石家莊鐵道大學
SHIJIAZHUANG TIEDAO UNIVERSITY

在线开放课程

线性表

顺序表的基本操作
-插入算法

主讲 ： 刘辉

目录

- 1 顺序表的插入算法描述
- 2 顺序表的插入算法实现
- 3 顺序表的插入算法复杂度分析
- 4 顺序表基本算法的调用

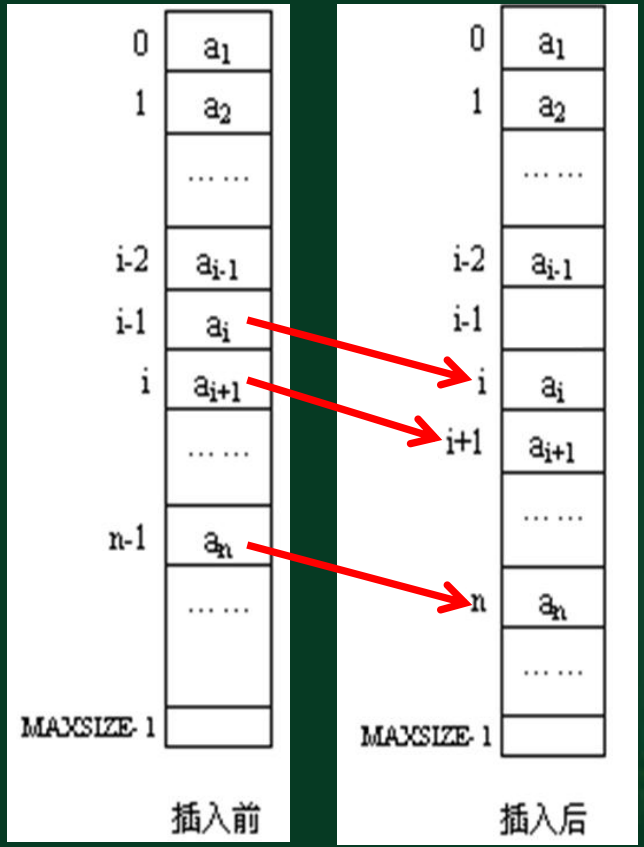
一、顺序表的插入算法

- 顺序表的插入是指在表的第i个位置上插入一个值为x的新元素。i的取值范围为 $1 \leqslant i \leqslant n$

① 将a_i～a_n顺序向下移动，为新元素让出位置；

② 将x 置入空出的第i个位置；

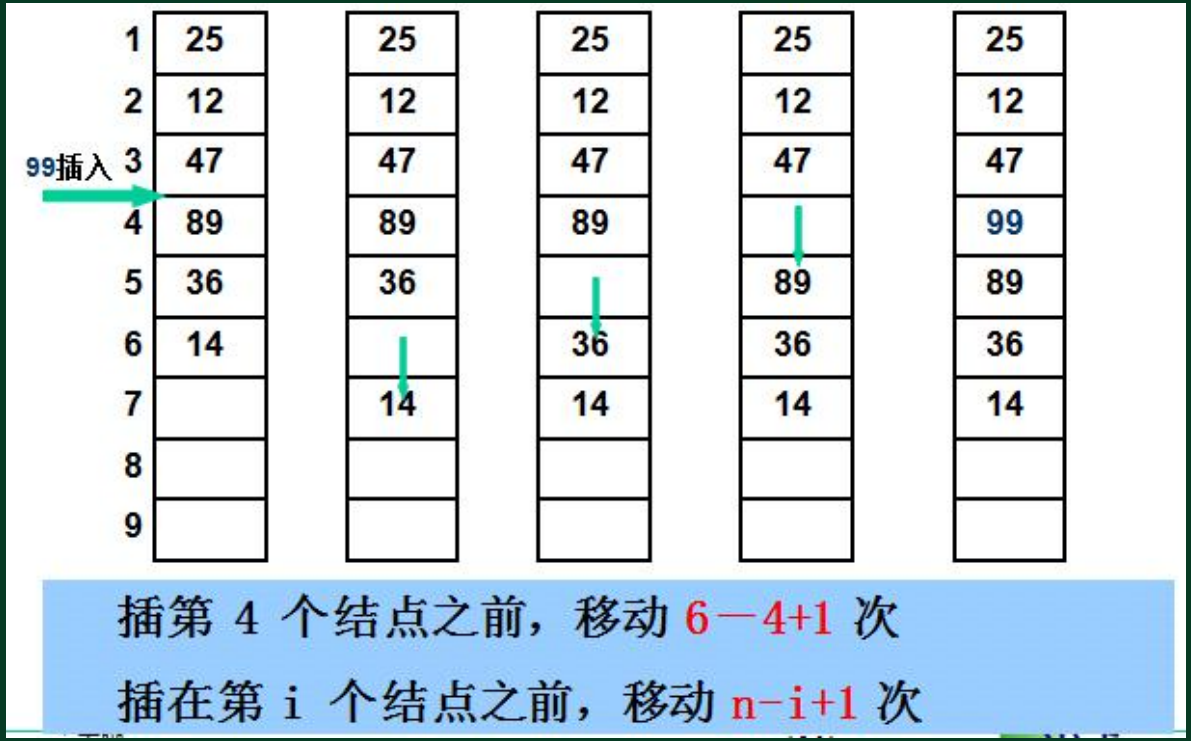
③ 表长值+1。



一、顺序表的插入算法

- **插入算法的示例**

在线性表的第4个结点前插入数据元素99



二、顺序表的插入算法实现


```
#define  MAXSIZE 100
typedef  struct {
     ElemType  *elem;
     int  length
 }SqList;
```

