

王道考研——数据结构

WWW.CSKAOYAN.COM

第二章 线性表

1

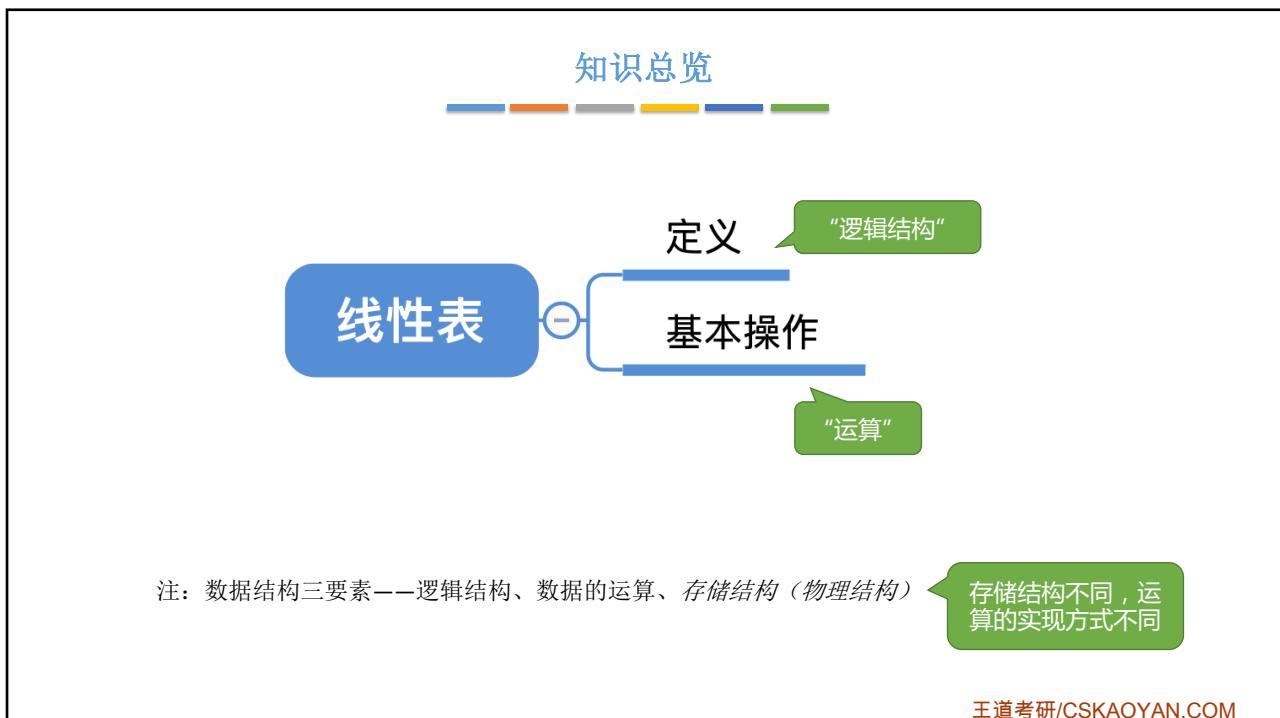
本节内容

线性表

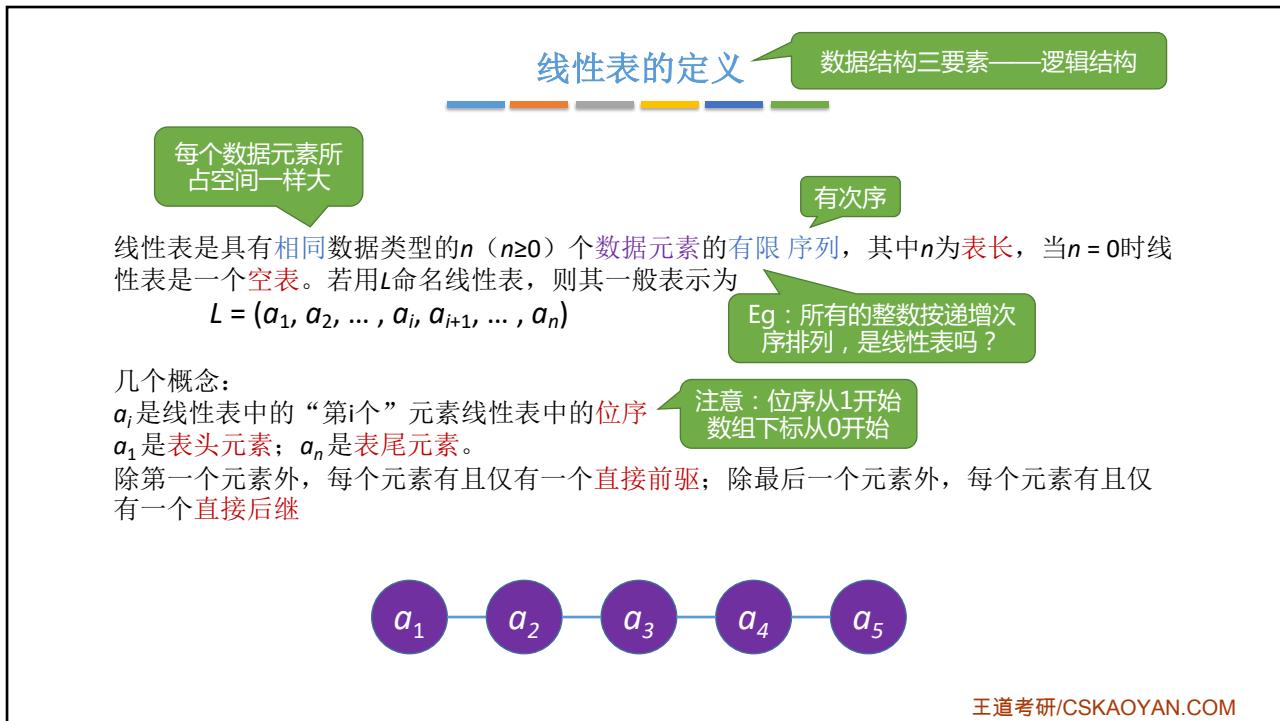
定义、
基本操作

王道考研/CSKAOYAN.COM

2



3



4

线性表的定义

线性表 —— Linear List

	A	B	C	D
1	学号	姓名	性别	专业
2	1120112100	张三	男	挖掘机
3	1120112101	李四	女	挖掘机
4	1120112102	王五	男	数据挖掘
5	1120112103	赵六	男	挖掘机
6	1120112104	钱七	女	挖掘机
7	1120112105	狗剩	男	数据挖掘
8	1120112106	铁柱	女	数据挖掘
9	1120112107	如花	女	数据挖掘
10	1120112108	二狗	男	数据挖掘
11	1120112109	傻根儿	男	数据挖掘
12	1120112110	旺财	女	数据挖掘

 吃惊

linear 英 ['lɪniə(r)] 美 ['lɪnɪər] adj. 线的, 线型的; 直线的, 线状的; 长度的

list 英 [lɪst] 美 [lɪst] n. **列表, 清单, 目录;**

词根 : Line 线 Eg : Sky line baby

