

```
按位序插入(带头结点)
//在第 i 个位置插插入元素 e (带头结点)
                                                 typedef struct LNode{
bool ListInsert(LinkList &L, int i, ElemType e){
                                                    ElemType data;
   if(i<1)
                                                    struct LNode *next;
      return false;
                                                 }LNode, *LinkList;
   LNode *p; //指针p指向当前扫描到的结点
   int j=0;
             //当前p指向的是第几个结点
   p = L;
             //L指向头结点,头结点是第0个结点(不存数据)
                                                 ①如果 i = 1 (插在表头)
   while (p!=NULL && j<i-1) { //循环找到第 i-1 个结点
      p=p->next;
      j++;
   if(p==NULL)
                //i值不合法
      return false;
 LNode *s = (LNode *)malloc(sizeof(LNode));
 ⇒ s->data = e;
 s->next=p->next;
 p->next=s; //将结点s连到p之后
              //插入成功
   return true;
                              注意: 绿绿和黄黄顺序不能颠倒鸭!
                                                             王道考研/CSKAOYAN.COM
```

按位序插入(带头结点) //在第 i 个位置插插入元素 e (带头结点) typedef struct LNode{ bool ListInsert(LinkList &L, int i, ElemType e){ ElemType data; **if**(i<1) struct LNode *next; return false; }LNode, *LinkList; LNode *p; //指针p指向当前扫描到的结点 int j=0; //当前p指向的是第几个结点 分析: //L指向头结点,头结点是第0个结点(不存数据) p = L;while (p!=NULL && j<i-1) { //循环找到第 i-1 个结点 ②如果i=3(插在表中) p=p->next; j++; if(p==NULL) //i值不合法 return false; ■ LNode *s = (LNode *)malloc(sizeof(LNode)); \Rightarrow s->data = e; ⇒ s->next=p->next; \Rightarrow p->next=s; //将结点s连到p之后 return true; //插入成功 王道考研/CSKAOYAN.COM

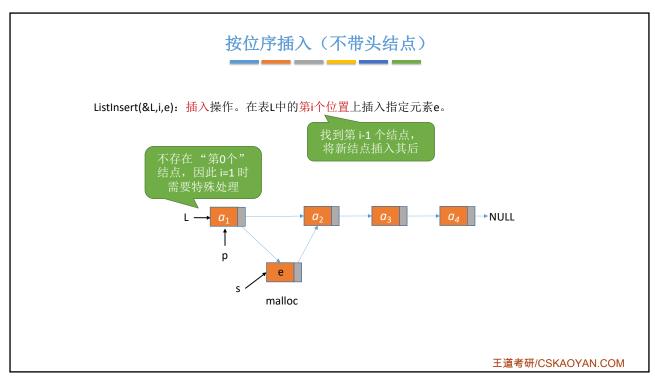
7

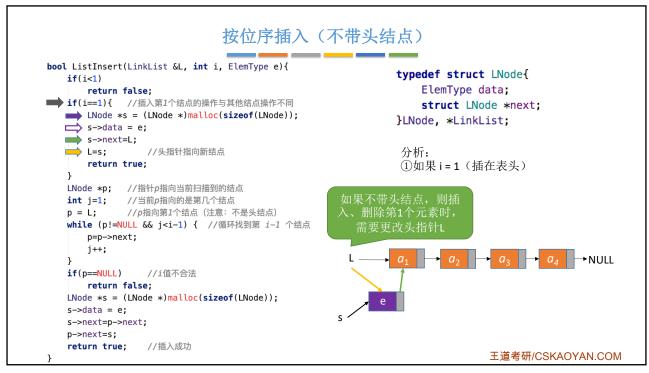
```
按位序插入(带头结点)
//在第 i 个位置插插入元素 e (带头结点)
                                                   typedef struct LNode{
bool ListInsert(LinkList &L, int i, ElemType e){
                                                      ElemType data;
   if(i<1)
                                                      struct LNode *next;
       return false;
                                                   }LNode, *LinkList;
   LNode *p; //指针p指向当前扫描到的结点
   int j=0;
             //当前p指向的是第几个结点
   p = L;
             //L指向头结点,头结点是第0个结点(不存数据)
                                                   ③如果i=5(插在表尾)
   while (p!=NULL && j<i-1) { //循环找到第 i-1 个结点
       p=p->next;
       j++;
   if(p==NULL)
                 //i值不合法
      return false;
  LNode *s = (LNode *)malloc(sizeof(LNode));
\Rightarrow s->data = e;
  s->next=p->next;
→ p->next=s;
                //将结点s连到p之后
                                           最坏时间复杂度: O(n)
   return true;
                //插入成功
1
                                                               王道考研/CSKAOYAN.COM
```

