

本节内容

链栈
的实现

王道考研/CSKAOYAN.COM

1

知识总览

用链式存储方式实现的栈

链栈

- 基本操作
 - 创（初始化）
 - 增（进栈）
 - 删（出栈）
 - 查（获取栈顶元素）
 - 判空、判满

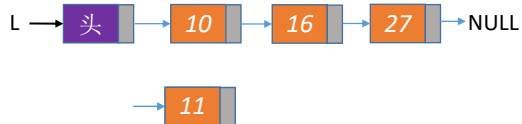
叫你偷懒

王道考研/CSKAOYAN.COM

2

穿越：头插法建立单链表

对应：进栈

对头结点的
后插操作

```
//后插操作：在p结点之后插入元素 e
bool InsertNextNode (LNode *p, ElemType e){
    if (p==NULL)
        return false;
    LNode *s = (LNode *)malloc(sizeof(LNode));
    if (s==NULL) //内存分配失败
        return false;
    s->data = e; //用结点s保存数据元素e
    s->next=p->next;
    p->next=s; //将结点s连到p之后
    return true;
}
```

头插法建立单链表：

初始化单链表

```
While 循环 {
    每次取一个数据元素 e;
    InsertNextNode (L, e);
}
```

叫你偷懒



王道考研/CSKAOYAN.COM

3

穿越：单链表的删除操作

对应：出栈

对头结点的
“后删”操作

又偷懒!!



王道考研/CSKAOYAN.COM

4

链栈的定义

```
typedef struct Linknode{
    ElemType data;           //数据域
    struct Linknode *next;   //指针域
} *LiStack;                 //栈类型定义
```

带头结点的初始化

S → 头 → NULL

S → 头 → d → c → b → a → NULL

不带头结点的初始化

S → NULL

S → d → c → b → a → NULL

进栈/出栈都只能在栈顶一端进行
(链头作为栈顶)

王道考研/CSKAOYAN.COM

5

知识回顾与重要考点

链栈

用链式存储方式实现的栈

两种实现方式

- 带头结点
- 不带头结点 (推荐)

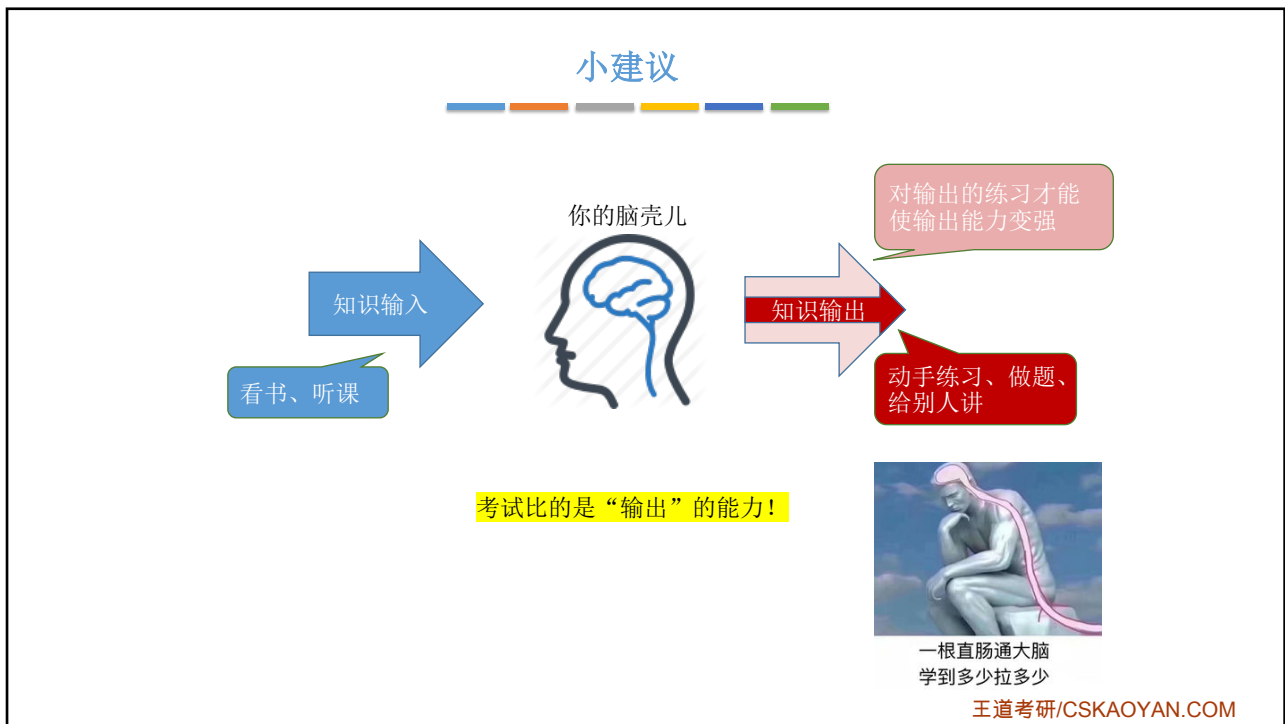
重要基本操作

- 创 (初始化)
- 增 (进栈)
- 删 (出栈)
- 查 (获取栈顶元素)
- 如何判空、判满?

动手写一遍

王道考研/CSKAOYAN.COM

6



7