

本节内容

顺序表

查找

王道考研/CSKAOYAN.COM

1

知识总览

顺序表的基本操作



王道考研/CSKAOYAN.COM

2

顺序表的按位查找

GetElem(L,i): 按位查找操作。获取表L中第i个位置的元素的值。

```
#define MaxSize 10           // 定义最大长度
typedef struct{             // 用静态的“数组”存放数据元素
    ElemenType data[MaxSize]; // 静态分配
    int length;              // 顺序表的当前长度
} SqList;                  // 顺序表的类型定义（静态分配方式）

ElemenType GetElem(SqList L, int i){
    return L.data[i-1];
}
```

王道考研/CSKAOYAN.COM

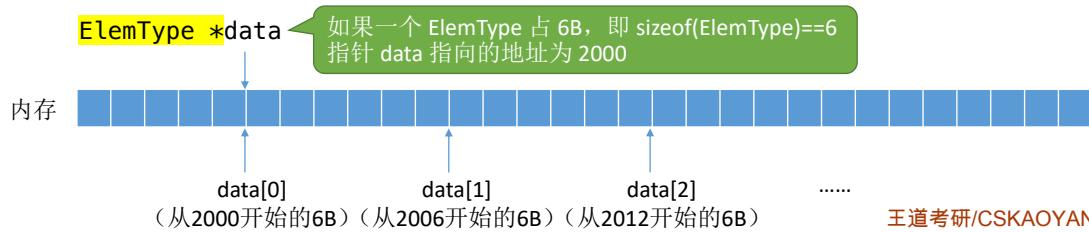
3

顺序表的按位查找

GetElem(L,i): 按位查找操作。获取表L中第i个位置的元素的值。

```
#define InitSize 10           // 顺序表的初始长度
typedef struct{              // 指示动态分配数组的指针
    ElemenType *data;        // 动态分配
    int MaxSize;             // 顺序表的最大容量
    int length;              // 顺序表的当前长度
} SeqList;                  // 顺序表的类型定义（动态分配方式）

ElemenType GetElem(SeqList L, int i){ // 和访问普通数组的方法一样
    return L.data[i-1];
}
```



4

顺序表的按位查找

GetElem(L,i): 按位查找操作。获取表L中第i个位置的元素的值。

```
#define InitSize 10           //顺序表的初始长度
typedef struct{
    ELEMTYPE *data;          //指示动态分配数组的指针 动态分配
    int MaxSize;             //顺序表的最大容量
    int length;               //顺序表的当前长度
} SeqList;                  //顺序表的类型定义（动态分配方式）
```

```
ElemType GetElem(SeqList L, int i){
    return L.data[i-1];
```

如果换一个类
型的指针，指
向同一个地址

int *p

p[0]

p[1]

p[2]

p[4]

.....

（注：一个int 占4B）

内存

data[0]

data[1]

data[2]

.....

（从2000开始的6B）（从2006开始的6B）（从2012开始的6B）

王道考研/CSKAOYAN.COM

5

顺序表的按位查找

int *p

p[0]

p[1]

p[2]

p[4]

.....

（注：一个int 占4B）

内存

data[0]

data[1]

data[2]

.....

ElemType *data

```
#define InitSize 10 //默认的最大长度
typedef struct{
    int *data;          //指示动态分配数组的指针
    int MaxSize;        //顺序表的最大容量
    int length;         //顺序表的当前长度
} SeqList;
```

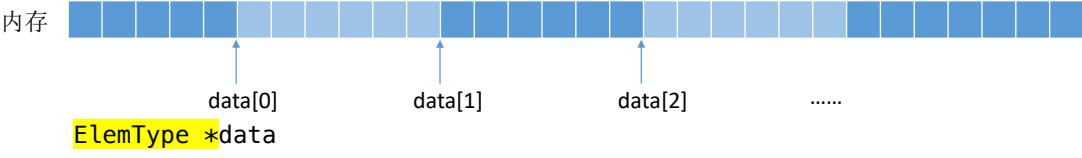
```
void InitList(SeqList &L){
    //用 malloc 函数申请一片连续的存储空间
    L.data=(int *)malloc(InitSize*sizeof(int));
    L.length=0;
    L.MaxSize=InitSize
}
```

再次理解，为何 malloc 函数返回的
存储空间起始地址要转换为与数
据元素的数据类型相对应的指针

王道考研/CSKAOYAN.COM

6

按位查找的时间复杂度



```

    内存
    data[0]   data[1]   data[2]   .....
    ElemType *data
  