TODO List 6

- 学习《程序员成长之路》
- 学习《C 语言入门到精通》
- 学习《数据结构》
- 学习《如何成为软件架构师》
- 学习《腰椎间盘突出的预防与治疗》
- 看书《活着》

王道考研/CSKAOYAN.COM

5

线性表的基本操作

数据结构三要素——“运算”

WHY?

为什么要实现对数据结构的基本操作？

① 团队合作编程，你定义的数据结构要让别人能够很方便的使用（封装）
 ② 将常用的操作/运算封装成函数，避免重复工作，降低出错风险

咸鱼要翻身

Tips : 比起学会 “How”,
更重要的是想明白 “Why”

王道考研/CSKAOYAN.COM

6

线性表的基本操作

从无到有
从有到无

InitList(&L): 初始化表。构造一个空的线性表L, 分配内存空间。

DestroyList(&L): 销毁操作。销毁线性表, 并释放线性表L所占用的内存空间。

ListInsert(&L,i,e): 插入操作。在表L中的第i个位置上插入指定元素e。

ListDelete(&L,i,&e): 删除操作。删除表L中第i个位置的元素, 并用e返回删除元素的值。

LocateElem(L,e): 按值查找操作。在表L中查找具有给定关键字值的元素。

GetElem(L,i): 按位查找操作。获取表L中第i个位置的元素的值。

其他常用操作:

Length(L): 求表长。返回线性表L的长度, 即L中数据元素的个数。

PrintList(L): 输出操作。按前后顺序输出线性表L的所有元素值。

Empty(L): 判空操作。若L为空表, 则返回true, 否则返回false。

Tips:

- ①对数据的操作(记忆思路) —— 创销、增删改查
- ②C语言函数的定义 —— <返回值类型> 函数名(<参数1类型> 参数1, <参数2类型> 参数2,)
- ③实际开发中, 可根据实际需求定义其他的基本操作
- ④函数名和参数的形式、命名都可改变 (Reference: 严蔚敏版《数据结构》) → Key: 命名要有可读性
- ⑤什么时候要传入引用“&” —— 对参数的修改结果需要“带回来”

为什么这里没有说明
各个参数的具体类型?

王道考研/CSKAOYAN.COM

7

线性表的基本操作

什么时候要传入参数的引用“&” —— 对参数的修改结果需要“带回来”

菜鸟工具 WEB 在线编辑器 SVG 在线编辑器 实例归档 菜鸟教程 输入关键字.....

点击运行 标准输入(stdin) C++ 在线工具 清空

```

1 #include<stdio.h>
2
3 void test(int x) {
4     x=1024;
5     printf("test函数内部 x=%d\n",x);
6 }
7
8 int main() {
9     int x = 1;
10    printf("调用test前 x=%d\n",x);
11    test(x);
12    printf("调用test后 x=%d\n",x);
13 }
```

对参数的修改
“没带回来”

调用test前 x=1
test函数内部 x=1024
调用test后 x=1

王道考研/CSKAOYAN.COM

8

线性表的基本操作

The screenshot shows a C++ code example demonstrating pass-by-reference:

```

1 #include<stdio.h>
2
3 void test(int & x) {
4     x=1024;
5     printf("test函数内部 x=%d\n",x);
6 }
7
8 int main() {
9     int x = 1;
10    printf("调用test前 x=%d\n",x);
11    test(x);
12    printf("调用test后 x=%d\n",x);
13 }

```

A callout bubble points to the line `printf("调用test后 x=%d\n",x);` with the text "对参数的修改“带回来”了". To the right, a memory diagram shows a blue box labeled "内存" containing an orange "X" with the text "调用test前 x=1" above it, "test函数内部 x=1024" in the middle, and "调用test后 x=1024" below it.

王道考研/CSKAOYAN.COM

9

线性表的基本操作

InitList(&L): 初始化表。构造一个空的线性表L, 分配内存空间。
DestroyList(&L): 销毁操作。销毁线性表，并释放线性表L所占用的内存空间。

ListInsert(&L,i,e): 插入操作。在表L中的第i个位置上插入指定元素e。
ListDelete(&L,i,&e): 删除操作。删除表L中第i个位置的元素，并用e返回删除元素的值。

LocateElem(L,e): 按值查找操作。在表L中查找具有给定关键字值的元素。
GetElem(L,i): 按位查找操作。获取表L中第i个位置的元素的值。

其他常用操作：
Length(L): 求表长。返回线性表L的长度，即L中数据元素的个数。
PrintList(L): 输出操作。按前后顺序输出线性表L的所有元素值。
Empty(L): 判空操作。若L为空表，则返回true，否则返回false。

Tips:

- ①对数据的操作（分析思路）——创销、增删改查
- ②C语言函数的定义——<返回值类型> 函数名 (<参数1类型> 参数1, <参数2类型> 参数2,)
- ③实际开发中，可根据实际需求定义其他的基本操作
- ④函数名和参数的形式、命名都可改变 (Reference: 严蔚敏版《数据结构》) → **Key:** 命名要有可读性
- ⑤什么时候要传入参数的引用“&”——对参数的修改结果需要“带回来”

为什么此处没有说明各个参数的具体类型？

王道考研/CSKAOYAN.COM

10

