实验2单链表实验

2.1实验目的

1. 熟练掌握线性表的链式存储结构。
2. 熟练掌握单链表的有关算法设计和实现。
3. 根据具体问题的需要，设计出合理的表示数据的链式存储结构，并设计相关算法。

2.2 实验要求

1. 本次实验中的链表结构指带头结点的单链表；
2. 单链表结构和运算定义，算法的实现以库文件方式实现，不得在测试主程序中直接实现；比如存储、算法实现放入文件：linkedList.h
3. 程序运行、测试正确；
4. 实验程序有较好可读性，各运算和变量的命名直观易懂，符合软件工程要求；
5. 程序有适当的注释，程序的书写要采用缩进格式。

2.3 实验任务

编写算法实现下列问题的求解。

1. 将单链表Ｌ中的奇数项和偶数项结点分解开（元素值为奇数、偶数），分别放入新的单链表中，然后原表和新表元素同时输出到屏幕上，以便对照求解结果。

实验测试数据基本要求：

第一组数据：单链表元素为 （1,2,3,4,5,6,7,8,9,10,20,30,40,50,60）

第二组数据：单链表元素为 （10,20,30,40,50,60,70,80,90,100）

1. 求两个递增有序单链表L1和L2中的公共元素，放入新的单链表L3中。

实验测试数据基本要求：

第一组

第一个单链表元素为 （1，3，6，10，15，16，17，18，19，20）

第二个单链表元素为 （1，2，3，4，5，6，7，8，9，10，18，20，30）

第二组

第一个单链表元素为 （1，3，6，10，15，16，17，18，19，20）

第二个单链表元素为 （2，4，5，7，8，9，12，22）

第三组

第一个单链表元素为 （）

第二个单链表元素为 （1，2，3，4，5，6，7，8，9，10）

1. 删除递增有序单链表中的重复元素，要求时间性能最好。

实验测试数据基本要求：

第一组数据：单链表元素为 （1,2,3,4,5,6,7,8,9）

第二组数据：单链表元素为 （1,1,2,2,2,3,4,5,5,5,6,6,7,7,8,8,9）

第三组数据：单链表元素为 （1,2,3,4,5,5,6,7,8,8,9,9,9,9,9）

1. 递增有序单链表L1、L2，不申请新结点，利用原表结点对两表进行合并，并使得合并后成为一个集合，合并后用L1的头结点作为头结点，删除多余的结点，删除L2的头结点。要求时间性能最好。

实验测试数据基本要求：

第一组

第一个单链表元素为 （1，3，6，10，15，16，17，18，19，20）

第二个单链表元素为 （1，2，3，4，5，6，7，8，9，10，18，20，30）

第二组

第一个单链表元素为 （1，3，6，10，15，16，17，18，19，20）

第二个单链表元素为 （2，4，5，7，8，9，12，22）

第三组

第一个单链表元素为 （）

第二个单链表元素为 （1，2，3，4，5，6，7，8，9，10）

1. 已知一个带有表头结点的单链表，结点结构如下图。假设该链表只给出了头指针list。在不改变链表的前提下，请设计一个尽可能高效的算法，查找链表中倒数第k个位置上的结点（k为正整数）。若查找成功，算法输出该结点的data值，并返回1；否则，只返回0。要求：

（1）描述算法的基本设计思想

（2）描述算法的详细实现步骤

（3）根据设计思想和实现步骤，采用程序设计语言描述算法（使用C 或C++语言实现），关键之处请给出简要注释。