计算机与信息工程学院实验报告

••••••••••••••••••••••••••••••••• 密 ••••••••••••••••••••••••••••••••• 封 ••••••••••••••••••••••••••••••••• 线 •••••••••••••••••••••••••••••••••

姓名：朱龙康 学号：1824120002 专业：计算机科学与技术\_ 年级：\_2018级

课程：数据结构 主讲教师：\_袁彩虹\_ 辅导教师：\_\_袁彩虹\_\_

实验时间：\_\_2020\_年 \_3\_月 \_\_10\_日 \_上\_午\_8\_时至\_10\_时，实验地点\_\_\_\_\_\_\_\_

实验题目： 顺序表的操作

实验目的： 掌握顺序栈和链栈的基本操作，循环队列和链队列的基本操作，为后续学习打下基础，以便更好地达成后续更高层次的课程目标。

实验环境（硬件和软件） CodeBlocks

实验内容：

（*1）编程实现顺序表的基本操作：建立顺序表，修改顺序表，插入顺序表，删除顺序表；*

*（2）采用顺序表结构编程实现：两个集合的运算：交集/并集/差集。*

实验步骤：

1）**顺序表的建立和基本操作**

//----顺序表存储结构

struct SqList{

int\*elme;//定义存储空间基地址

int length;//当前元素个数

}l;

//初始化

bool InitList(SqList &l){

l.elme=new int[MAXSIZE];//分配存储空间

if(!l.elme)return false;//分配失败

l.length=0;//元素个数为0

return true;

}

//取值

bool getelem(SqList l,int i,int &e){

if(i>l.length||i<1)return false;//i是否合法

e=l.elme[i-1];//将第i个取得赋值给e，但不是删除该元素

return true;

}

//查找

int LocateElem(SqList l,int e){

for(int i=0;i<l.length;i++){//遍历顺序表

if(l.elme[i]==e)return i+1;//找到，返回元素下标加1

}

return 0;

}

//插入

bool Insert(SqList &l,int i,int e){//将新元素出入到第i个位置

if(i<0||i>l.length+1)return false;//判断是否合法 i共有length+1种查法

if(l.length==MAXSIZE)return false;//顺序表已满

for(int j=l.length-1;j>=i-1;j--)//从后向前遍历 每遍历到一个，便将其向后移动 移动第i-1个元素后停止

l.elme[j+1]=l.elme[j];//

l.elme[i-1]=e;//插入到第i-1个位置

l.length++;//长度加1

return true;

}

//删除 第i个元素

bool Delete(SqList &l,int i){

if(i<1||i>l.length)return false;//i不合法 长度为length 共有length元素可删除

for(int j=i;j<=l.length-1;j++)//从第i个开始遍历 将第i个移到第i-1的位置 然后依次后移

l.elme[j-1]=l.elme[j];//

l.length--;//长度减1

return true;

}

2）**顺序表的交，差，并，操作**

//求并集

SqList And(SqList m1,SqList m2){

for(int i=0;i<m2.length;i++){//遍历m2

if(LocateElem(m1,m2.elme[i])==NULL){//如果m1中不含有m2中的该元素

if(m1.length!=MAXSIZE) Insert(m1,m1.length+1,m2.elme[i]);//如果m1没有满 则从顺序表最后位置插入m1的该元素

else exit(0);}//如果满了 则表明MAXSIZE长度的顺序表存不下该并集 终止

}

return m1;//返回并集结果

}

//求交集

SqList Intersection(SqList m1,SqList m2){

SqList tem;//建立顺序表保存交际结果

InitList(tem);//初始化该表

for(int i=0;i<m2.length;i++){//遍历表m2

if(LocateElem(m1,m2.elme[i])!=0)//如果m1中含有m2中的这个元素

Insert(tem,tem.length+1,m2.elme[i]);//将这个元素插入到顺序表最后位置

}

return tem;//返回交集结果

}

//求差

SqList Subtraction(SqList m1,SqList m2){

for(int i=0;i<m2.length;i++){//遍历表m2

if(LocateElem(m1,m2.elme[i])!=NULL)//如果m1中含有m2中的这个元素

Delete(m1,i+1);//将这个元素从m1中删除

}

return m1;//返回差结果

}

//遍历顺序表

void Printsql(SqList sl){

for(int i=0;i<sl.length;i++){

cout<<sl.elme[i]<<" ";

}

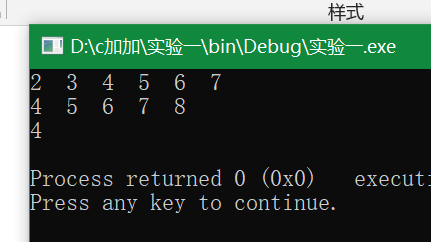
cout<<endl;

return;

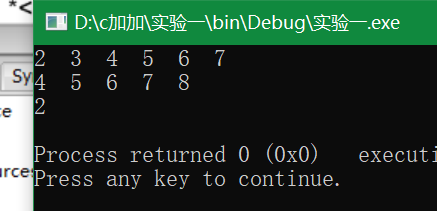
}

实验数据记录：

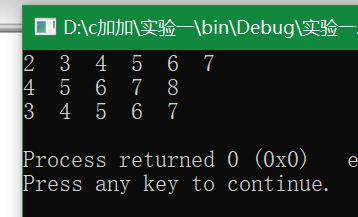
1. 取第一个元素给a



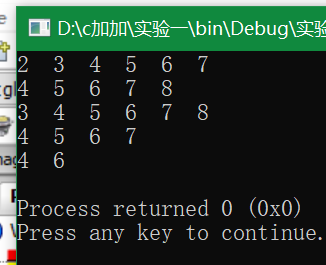
2.查找3的位置



3.删除第1个元素



4.求 交，并，差



问题讨论：