```

时间复杂度: $O(1)$

由于顺序表的各个数据元素在内存中连续存放，因此可以根据起始地址和数据元素大小立即找到第 i 个元素——“随机存取”特性

王道考研/CSKAOYAN.COM

7

顺序表的按值查找

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。

```

#define InitSize 10          //顺序表的初始长度
typedef struct{
    ElemType *data;        //指示动态分配数组的指针
    int MaxSize;           //顺序表的最大容量
    int length;             //顺序表的当前长度
} SeqList;                //顺序表的类型定义（动态分配方式）

//在顺序表L中查找第一个元素值等于e的元素，并返回其位序
int LocateElem(SeqList L,ElemType e){
    for(int i=0;i<L.length;i++)
        if(L.data[i]==e)
            return i+1; //数组下标为i的元素值等于e，返回其位序i+1
    return 0;           //退出循环，说明查找失败
}
  
```

王道考研/CSKAOYAN.COM

8

顺序表的按值查找

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。

```

typedef struct {
    int *data;          // 指示动态分配数组的指针
    int MaxSize;       // 顺序表的最大容量
    int length;        // 顺序表的当前长度
} SeqList;

```

```

// 在顺序表L中查找第一个元素值等于e的元素，并返回其位序
int LocateElem(SeqList L, int e){
    for(int i=0; i<L.length; i++){
        if(L.data[i]==e)
            return i+1;
        return 0;
    }
}

```

调用: LocateElem(L, 9);

基本数据类型: int、char、double、float 等可以直接用运算符“==”比较
 结构类型的数据元素也这样吗?

内存
 L.data
 i=0 → ⑥data[0]
 i=1 → ②data[1]
 i=2 → ⑨data[2]
 ④data[3]
 ⑨data[4]
 ③data[5]
 data[6]
 data[7]
 data[8]
 data[9]
 length = 6

王道考研/CSKAOYAN.COM

9

结构类型的比较

```

typedef struct {
    int num;
    int people;
} Customer;

```

```

void test () {
    Customer a;
    a.num = 1;
    a.people = 1;
    Customer b;
    b.num = 1;
    b.people = 1;
    if (a == b) {
        printf("相等");
    } else {
        printf("不相等");
    }
}

```

不能

更好的办法: 定义一个函数, 童叟无欺, 用过都说好

注意: C语言中, 结构体的比较不能直接用“==”

```

bool isCustomerEqual (Customer a, Customer b){
    if (a.num == b.num && a.people == b.people)
        return true;
    else
        return false;
}

```

需要依次对比各个分量来
 判断两个结构体是否相等

王道考研/CSKAOYAN.COM

10

顺序表的按值查找

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。

```
//在顺序表L中查找第一个元素值等于e的元素，并返回其位序
int LocateElem(SeqList L, ELEMType e){
    for(int i=0; i<L.length; i++)
        if(L.data[i]==e)
            return i+1; //数组下标为i的元素值等于e，返回其位序i+1
    return 0; //退出循环，说明查找失败
}
```

Tips:

《数据结构》考研初试中，手写代码可以直接用“==”，无论 ELEMType 是基本数据类型还是结构类型

手写代码主要考察学生是否能理解算法思想，不会严格要求代码完全可运行

有的学校考《C语言程序设计》，那么...也许就要语法严格一些

王道考研/CSKAOYAN.COM

11

按值查找的时间复杂度

```
//在顺序表L中查找第一个元素值等于e的元素，并返回其位序
int LocateElem(SeqList L, ELEMType e){
    for(int i=0; i<L.length; i++)
        if(L.data[i]==e)
            return i+1; //关注最深层循环语句的执行次数与问题规模 n 的关系
    return 0; //返回其位序 i+1
} //问题规模 n = L.length (表长)
```

最好情况：目标元素在表头

循环1次；**最好时间复杂度 = O(1)**

最坏情况：目标元素在表尾

循环 n 次；**最坏时间复杂度 = O(n)**；

平均情况：假设目标元素出现在任何一个位置的概率相同，都是 $\frac{1}{n}$

目标元素在第1位，循环1次；在第2位，循环2次；.....；在第 n 位，循环 n 次

$$\text{平均循环次数} = 1 \cdot \frac{1}{n} + 2 \cdot \frac{1}{n} + 3 \cdot \frac{1}{n} + \dots + n \cdot \frac{1}{n} = \frac{n(n+1)}{2} \cdot \frac{1}{n} = \frac{n+1}{2} \rightarrow \text{平均时间复杂度} = O(n)$$

王道考研/CSKAOYAN.COM

12

