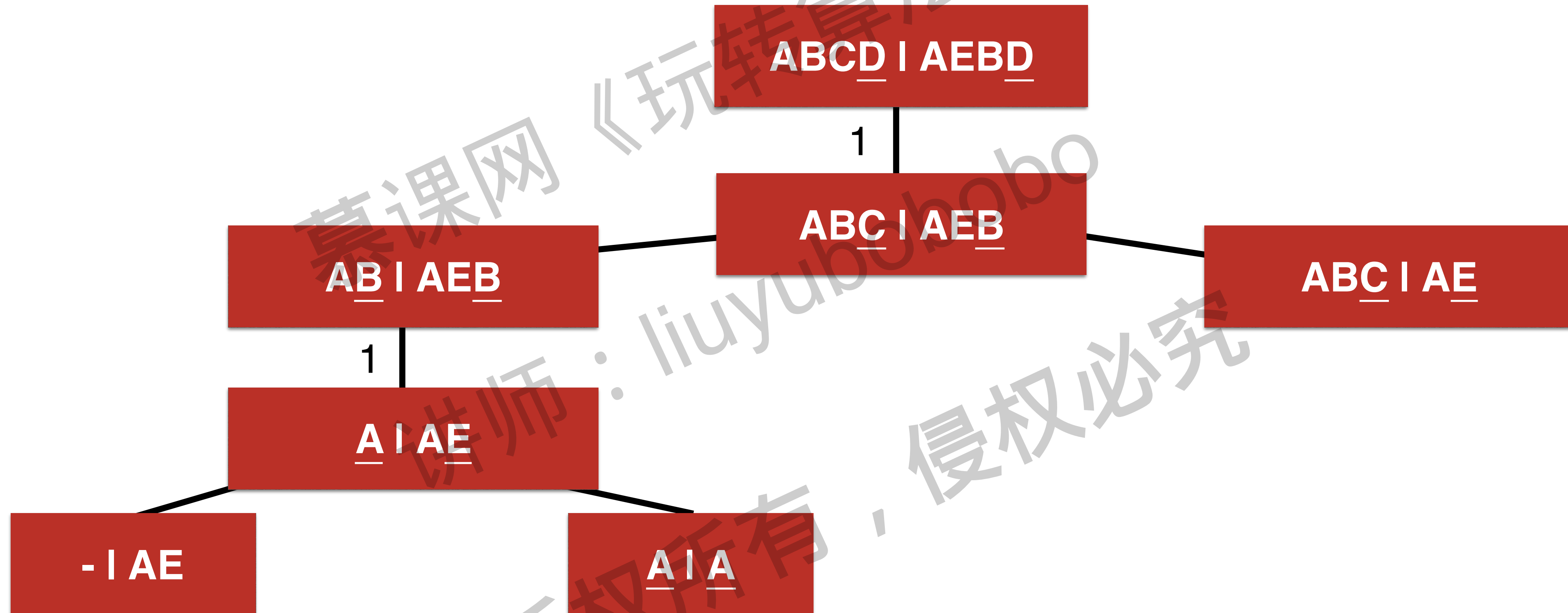
最长公共子序列



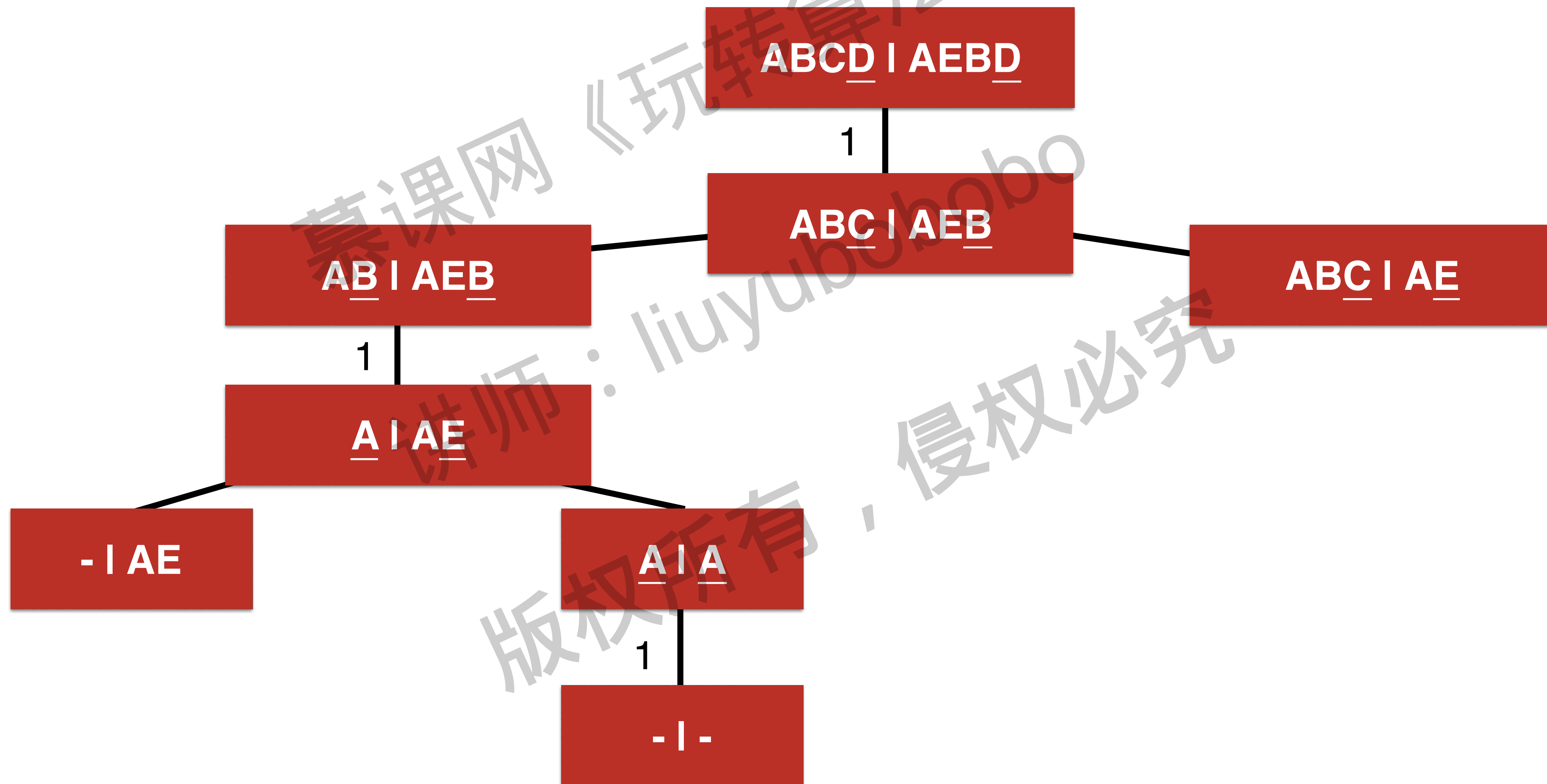
最长公共子序列



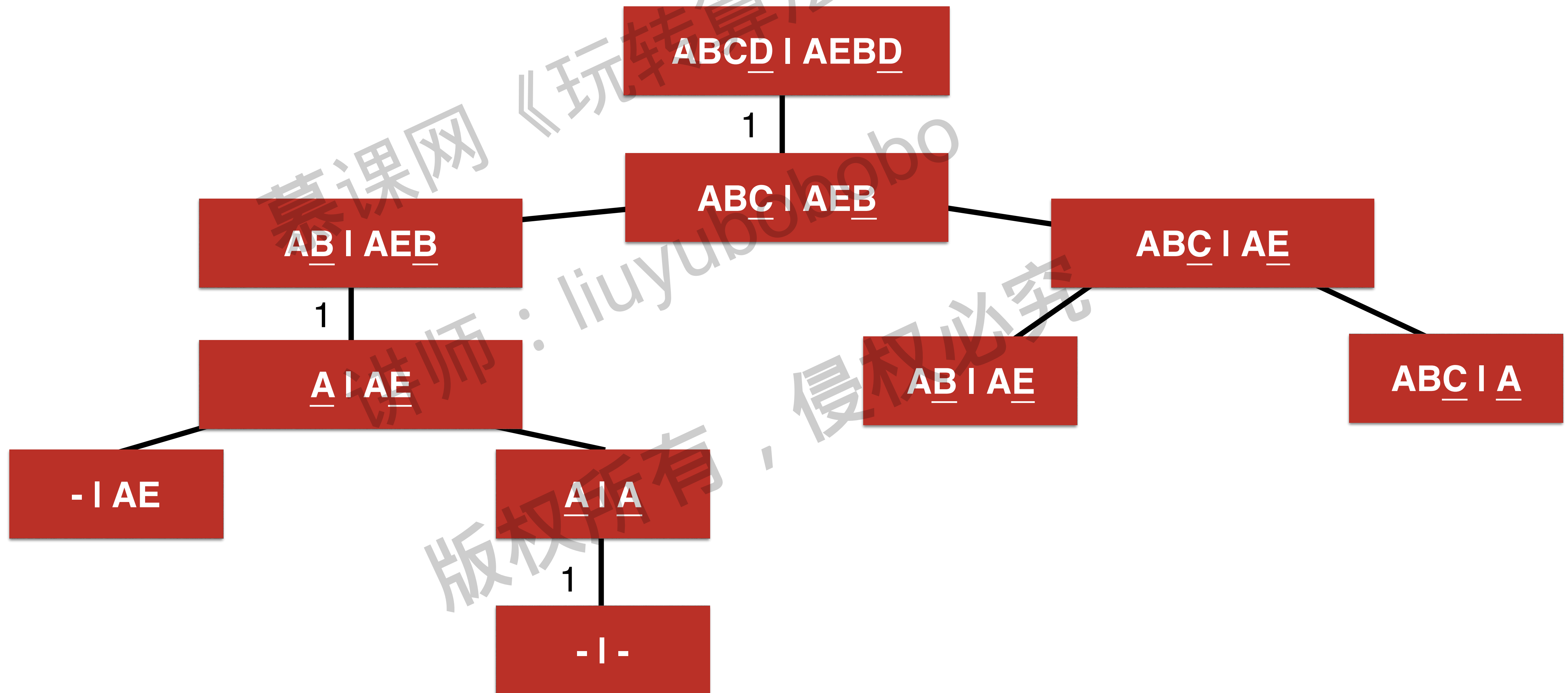
最长公共子序列



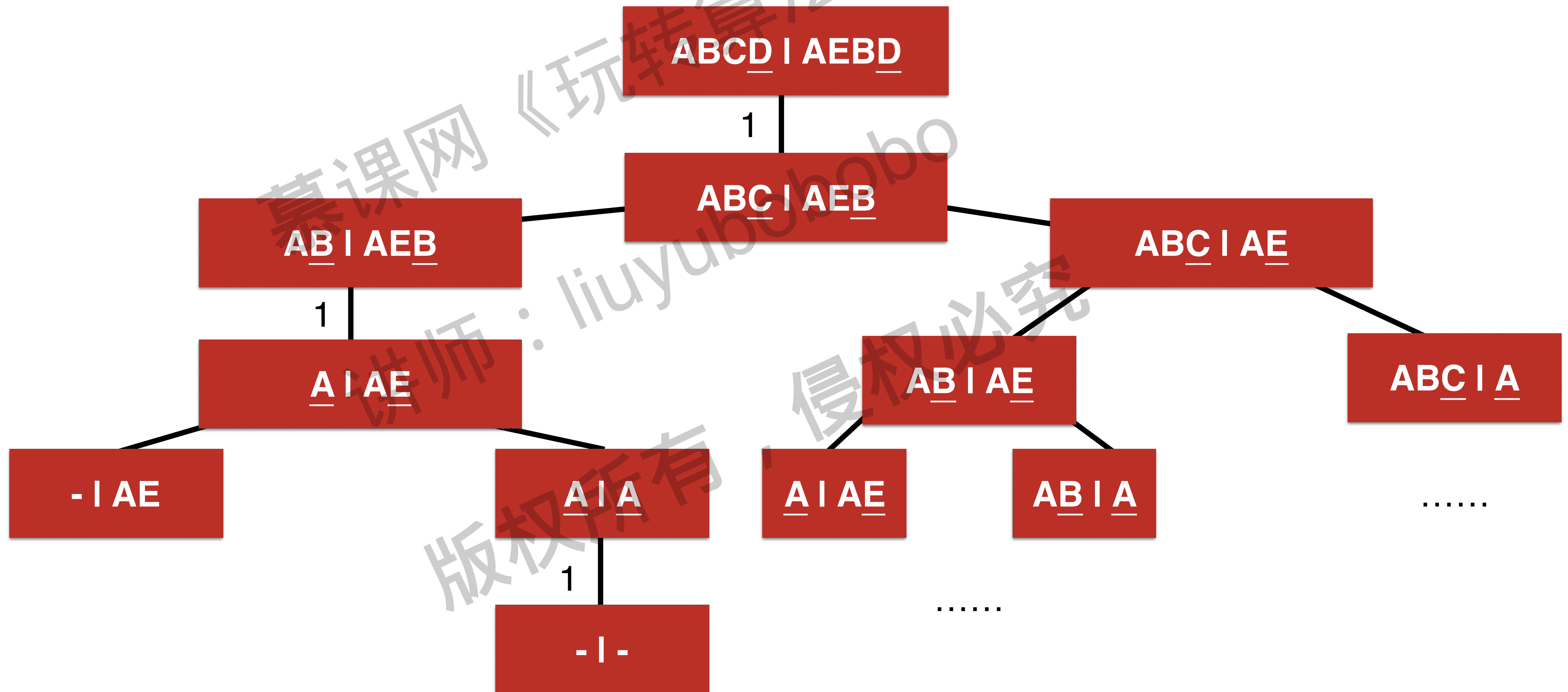
最长公共子序列



最长公共子序列



最长公共子序列



慕课网《玩转算法面试》

