武汉理工大学理学院数学系课程实验报告

**课 程 名 称： 数据结构**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **班级** | **信计2001** | **日 期** | **2022/9/23** | **成绩评定** |  |
| **姓名** | **石一佐** | **实验室** | **理学院104** | **老师签名** |  |
| **实验名称** | **针对链式或顺序存储的线性表实现指定的操作** | | | | |
| **所用软件** | VScode | | | | |
| **实 验 目 的**  **及**  **内**  **容** | **一、实验目的**  1.熟悉C语言的上机环境，进一步掌握C语言的结构特点。  2.掌握线性表的链式存储结构——单链表的定义及各种基本操作。  **二、实验内容**  编写一个程序linklist.cpp,实现单链表的各种基本运算和建表算法;  并在此基础上设计一个主程序exp1.cpp顺次完成以下功能：  （1）采用头插法建表：1 3 2 3 4 10 5 7 8 5 9；  （2）输出单链表；  （3）输出单链表的长度；  （4）删除单链表的第5个元素；  （5）在第7个位置上插入元素4；  （6）删除单链表中元素值最大的结点；  （7）输出单链表。 | | | | |
| **实**  **验**  **原**  **理**  **、**  **步**  **骤** | **三、实验步骤**  （1）**linklist.cpp程序中包含如下函数：**  **Initlist(LinkNode \*&L); //初始化单链表L，建立一个带头结点的单链表**  **ListLength(LinkNode \*L); //返回单链表L的长度**  **ListInsert(LinkNode \*&L,int i,ElemType e); //在单链表L的第i个位置上插入元素e**  **ListDelete(LinkNode \*&L,int i,ElemType &e);//删除单链表L的第i个元素**  **DispList(LinkNode \*L);//输出单链表L**  **CreateListT(LinkNode \*&L，ElemType A[]，int n);//头插法建表L**  **DelMaxNode(LinkNode \*&L);//删除单链表中元素值最大的结点**  **（2）主程序**exp1.cpp  执行以上函数实现功能（1）-（7） | | | | |
| **实**  **验**  **结**  **果**  **及**  **分**  **析** | * 通过运行程序得到的结果如下：   结果均符合预期   * 所编写的程序如下：   linklist.cpp   #include<iostream>  #include<malloc.h>  using namespace std;  typedef struct LNode  {      int data;      struct LNode \* next;  } LinkNode;  void Initlist(LinkNode \*&L);  unsigned int ListLength(LinkNode \*L);  bool ListInsert(LinkNode \*&L,int i,int e);  bool ListDelete(LinkNode \*&L,int i,int &e);  void DispList(LinkNode \*L);  void CreateListT(LinkNode \*&L, int A[], int n);  int DelMaxNode(LinkNode \*&L);  void Initlist(LinkNode \*&L)  {      L = (LinkNode \*)malloc(sizeof(LinkNode));      L->next = NULL;  }  unsigned int ListLength(LinkNode \*L)  {      int i=0;      LinkNode\* p=L;      while(p->next!=NULL)      {          i++;          p = p->next;      }      return i;  }  bool ListInsert(LinkNode \*&L,int i,int e)  {      int j=0;      LinkNode\* p=L,\*s;      if(i<=0) return false;      while(j<i-1 && p!=NULL)      {          j++;          p = p->next;      }      if (p==NULL) return false;      s = (LinkNode \*) malloc(sizeof(LinkNode));      s->data = e;      s->next = p->next;      p->next = s;      return true;  }  bool ListDelete(LinkNode \*&L,int i,int &e)  {      int j=0;      LinkNode \* p = L, \*q;      if (i<=0) return false;      while(j<i-1 && p!=NULL)      {          j++;          p = p->next;      }      if (p==NULL) return false;      q=p->next;      e=q->data;      p->next = q ->next;      free(q);      return true;  }  void DispList(LinkNode \*L)  {      LinkNode\* p=L->next;      while(p->next!=NULL)      {          cout<< p->data<<" -> ";          p = p->next;      }      cout<<p->data<<endl;  }  void CreateListT(LinkNode \*&L, int A[], int n)  {      LinkNode \*s;      Initlist(L);      for (int i=0; i<n; i++)      {          s = (LinkNode\*) malloc(sizeof(LinkNode));          s->data = A[i];          s->next = L->next;          L->next = s;      }  }  int DelMaxNode(LinkNode \*&L)  {      int i=2,nowmax=L->next->data,argmax=1, nowval, res;      LinkNode \* p=L->next->next;      while(p->next != NULL)      {          nowval = p->data;          if (nowval>nowmax)          {              nowmax = nowval;              argmax = i;          }          i++;          p = p->next;      }      ListDelete(L, argmax, res);      return res;  }   exp1.cpp   #include "linklist.cpp"  int main()  {      int a[]={1,3,2,3,4,10,5,7,8,5,9};      LinkNode\* L;      // 1      cout<<"1. Create a linktable using head-insert-method"<<endl<<endl;      CreateListT(L,a,11);      // 2      cout<<"2. Display the list: "<<endl;      DispList(L);      cout<<endl;      // 3      cout<<"3. List length: "<<ListLength(L)<<endl<<endl;      // 4      int e,pos =5;      if (ListDelete(L, pos, e)) cout<<"4. Delete "<<e<<" at the position of "<<pos<<":"<<endl;      else cout<<"[Warning] Couln't delete it!"<<endl;      DispList(L);      cout<<endl;      // 5      e=4,pos =7;      if (ListInsert(L, pos, e)) cout<<"5. Insert "<<e<<" at the position of "<<pos<<":"<<endl;      else cout<<"[Warning] Couln't insert it!"<<endl;      DispList(L);      cout<<endl;      // 6      e = DelMaxNode(L);      cout<<"6. Delete node value "<<e<<" as the max value node:"<<endl;      DispList(L);      cout<<endl;  }  （写不完时，可另附稿纸。） | | | | |

备注：不交此报告者，本次实验为“不合格”。