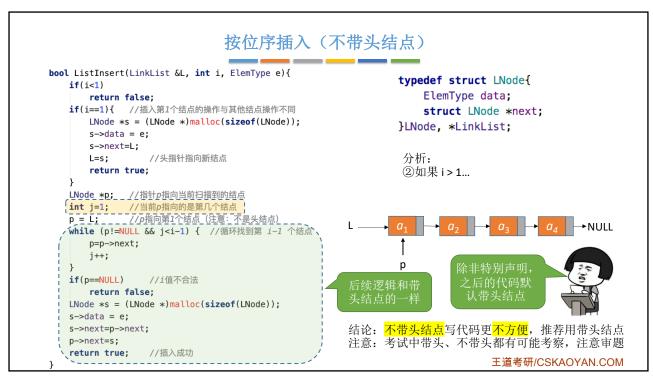
按位序插入(带头结点) //在第 i 个位置插插入元素 e (带头结点) typedef struct LNode{ bool ListInsert(LinkList &L, int i, ElemType e){ ElemType data; **if**(i<1) struct LNode *next; return false; }LNode, *LinkList; LNode *p; //指针p指向当前扫描到的结点 int j=0; //当前p指向的是第几个结点 p = L; //L指向头结点,头结点是第0个结点(不存数据) 分析: while (p!=NULL && j<i-1) { //循环找到第 *i-1* 个结点 ④如果 i = 6 (i>Lengh) p=p->next; j++; if(p==NULL) //i值不合法 return false; LNode *s = (LNode *)malloc(sizeof(LNode)); s->data = e;s->next=p->next; p->next=s; //将结点s连到p之后 return true; //插入成功 } 王道考研/CSKAOYAN.COM

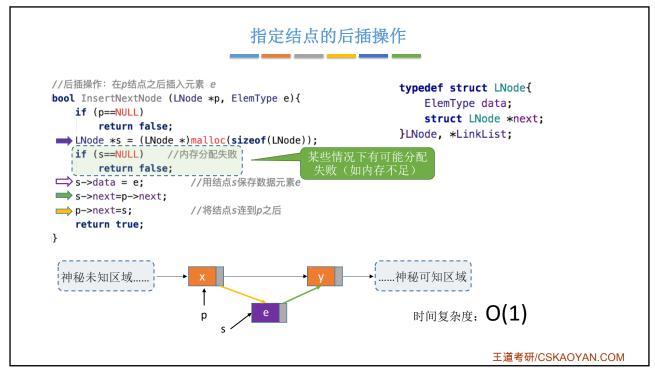
按位序插入(带头结点) //在第 i 个位置插插入元素 e (带头结点) typedef struct LNode{ bool ListInsert(LinkList &L, int i, ElemType e){ ElemType data; **if**(i<1) struct LNode *next; return false; }LNode, *LinkList; LNode *p; //指针p指向当前扫描到的结点 int j=0; //当前p指向的是第几个结点 //L指向头结点,头结点是第0个结点(不存数据) 平均时间复杂度: O(n)while (p!=NULL && j<i-1) { //循环找到第 i-1 个结点 p=p->next; 原来如此,简单! j++; **if**(p==NULL) //i值不合法 return false; LNode *s = (LNode *)malloc(sizeof(LNode)); s->data = e; s->next=p->next; p->next=s; //将结点s连到p之后 return true; //插入成功 } 王道考研/CSKAOYAN.COM

