

线性表

田船长

学习前需要掌握的C语言知识

1. 指针 —— 指向某元素的内存地址
2. 成员访问运算符的区别 —— “.” 与 “->”
3. 空间的分配与释放 —— 内存管理系列函数
4. 参考手册: cppreference.com

两个式子

程序 = 算法 + 数据结构

数据结构 = 结构定义 + 结构操作

线性表的顺序存储又叫做顺序表，它是由一组地址连续的存储单元依次存储线性表中的数据元素，从而使得逻辑上相邻的两个元素，在物理位置上也相邻。（大家可以先简单的理解成数组）

num[0]	num[1]	num[2]	num[3]	num[4]	num[5]	num[6]	num[7]	num[8]
--------	--------	--------	--------	--------	--------	--------	--------	--------

顺序表的结构定义

```
#define MaxCnt 50 //顺序表的最长长度
struct Sqlist { //顺序表的定义
    EleType data[MaxCnt]; //顺序表的数据
    int len; //顺序表的长度
};
```

顺序表既可以静态分配空间，又可以动态分配空间
动态分配时，需要将data变为指针

顺序表的插入

6	1	9	3	2				
---	---	---	---	---	--	--	--	--



在这里插入元素5

size = 5

顺序表的插入

6	1		9	3	2			
---	---	--	---	---	---	--	--	--



先将后面的元素后移一位
空出位置

size = 5

顺序表的插入

6	1	5	9	3	2			
---	---	---	---	---	---	--	--	--



插入元素5
插入完毕

size = 6

顺序表的删除

6	1	5	9	3	2			
---	---	---	---	---	---	--	--	--



删除元素5

size = 6

顺序表的删除

6	1	9	3	2				
---	---	---	---	---	--	--	--	--



将后面的元素依次向前移动
删除完毕

size = 5

顺序表的链式存储又叫做链表，在链表中每个元素的内存空间不连续，中间使用指针连接



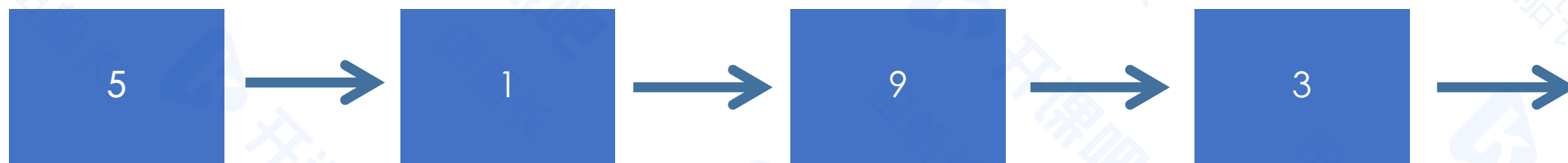
单链表

```
struct Node { //单链表的结点定义
    EleType data; //结点的数据域
    Node *next; //结点的指针域
};
```

```
struct list { //单链表的定义
    Node *head; //单链表头结点
    int len; //单链表的长度
};
```

