**东莞城市学院**

**实 验 报 告**

**课程名称：** 数据结构

**实验名称：** 顺序表的基本操作

**学 院：** 计算机与信息学院

**姓 名：** 张宇航

**学 号：** 202035020145

**专业班级：** 2020级华为创新班

**指导教师：** 彭 刚

2021 **年** 09 **月** 14 **日**

**实验报告成绩评定表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 评定项目 | 内 容 | 满 分 | 评 分 | 总 分 |
| 实验态度 | 实验认真，态度端正，遵守纪律，出勤情况。 | 20 |  |  |
| 实验过程 | 按实验要求完成各种功能或操作，代码书写规范，注释清晰，设计严谨，运行结果正确。 | 40 |  |
| 报告撰写 | 报告字迹整洁、内容丰富、条理清楚；图、表、文字表达准确规范，上交及时。 | 40 |  |
| 总成绩 | 采用五级分制：优、良、中、及格、不及格 |  | | |
| **评语：**  **指导老师： 年 月 日** | | | | |

实验报告正文

1. **实验目的**

1．掌握顺序存储结构的特点。

2．掌握顺序存储结构的常见算法。

1. **实验内容及结果**
2. 实验内容

编写一个完整的程序，实现顺序表的生成、插入、删除、输出等基本运算。

（1） 实现顺序表的初始化。

（2） 输出顺序表的所有元素。

（3） 在顺序表的位置 i 插入元素 e。

（4） 删除顺序表中指定位置 i 的元素。

（5） 实现把该表中所有奇数排在偶数之前，即表的前面为奇数，后面为偶数，要求算法的时间 复杂度为O(n)，空间复杂度为 O(1) 。

（6） 在主函数中测试上述算法。

1. 源程序

*#include* <stdio.h>

*#include* <stdlib.h>

*#include* <string.h>

*#include* <math.h>

*#define* OK 1

*#define* ERROR 0

*#define* MAXSIZE 10

*typedef* *int* Status;

*typedef* *int* ElemType;

*typedef* *struct* {

    ElemType \*elem;

*int* length;

}SqList;

// 初始化顺序表

Status initList(SqList *&*L) {

    L.elem = *new* ElemType[MAXSIZE];

*if*(!L.elem){

        printf("顺序表初始化错误!!!\n");

*return* ERROR;

    }

    L.length = 0;

*return* OK;

}

//输出顺序表的全部元素

*void* displayList(SqList L) {

*int* i;

*for* (i = 0; i < L.length; i++) {

        printf("%d ", L.elem[i]);

    }

    printf("\n");

}

// 在顺序表的位置 i 插入元素 e

Status ListInsert\_Sq(SqList *&*L,*int* i ,ElemType e){

*int* j;

*if*(i > MAXSIZE || i < 1 || i > L.length + 1){

        printf("插入位置错误!!!\n");

*return* ERROR;

    }

*for*(j = L.length - 1;j >= i - 1;j--){

        L.elem[j + 1] = L.elem[j];

    }

    L.elem[i-1] = e;

    L.length++;

*return* OK;

}

// 删除顺序表中指定位置 i 的元素

Status ListDelete\_Sq(SqList *&*L,*int* i){

*int* j;

*if*(j > MAXSIZE || i < 1 || i > L.length + 1){

        printf("删除位置错误!!!\n");

*return* ERROR;

    }

*for*(j = i;j <= L.length - 1;j++){

        L.elem[j - 1] = L.elem[j];

    }

    L.length--;

*return* OK;

}

// 表中所有奇数排在偶数之前

*void* OddEvenSplit(SqList *&*L){

*int* i = 0,j = L.length - 1;

*int* temp;

*while*(1){

*while*(L.elem[i]%2 != 0){

            i++;

        }

*while*(L.elem[j]%2 == 0){

            j--;

        }

*if*(i >= j){

*break*;

        }*else*{

            temp = L.elem[i];

            L.elem[i] = L.elem[j];

            L.elem[j] = temp;

        }

    }

}

*int* main() {

*int* i;

    SqList sq;

    initList(sq);

    //向顺序表中添加元素

*for* (i = 1; i <= MAXSIZE; i++)

        ListInsert\_Sq(sq,i, rand()%100+1);

    printf("顺序表中的元素分别是：\n");

    displayList(sq);

    printf("删除第 4 个元素后的结果：\n");

    ListDelete\_Sq(sq,4);

    displayList(sq);

    printf("插入第 6 个元素后的结果：\n");

