

本节内容

循环链表

王道考研/CSKAOYAN.COM

1

知识总览

循环链表

循环单链表

循环双链表

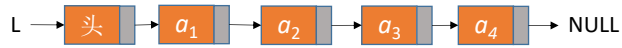
就那么简单



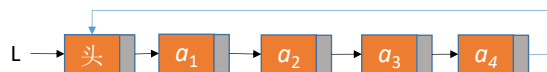
王道考研/CSKAOYAN.COM

2

循环单链表



单链表：表尾结点的next指针指向 NULL



循环单链表：表尾结点的next指针指向头结点

王道考研/CSKAOYAN.COM

3

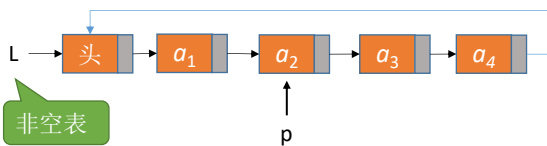
循环单链表

```
typedef struct LNode{
    ElemType data;
    struct LNode *next;
}LNode, *LinkList;
```

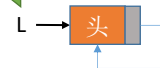
//定义单链表结点类型
//每个节点存放一个数据元素
//指针指向下一个节点

//初始化一个循环单链表

```
bool InitList(LinkList &L) {
    L = (LNode *) malloc(sizeof(LNode)); //分配一个头结点
    if (L==NULL) //内存不足，分配失败
        return false;
    L->next = L; //头结点next指向头结点
    return true;
}
```



空表



```
//判断循环单链表是否为空
bool Empty(LinkList L) {
    if (L->next == L)
        return true;
    else
        return false;
}
```

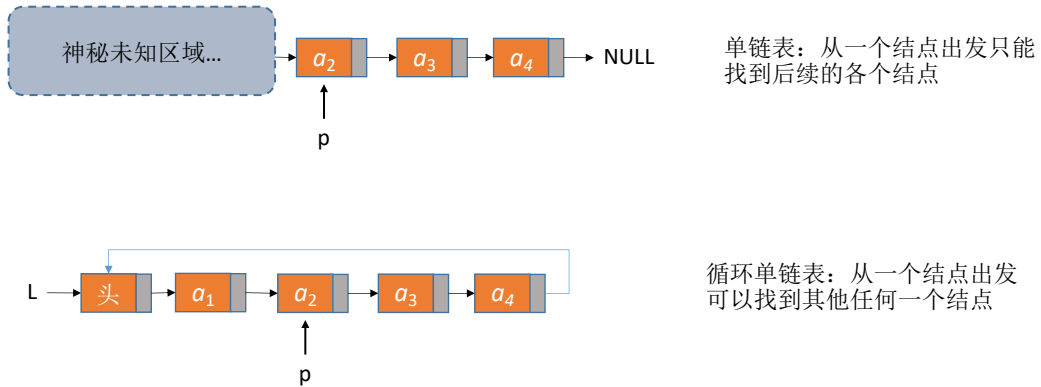
//判断结点p是否为循环单链表的表尾结点

```
bool isTail(LinkList L, LNode *p){
    if (p->next==L)
        return true;
    else
        return false;
}
```