10

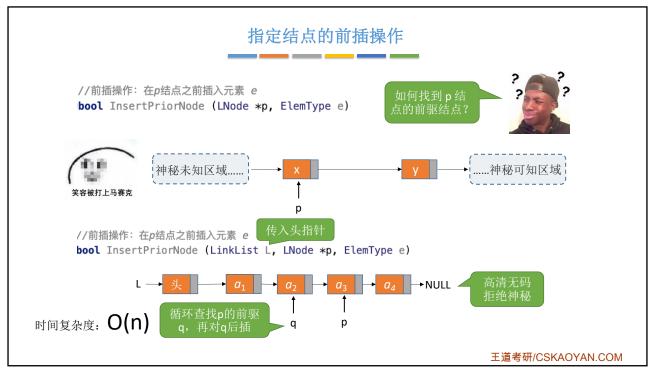


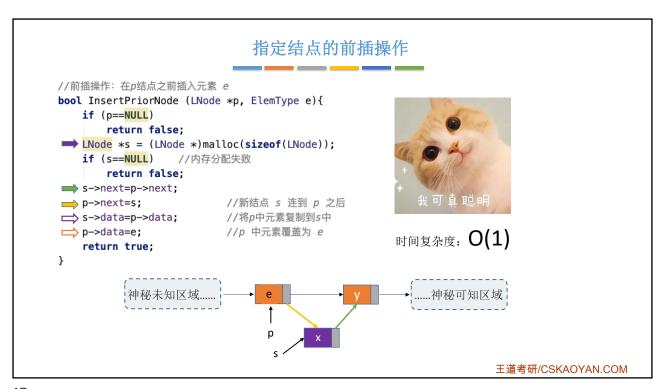




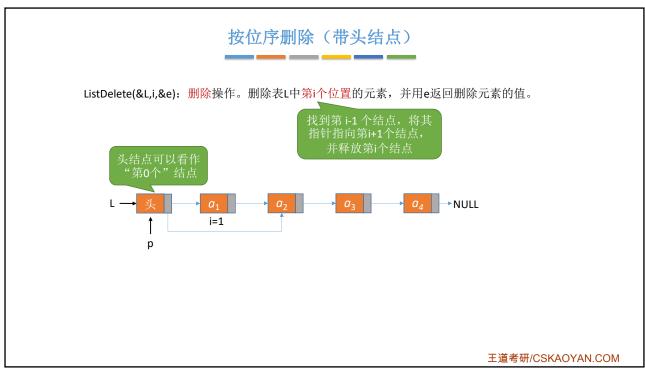


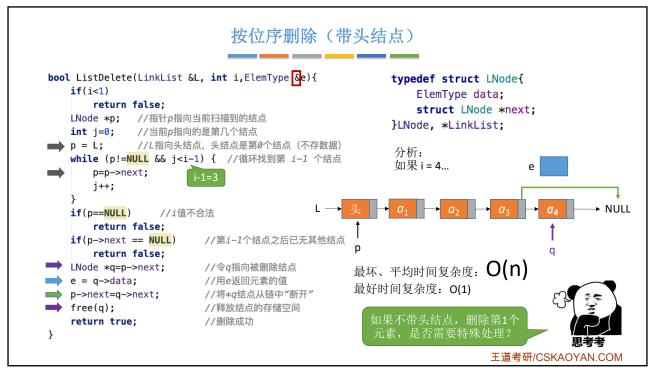
```
指定结点的后插操作
//在第 i 个位置插插入元素 e (带头结点)
                                                       typedef struct LNode{
bool ListInsert(LinkList &L, int i, ElemType e){
                                                           ElemType data;
   if(i<1)
                                                           struct LNode *next;
       return false;
                                                       }LNode, *LinkList;
   LNode *p; //指针p指向当前扫描到的结点
   int j=0;
             //当前p指向的是第几个结点
             //L指向头结点,头结点是第0个结点(不存数据)
   p = L;
   while (p!=NULL && j<i-1) { //循环找到第 i-1 个结点
                                                    //后插操作: 在p结点之后插入元素 e
      p=p->next;
                                                    bool InsertNextNode (LNode *p, ElemType e){
      j++;
                                                       if (p==NULL)
                                                          return false;
                                                       LNode *s = (LNode *)malloc(sizeof(LNode));
    return InsertNextNode(p, e);
                                                                   //内存分配失败
                                                       if (s==NULL)
                                                          return false;
                                                       s->data = e:
                                                                        //用结点s保存数据元素e
   s->next=p->next;
                                                       s->next=p->next;
   p->next=s;
                                                       p->next=s;
                                                                        //将结点s连到p之后
    eturn true;
                                                       return true;
                                                                      王道考研/CSKAOYAN.COM
```

