**暨南大学本科实验报告专用纸**

课程名称 数据结构实验 成绩评定

实验项目名称 线性表的操作 指导教师 刘波、郭科芩

实验项目编号 8060154903 实验项目类型 综合性 实验地点 N116

学生姓名 阮炜霖 学号 2020101603

学院 信息科学技术学院 系 计算机系 专业 网络工程

实验时间 2021 年 9 月 9 日 上 午～ 9 月 9 日 上 午

**（一）实验目的和要求；**

实验目的：性表的基本操作在顺序或链式存储结构上的实现。

实验要求：任选一种高级程序语言编写源程序，并调试通过，测试正确。

**（二）实验主要内容；**

1. 建立n个元素的顺序或链式表 ，实现顺序或链式表的基本操作；
2. 在线性表中元素i之后插入一个元素，实现顺序或链式表插入的基本操作；
3. 在线性表中删除指定位置i上的元素，实现顺序或链式表删除的操作。

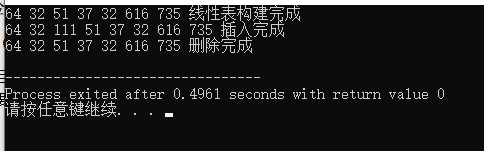
**（三）实验原理**

通过定义一个抽象数据类型，实现线性表的创建、插入和删除等基本操作。

**（四）实验步骤及调试分析**

首先定义出抽象数据类型，然后分配适当的空间，随后逐步实现initlist\insert\del三种类型的函数，完成实验操作。

1. **实验结果及分析；**



通过结果可以得知，线性表能够正常地完成创建、插入和删除等操作，通过测试。

**（六）附录：源程序**

#include<bits/stdc++.h>

#define LIST\_INIT\_SIZE 100 //线性表存储空间的初始分配量

#define LISTINCREMENT 10 //线性表存储空间的分配增量

#define ERROR (cout<<"溢出"<<endl),0

using namespace std;

struct List{

int \*elem,\*newbase;

int length;

int listsize;

void initlist(){

elem=(int \*)malloc(100\*sizeof(int));

if(!elem) exit(OVERFLOW);

length=0;

listsize=LIST\_INIT\_SIZE;

}

int insert(int x,int pos){ //在pos后插入x元素

if(pos<1||pos>length+1) return ERROR;

if(length>=listsize){

newbase=(int \*)realloc(elem,(listsize+LISTINCREMENT)\*sizeof(int));

if(!newbase) exit(OVERFLOW);

elem=newbase;

listsize+=LISTINCREMENT;

}

int q = pos-1;

for(int p = length-1;p>=q;--p) elem[p+1]=elem[p];

elem[q]=x;

++length;

}

int del(int pos){ //删除pos位置的元素

if((pos<1)||(pos>length)) return ERROR;

int p=pos;

int q=length-1;

for(int q=p;q<=length-1;q++) elem[q]=elem[q+1];

--length;

}

};

int a[]={532,12,64,32,51,37,32,616,735,111};

signed main(){

List L;

L.initlist(); //建立链表

for(int i=1;i<=8;i++) L.insert(a[i],i);

for(int i=1;i<L.length;i++){

cout<<L.elem[i]<<" ";

}

cout<<"线性表构建完成"<<endl;

L.insert(a[9],4); //在第4个位置上插入a[6]

for(int i=1;i<L.length;i++){

cout<<L.elem[i]<<" ";

}

cout<<"插入完成"<<endl;

L.del(3); //删除第三个位置的数

for(int i=1;i<L.length;i++){

cout<<L.elem[i]<<" ";

}

cout<<"删除完成"<<endl;

return 0;

}

**暨南大学本科实验报告专用纸(附页)**