王道考研/CSKAOYAN.COM

4

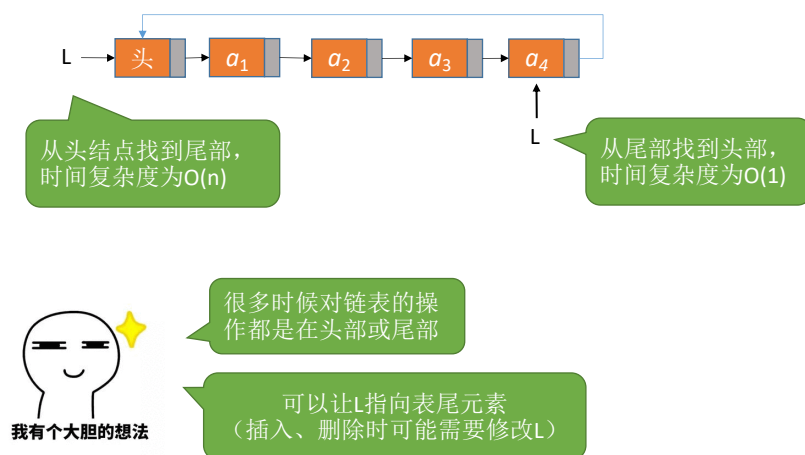
循环单链表



王道考研/CSKAOYAN.COM

5

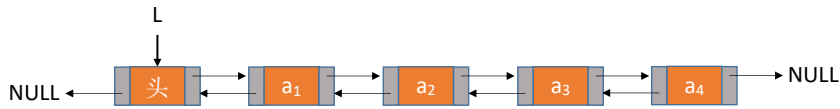
循环单链表



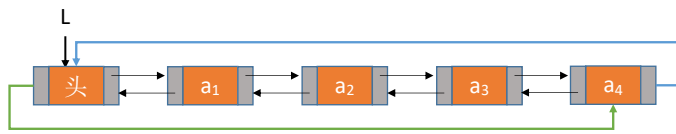
王道考研/CSKAOYAN.COM

6

循环双链表



双链表：
表头结点的 prior 指向 NULL；
表尾结点的 next 指向 NULL



循环双链表：
表头结点的 prior 指向表尾结点；
表尾结点的 next 指向头结点

王道考研/CSKAOYAN.COM

7

循环双链表的初始化

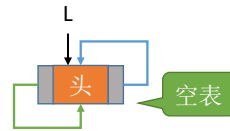
```
//初始化空的循环双链表
bool InitDLinkedList(DLinkedList &L){
    L = (DNode *) malloc(sizeof(DNode)); //分配一个头结点
    if (L==NULL) //内存不足, 分配失败
        return false;
    L->prior = L; //头结点的 prior 指向头结点
    L->next = L; //头结点的 next 指向头结点
    return true;
}

typedef struct DNode{
    ElemType data;
    struct DNode *prior,*next;
}DNode, *DLinkedList;

void testDLinkedList() {
    //初始化循环双链表
    DLinkedList L;
    InitDLinkedList(L);
    //...后续代码...
}

//判断循环双链表是否为空
bool Empty(DLinkedList L) {
    if (L->next == L)
        return true;
    else
        return false;
}

//判断结点p是否为循环单链表的表尾结点
bool isTail(DLinkedList L, DNode *p){
    if (p->next==L)
        return true;
    else
        return false;
}
```

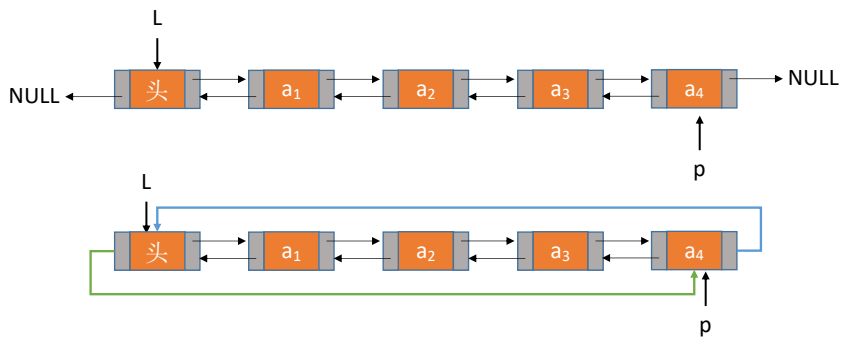


王道考研/CSKAOYAN.COM

8

双链表的插入

```
//在p结点之后插入s结点
bool InsertNextDNode(DNode *p, DNode *s){
    s->next=p->next;    //将结点*s插入到结点*p之后
    p->next->prior=s;
    s->prior=p;
    p->next=s;
}
```