练习：求解LCS问题

讲师：liuyubobobo

版权所有，侵权必究

dijkstra 单源最短路径算法也是动态规划

shortestPath(i) 为从start到i的最短路径长度

$$\text{shortestPath}(x) = \min(\text{shortestPath}(a) + w(a \rightarrow x))$$

练习：复习单源最短路径算法

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

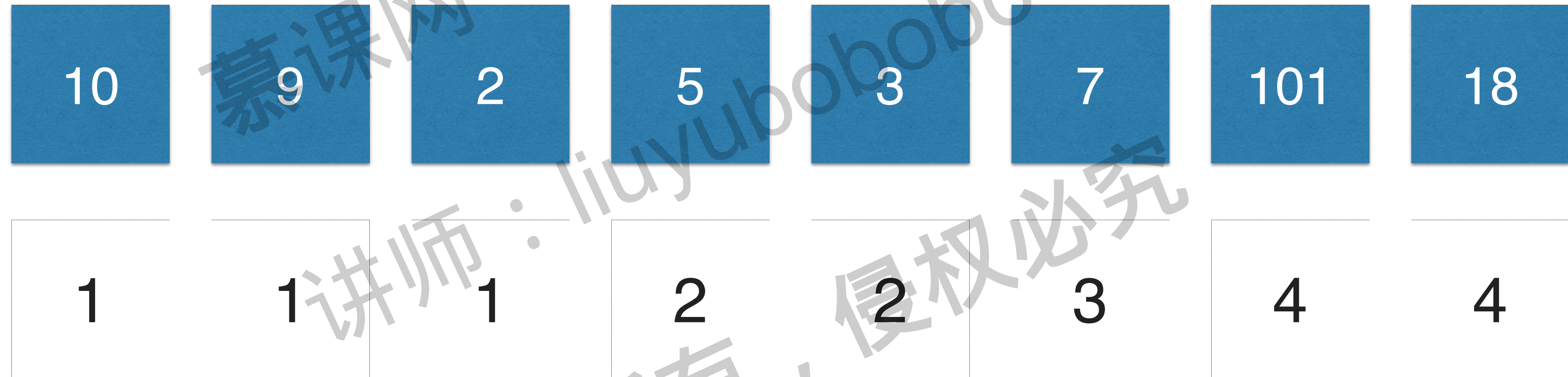
慕课网《玩转算法面试》

动态规划给出具体解

讲师：huayubobo

版权所有，侵权必究

300. Longest Increasing Subsequence

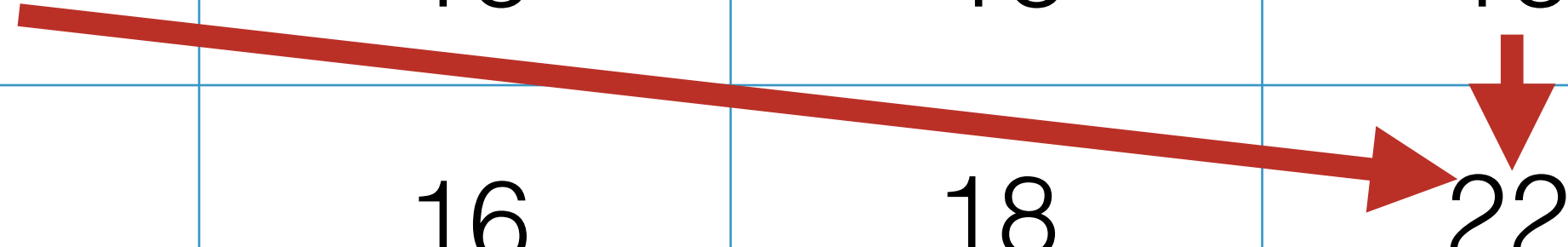


0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10	16	18	22

