1、简述以下算法的功能。

Status A(LinkedList L)

{ //L是无表头结点的单链表

if(L && L->next) {

Q=L; L=L->next; P=L;

while(P->next) P=P->next;

P->next=Q; Q->next=NULL;

}

return OK;

}

答案：如果L的长度不小于2，将L的首元结点变成尾元结点。

2.简述以下算法的功能（栈的元素类型SElemType为int）。

status algo1(Stack S)

{

inti,n,A[255];

n=0;

while(!StackEmpty(S)) { n++; Pop(S,A[n]); }

for(i=1;i<=n;i++) Push(S,A[i]);

}

答案：栈中的数据元素逆置

3.简述以下算法的功能

status algo2(Stack S,int e)

{

Stack T; int d;

InitStack(T);

while(!StackEmpty(S))

{

pop(S,d);

if(d!=e) push(T, d);

}

while(!StackEmpty(T))

{

pop(T,d);

push(S,d);

}

}

答案：如果栈中存在元素e，将其从栈中清除

算法题：单链表

1、已知单链表L，链表中第i个元素的值更改为第i-1元素值加上第i个元素的值。

2、已知单链表L，判断数据元素是否是递增的，若是则返回1，否则返回0。

3、已知单链表L，删除第i个结点，假设i是有效的。

4、已知单链表L，若单链表中（第i个元素的值）\*i是递增的，则返回1，否则返回0。