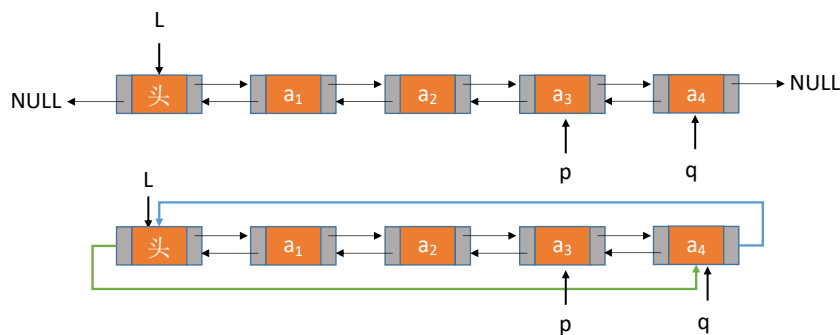


王道考研/CSKAOYAN.COM

9

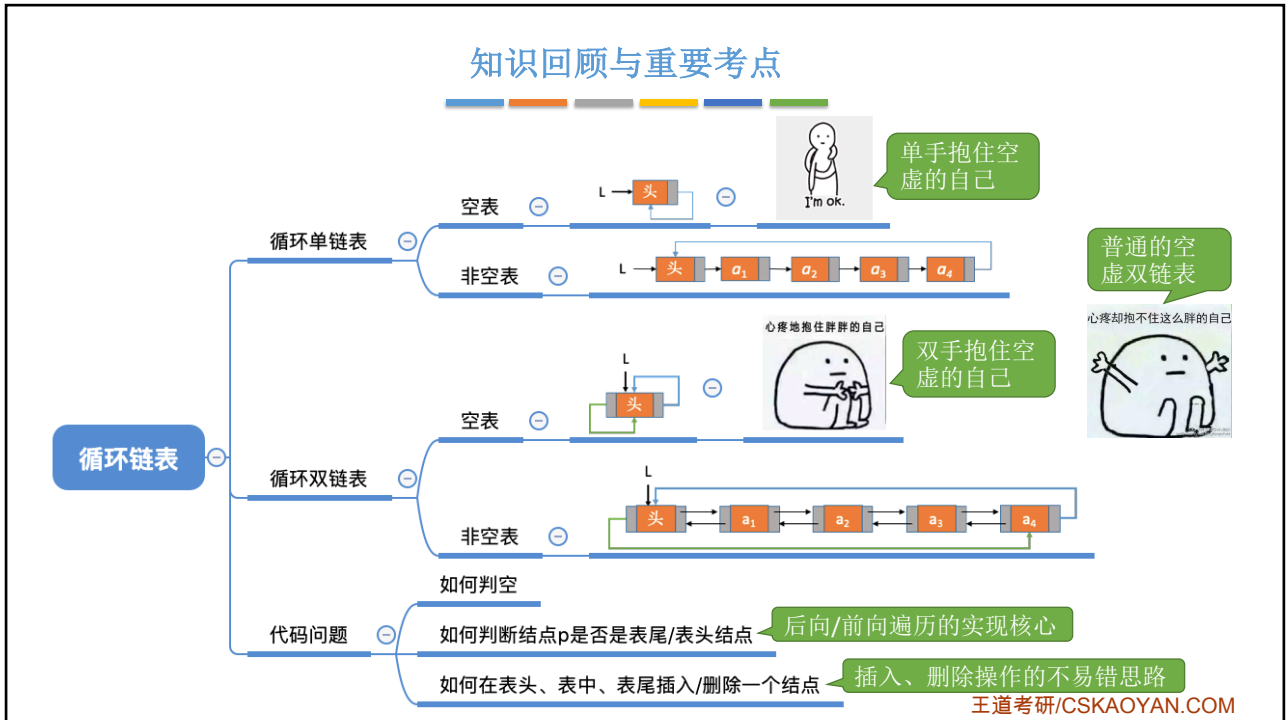
双链表的删除

```
//删除p的后继结点q
p->next=q->next;
q->next->prior=p;
free(q);
```



王道考研/CSKAOYAN.COM

10



11



@王道论坛



等撩



@王道计算机考研备考
@王道咸鱼老师-计算机考研
@王道楼楼老师-计算机考研



等撩



@王道计算机考研



@王道计算机考研



@王道计算机考研



@王道在线

12