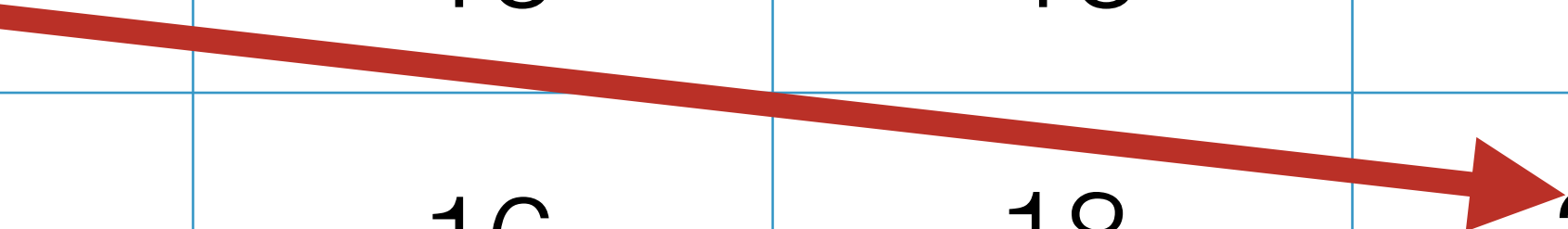


0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10	16	18	22

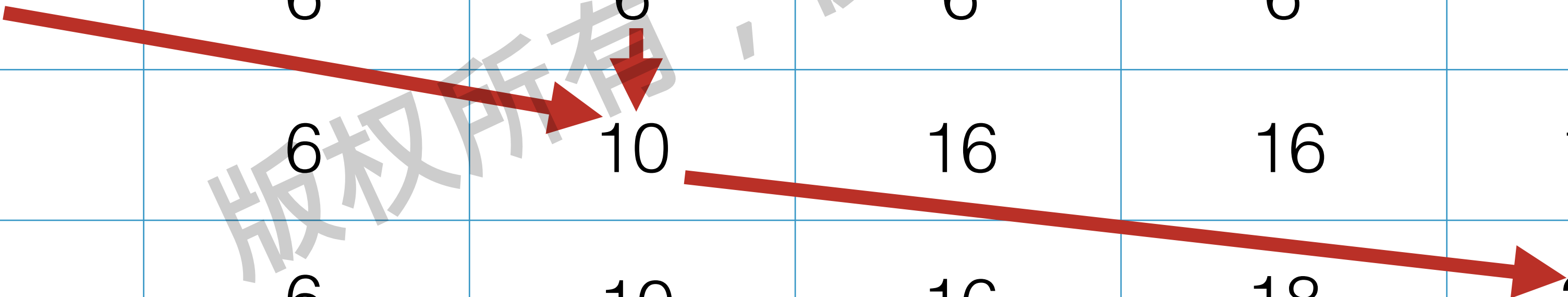


0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10	16	18	22




0-1 背包问题

id	0	1	2
weight	1	2	3
value	6	10	12

有一个容量为5的背包

	0	1	2	3	4	5
0	0	6	6	6	6	6
1	0	6	10	16	16	16
2	0	6	10	16	18	22



Leetcode上更多动态规划问题

慕课网《玩转算法面试》

讲师：liuyubobobo

版权所有，侵权必究

其他

欢迎大家关注我的个人公众号：是不是很酷



慕课网《玩转算法面试》

玩儿转算法面试

讲师：liuyubobobo

版权所有，侵权必究

liuyubobobo