通常用链表头结点指针与链表长度
来表示一个单链表

单链表的插入



在值为1的元素后插入值为2的结点

size = 4

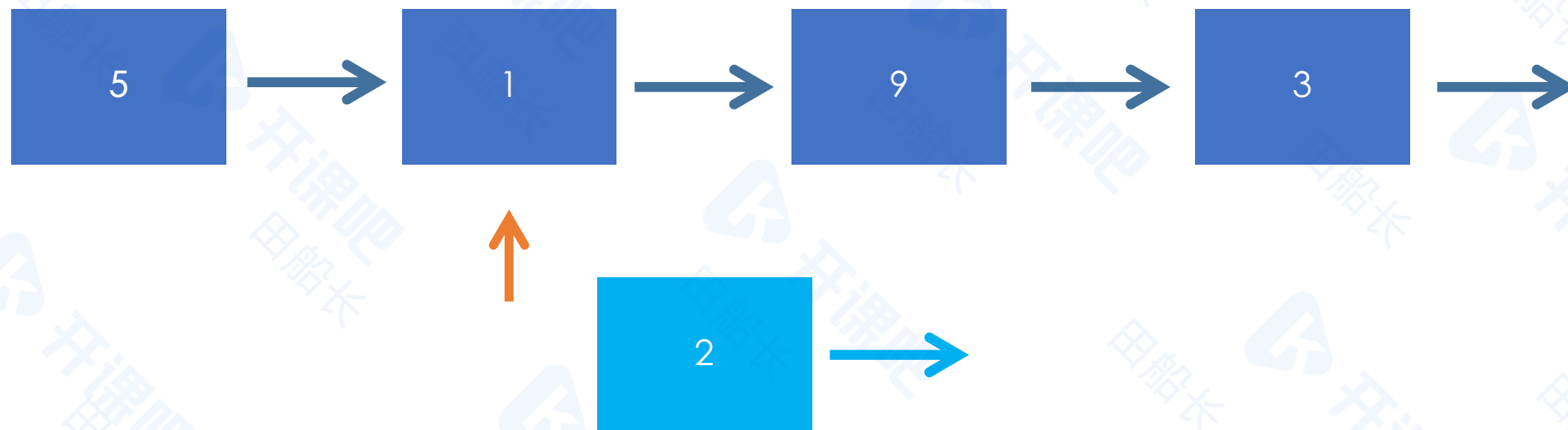
单链表的插入



先顺序查找到值为1的元素位置
在此之后进行插入操作

size = 4

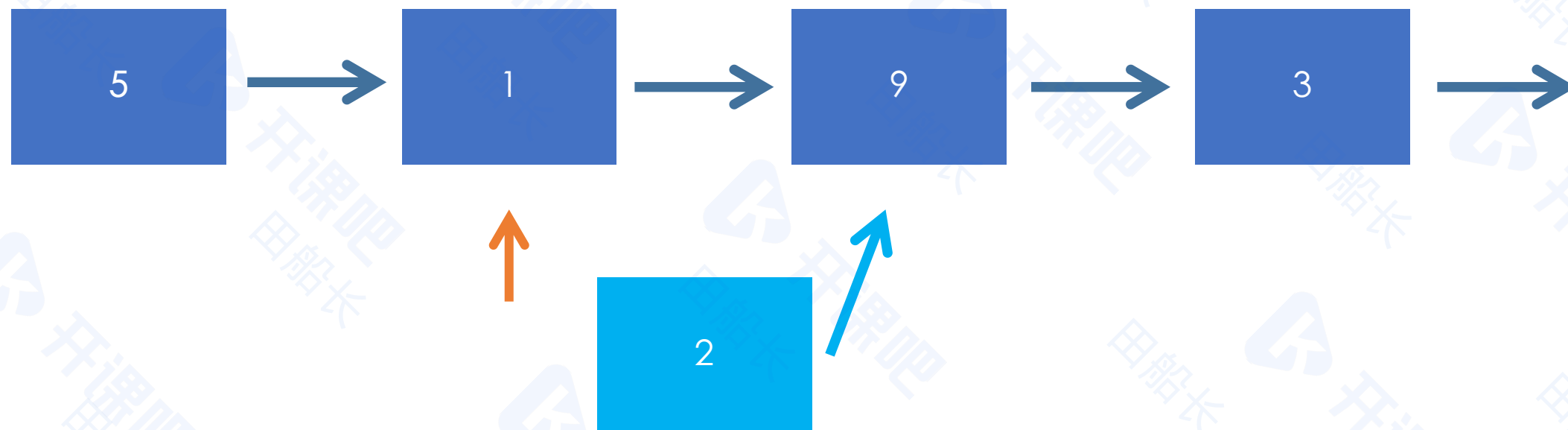
单链表的插入



size = 4

建立新的结点

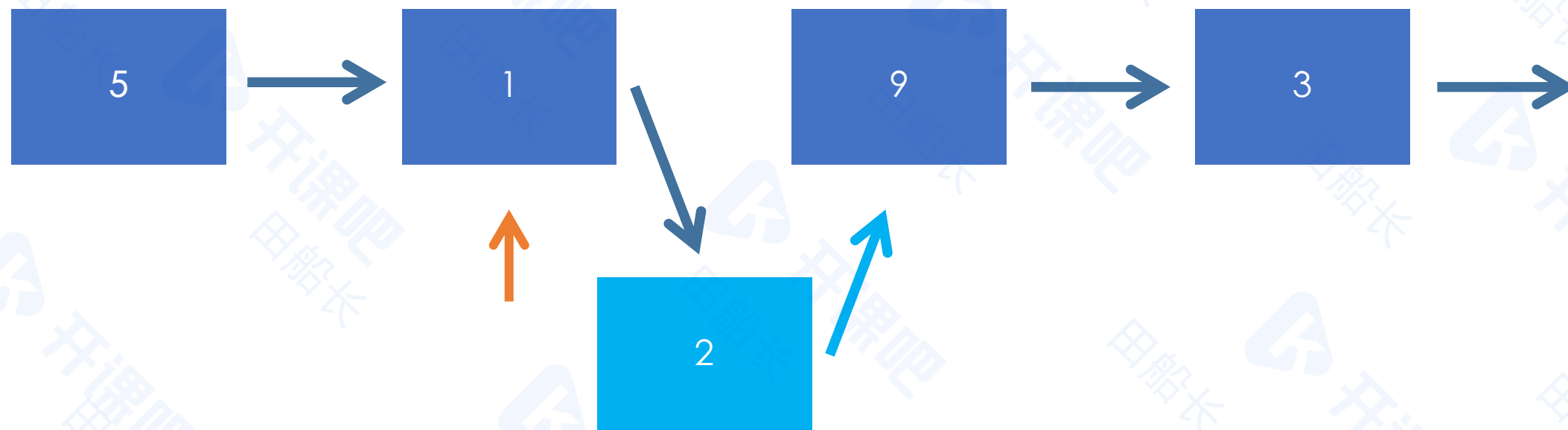
单链表的插入



size = 4

按顺序调整指针

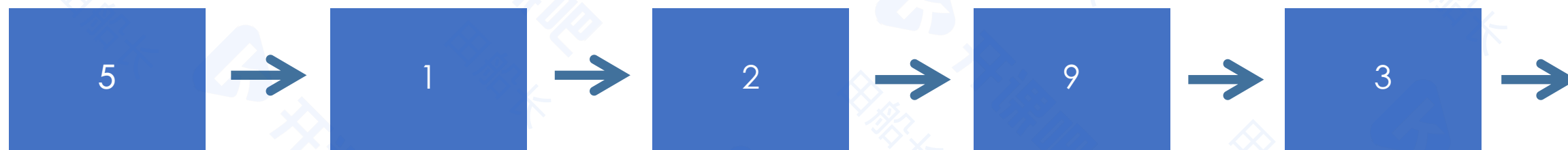
单链表的插入



size = 4

按顺序调整指针

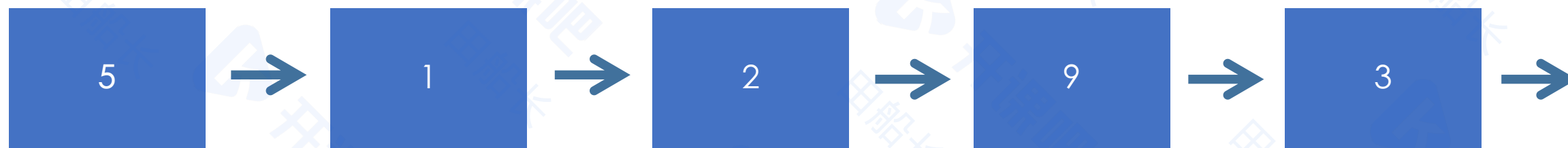
单链表的插入