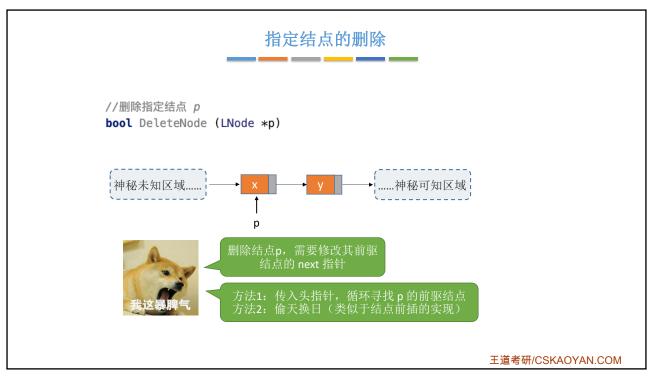


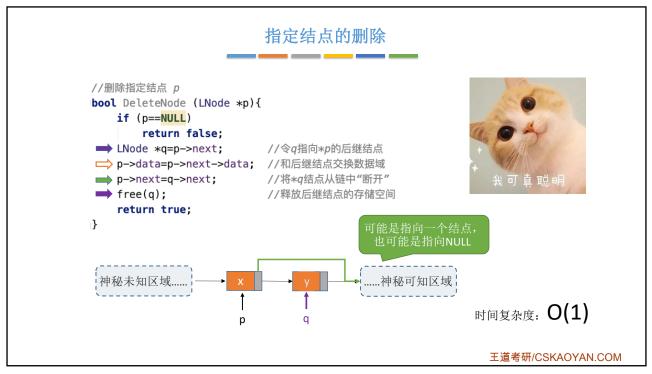


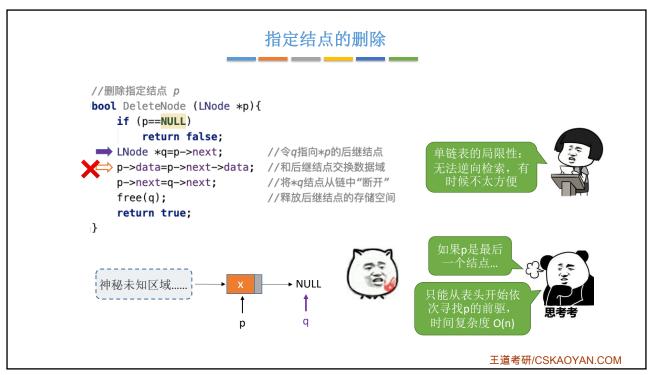
```
指定结点的前插操作
//前插操作: 在p结点之前插入结点 s
bool InsertPriorNode (LNode *p, LNode *s){
   if (p==NULL || s==NULL)
                                   王道书版本
       return false;
   s->next=p->next;
   p->next=s;
                       //s连到p之后
  ElemType temp=p->data; //交换数据域部分
   p->data=s->data;
   s->data=temp;
   return true;
                           temp
}
       神秘未知区域.....
                                             .....神秘可知区域
                                                      王道考研/CSKAOYAN.COM
```

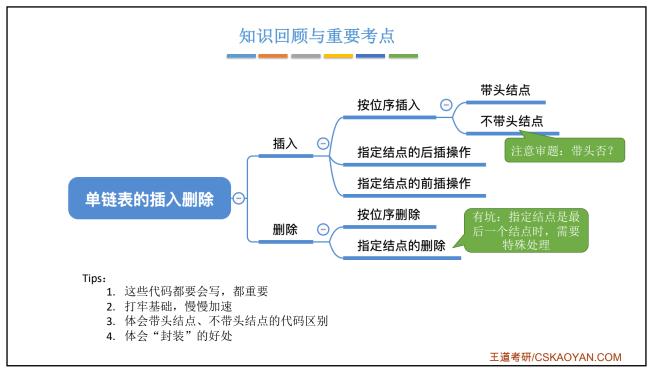


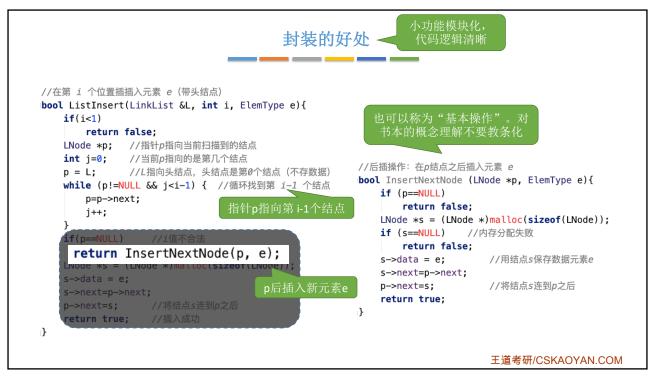


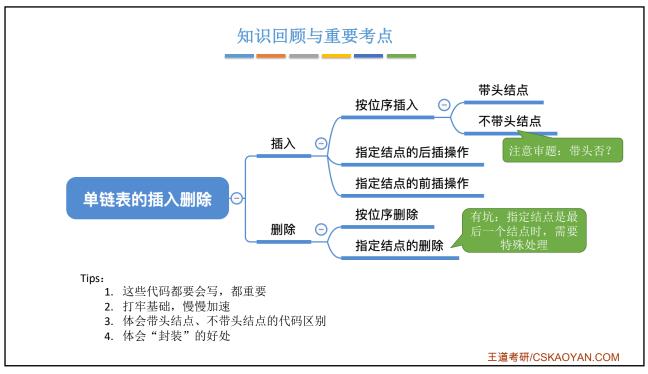


















@王道论坛



@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



@王道计算机考研

知乎

※ 微信视频号



@王道计算机考研

@王道计算机考研

@王道在线