浙江水利水电学院

实验报告

（18-19学年 第2学期）

|  |  |
| --- | --- |
| 课程名称： | 数据结构与算法 |
| 班 级： | 软工S18-1 |
| 学 号： | 2018b31007 |
| 姓 名： | 曹愉 |
| 实验室（中心）名 称： | 软件实施分室(一) |
| 教学单位： | 信息工程与艺术设计学院 |

2019 年 3 月 18 日

实 验 名 称：链表的存储结构及运算 指导教师：寿焕君

实 验 日 期：2019.3.18 地 点：现北306

同组学生姓名：无

|  |
| --- |
| 实验内容及要求：  实验内容：   1. 创建单向链表 2. 分别编写在带头结点单链表中插入结点、删除结点的函数 3. 编写获得两个带头结点单链表的交集函数 4. 编写主函数进行测试   实验要求：   1. 掌握线性表的链式存储结构和操作 2. 掌握在带头结点单链表中插入结点、删除结点的操作 3. 熟练掌握运用带头结点链表表示特定形式的数据的方法，并设计出有关算法 4. 会根据具体的需求问题采用合适的存储方法 5. 掌握调试程序的方式，培养编程的耐心 |
| 主要仪器名称及型号：   1. 计算机：I5-2400/IntelH61/4G DDR3/500G SATA/HD7350 1GB显存独立显卡 2. 开发工具：Cfree |
| 实验过程：（可附页）  **任务1：库文件的使用**  （1）打开库文件“slnklist.h”，熟悉链表中结点的定义。  （2）弄懂库文件中利用头插法和尾插法建立带头结点单链表的关键代码。  **任务2：单链表中结点的删除操作（保存文件为：学号-1.c）**  编写函数void delx(linklist head, datatype x)，删除带头结点单链表head中第一个值为x 的结点，并构造测试用例进行测试。  **任务3：单链表中结点的插入操作（保存文件为：学号-2.c）**  假设带头结点的单链表head是升序排列的，设计算法函数void insert(linklist head,datatype x)，将值为x的结点插入到链表head中，并保持链表有序性。分别构造插入到表头、表中和表尾三种情况的测试用例进行测试。  **任务4：两个带头结点单链表的交集（保存文件为：学号-3.c）**  设计一个算法linklist interSection(linklist L1,linklist L2)，求两个单链表表示的集合L1和L2的交集，并将结果用一个新的带头结点的单链表保存并返回表头地址。  **任务5：成果提交**  实验报告保存文件为：**学号实验报告-单链表.docx**，连同3个C源程序文件上传到<ftp://192.168.106.99>上。（不要打包） |
| 实验数据记录或图片：（可附页）   1. 任务2：  |  | | --- | | 关键代码：  p = head;  while ((p != NULL) && (p->data != x))  {  q = p;  p = p->next;  }  if (p == NULL)  {  printf("\n值为x的节点不存在!");  }  else  {  q->next = p->next;  printf("\n删除成功！");  free(p);} | | 实验结果：（尽可能小，看得清楚即可） | | 算法的时间复杂度：  O(n) |  1. 任务3：  |  | | --- | | 关键代码：  p = head;  q = p->next;  while ((q != NULL) && (s->data > q->data))  {  p = q;  q = q->next;  }  if (q == NULL)  {  p->next = s;  }  else  {  s->next = p->next;  p->next = s;} | | 实验结果：（分三种情况测试的结果）  分别是表头、表中、表尾插入结果。 | | 思考：插入时分三种情况，写代码时是否需要用选择语句分三种情况考虑？如果是不带头结点的单链表呢，要分几种情况考虑？  插入时分三种情况，但是写代码只要分两种情况即可，因为有头节点，所以插在表头的情况依然是在头节点之后插入，与表中插入属于一种。  不带头节点的单链表的话需要分三种情况，分别是表头、表中、表尾。 |  1. 任务4：  |  | | --- | | 关键代码：  while (p1->next != NULL)  {  while ((p2->next != NULL) && (p1->next->data != p2->next->data)){  p2 = p2->next;}  if (p2->next == NULL){  p2 = L2;}  else{  n = (linklist)malloc(sizeof(Lnode));  m->next = n;  n->data = p2->next->data;  m = n;  p2 = L2;}  p1 = p1->next;  }  m->next = NULL; | | 实验结果：（尽可能小，看得清楚即可） | |
| 实验结论：  本实验共有（ 3 ）小题，已完成（ 3 ）小题，完成的题目经运行测试，结果正确。 |
| 教师评语：  该同学在本次实验过程中学习态度认真，具有很强的观察、分析和解决问题的实验能力，能按时完成本次实验的全部任务。经程序运行测试，实验结果正确。  实验完成情况：优；代码规范：优；实验报告表达能力：好。  成绩：优 批阅教师：寿焕君  2019年3月18日 |