    ListInsert\_Sq(sq,6,99);

    displayList(sq);

    printf("表中所有奇数排在偶数之前的结果：\n");

    OddEvenSplit(sq);

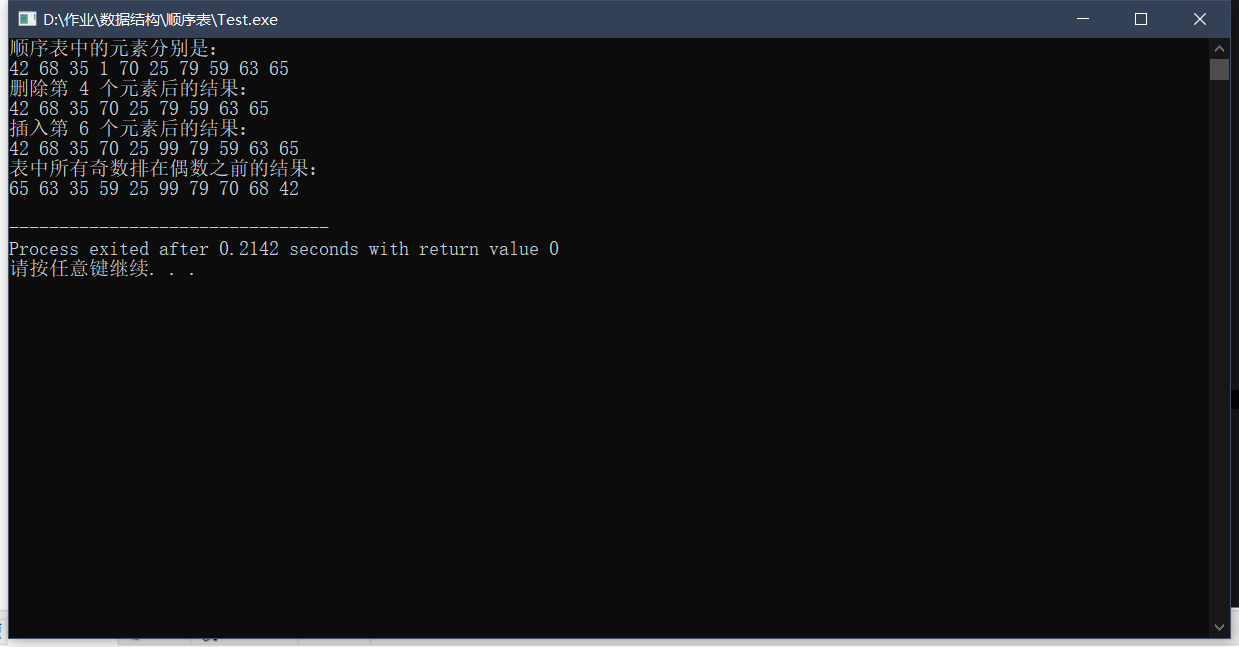
    displayList(sq);

    free(sq.elem);

*return* 0;

}

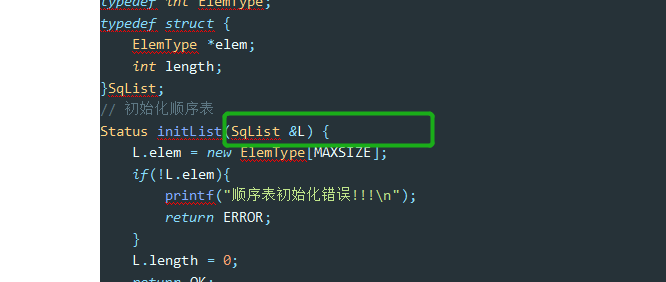
1. 上机实验截图



1. **总结及体会**

（本次实验遇到的主要问题及解决办法，实验体会及收获。）

本次作业是实现顺序表的基本操作,遇到的问题只有刚开始看实验报告的时候,由于头文件导包用的是gcc,以为是要用C语言编写,没想到复制进去报错了。后来在DEV中将文件改成C++就好了。



因为C语言只有值传递和指针传递,而C++除了C语言这两种传递方式之外还有引用传递第三种方式,所以使用C语言进行引用传递会报错。

这次作业对我最大的帮助就是激起了我学习C++的兴趣,可是我是学Java和Go的,没有空闲时间学习C++,所以希望老师平时可以带着我们多学习一点C++的语法吧。