size = 5

插入完毕

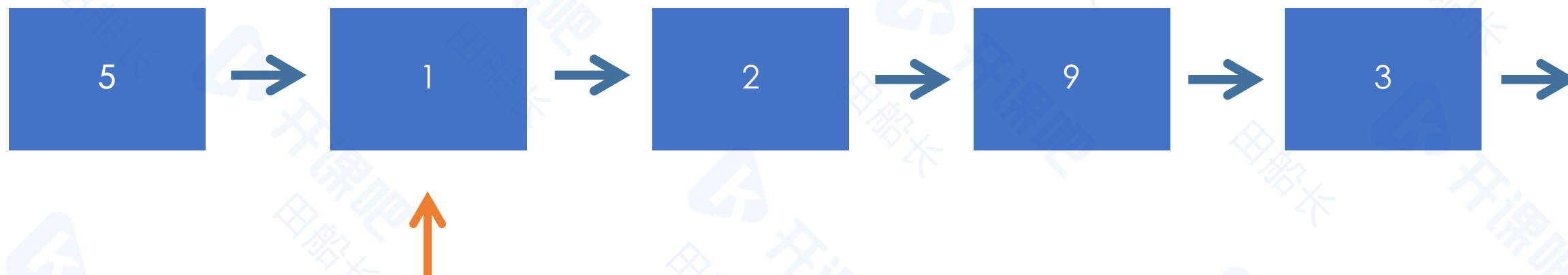
单链表的删除



想要删除值为1的元素后面的元素

size = 5

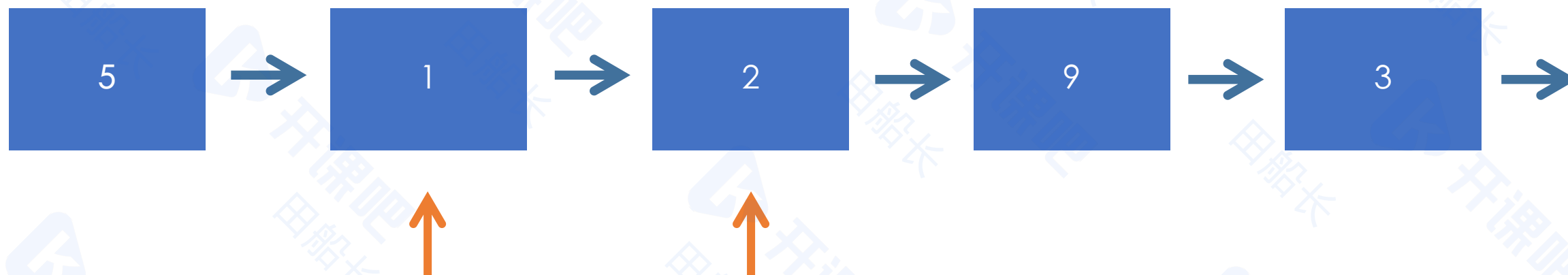
单链表的删除



先顺序查找到值为1的元素位置
准备删除

size = 5

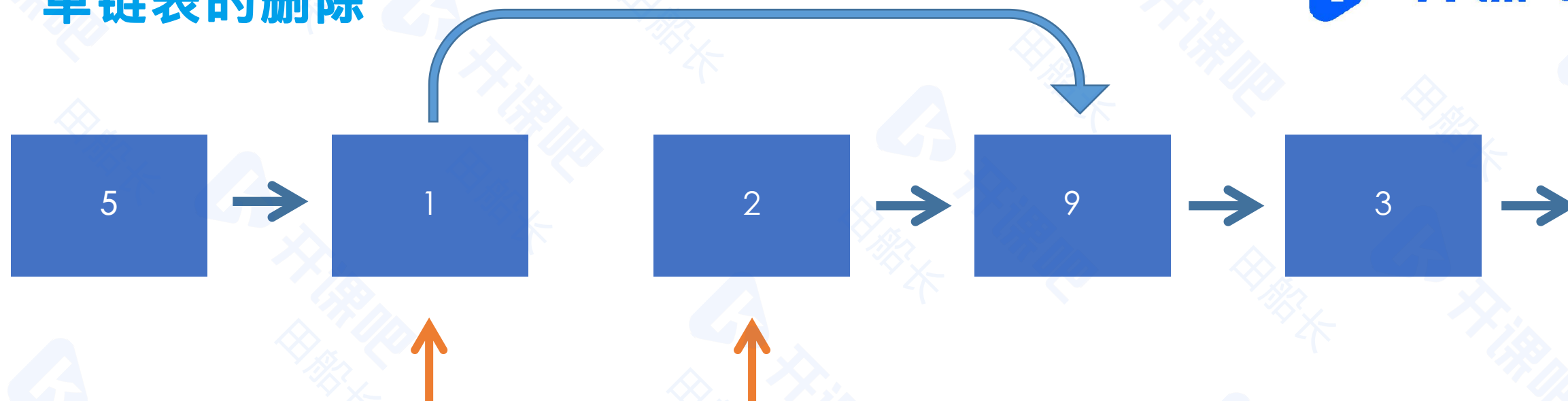
单链表的删除



记录要删除的元素地址

size = 5

单链表的删除



调整指针

size = 5

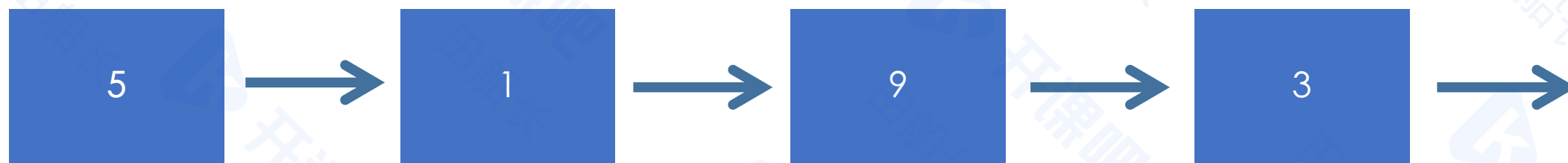
单链表的删除



释放空间

size = 5

单链表的删除



删除完毕

size = 4

单链表的注意点

1. 虚拟头结点问题

2. 头插法和尾插法

指针域除了指向后一个元素的指针外，还有指向前一个元素的指针，这样的链表是双链表



双链表

```
struct Node { //双链表的结点定义
    EleType data; //结点的数据域
    Node *front, *next; //结点的指针域
};

struct list { //双链表的定义
    Node *head, *tail; //双链表的头结点与尾结点
    int len; //双链表的长度
};
```

通常用链表头结点指针、尾结点指针与链表长度
来表示一个双链表

双链表与单链表在操作上相似，只需在单链表的基础上额外操作指向前一个结点的指针即可

在单链表的基础上，最后一个元素的指针指向第一个元素的链表，就是循环链表



循环链表在存储时，一般存储尾结点
与双链表相似，还有双向循环链表

顺序表与链表的对比

1. 存取方式不同，是否支持随机访问
2. 插入、删除、查找操作不同
3. 空间分配不同