- 插入算法实现
  - 第i个位置插入数据元素e

```
Status ListInsert_Sq(SqList &L, int i ,ElemType e){
 if(i<1 || i>L.length+1) return ERROR;     //i值不合法
 if(L.length==MAXSIZE) return ERROR; //当前存储空间已满
 for(j=L.length-1;j>=i-1;j--)
     L.elem[j+1]=L.elem[j];    //插入位置及之后的元素后移
 L.elem[i-1]=e;                //将新元素e放入第i个位置
 ++L.length;                   //表长增1
 return OK;
}
```

三、顺序表的插入算法复杂度分析

- 算法时间主要耗费在移动元素的操作上

•若插入在尾结点之后，则根本无需移动（特别快）；

•若插入在首结点之前，则表中元素全部后移（特别慢）；

•若要考虑在各种位置插入（共n+1种可能）的平均移动次数，该如何计算？

$$\mathrm{AMN} = \frac{1}{n+1}\sum_{i=1}^{n+1}(n-i+1) = \frac{1}{n+1}(n+\cdots+1+0)$$

$$= \frac{1}{(n+1)}\frac{n(n+1)}{2} = \frac{n}{2}$$

四、顺序表基本算法的调用

- 如何调用顺序表描述的这些算法？

```
Status InitList_Sq(SqList &L) ;
void DestroyList(SqList &L);
void ClearList(SqList &L) ;
int GetLength(SqList L);
int IsEmpty(SqList L);
int GetElem(SqList L, int i ,ElemType &e);
int LocateELem(SqList L,ElemType e);
Status ListInsert_Sq(SqList &L, int i ,ElemType e);
```

四、顺序表基本算法的调用

- 如何调用顺序表描述的这些算法？

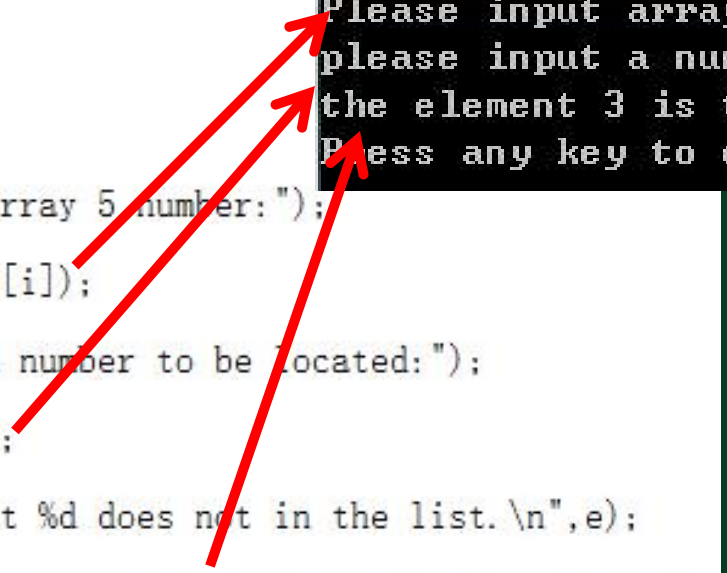
```
typedef struct{
    ElemType *elem;
    int length;
    int listsize;
}sqList;
```

```
void main()
{
    int i,l,e;
    sqList a;
    InitList(&a);

    printf("Please input array 5 number:");
    for(i=0;i<5;i++)
        scanf("%d",&a.elem[i]);
    a.length=i;
    printf("please input a number to be located:");
    scanf("%d",&e);
    l=LocateElemList(&a,e);
    if(l==0)
        printf("the element %d does not in the list.\n",e);
    else
        printf("the element %d is the %dth in the list\n",e,l);

}
```

```
Please input array 5 number:23 45 87 3 12
please input a number to be located:3
the element 3 is the 4th in the list
Press any key to continue
```



四、顺序表基本算法的调用

- 如何调用顺序表描述的这些算法？

```
typedef struct{
    ElemType *elem;
    int length;
    int listsize;
}sqList;
```

```
void main()
{   int i,l,e;
    sqList a;
    InitList(&a);

    printf("Please input array 5 number:");
    for(i=0;i<5;i++)
        scanf("%d",&a.elem[i]);
    a.length=i;

    printf("Input the Insert data:");
    scanf("%d",&e);
    printf("Input the Insert locate:");
    scanf("%d",&i);
    if(InsertList(&a,i,e))
    {   printf("output the datas:");
        for(i=0;i<a.length;i++)
            printf("%d ",a.elem[i]);
        printf("\n");
    }
    else
        printf("Can't insert the data!");
}
```

```
Please input array 5 number:12 5 234 7 12
Input the Insert data:67
Input the Insert locate:3
output the datas:12 5 67 234 7 12
Press any key to continue
```

小结

- 掌握顺序表的插入算法
- 掌握顺序表插入算法的效率分析
- 了解顺序表基本算法的调用方式

谢谢！