**《数据结构与算法》第一次书面作业**

**作业要求：**

**（1）本次作业占总成绩的10%。其中作业中的姓名、班级和学号及程序均要求机打（不收电子作业）。若其中各项手写只按分数一半计（即总成绩的5%）；**

**（2）打分按程序的正确性和质量；**

**（3）本次作业提交日期：2019年10月27日上课时**

**姓名：\_\_\_\_\_杨轶\_\_\_\_\_\_ 班级和学号：\_\_\_\_619231386\_\_\_\_\_\_\_\_\_**

阅读下列程序片段，完成题目的要求

第一题：画出单链表

int n=0; LinkList p,k;

k=(LinkList)malloc(sizeof(struct node));

k🡪next=NULL;

while(n<5)

{ p=(LinkList)malloc(sizeof(struct node));

p🡪data=2\*n+1;

p🡪next= k🡪next;

k🡪next=p;

n++;

}

k🡪data=n;

答：**程序功能**：**用头插法的方式完成一个单向链表的初始化和创建，此单向链表中自始至终都有一个头结点k，k这个头结点的存在是为了方便新创建的节点在插入单向链表时统一插入操作。**

**程序分解步骤图示：**

1. k→

1

∧

θ

p

2. k→

1

∧

3

θ

p

3. k→

1

∧

θ

3

5

p

4. k→

1

∧

3

5

7

θ

p

p

5. k→

5

9

7

5

1

∧

3

第二题:画出单链表并说明程序的功能

int n; LinkList f1,f2,f3,k;

f1=(LinkList)malloc(sizeof(struct node));

f2=(LinkList)malloc(sizeof(struct node));

f1🡪data=f2🡪data=1;

k=f1;

f1🡪next=f2;

for(n=3;n<=10;n++)

{ f3=(LinkList)malloc(sizeof(struct node));

f3🡪data=f1🡪data+f2🡪data;

f2🡪next=f3;

f1=f2;f2=f3;

}

f3🡪next=NULL;

答：**程序功能：斐波拉契数列。用尾插法初始化并构建出单向链表，此单向链表不带头节点。每一个节点为前两个节点的值相加，即f3=f1+f2。由此来构成一个斐波拉契数列。**

**程序图示：**

1

1

2

3

8

5

k→

55

∧

34

21

13

第三题：说明执行下列函数后，操作单链表的作用

void fun(LikList h)

{ int m; LinkList p,q,s;

p=h🡪next;

q=h;

m=p🡪data;

s=q;

while(p!=NULL){

if(p🡪data>m){m=p🡪data;s=q;}

q=p;

p=p🡪next;

}

p=s🡪next;

s🡪next=p🡪next;

free(p);

}

答：**操作单链表的作用为: 删除单向链表当中的值最大的节点。当拥有多个相同的最大值的节点时，删除查找到的第一个节点。**

**程序分析：**

**第一步：定义了三个指针变量p,q,s和一个int类型变量m；**

**第二步：进行走链，走链的目的是为了找寻出单向链表当中的最大值以及最大值所属的节点；**

**第三步：用m记录走链过程中的单向链表节点中的最大值，用s记录最大值所在的节点的上一个节点，用p记录走链找寻出来的最大值所属的节点本身，通过**p=s🡪next;s🡪next=p🡪next;**两行代码完成最终的删除操作删除p所记录的节点，即最大值所在的节点。**

**第四步：通过**free(p);**这行代码，** **完成p从单向链表当中删除完之后，回收此节点的内存空间的动作。**