**实验一 线性表的操作**

**一、实验目的**

掌握线性表的基本操作在存储结构上的实现，其中以单链表的操作作为重点。

**二、实验题目**

1.以**单链表**作为存储结构，实现线性表的就地逆置。

2. 创建一个**递增序（有重复值）的单链表**，实现删除值相同的多余结点。

**实验二 栈和队列的操作**

**一、实验目的**

掌握栈和队列的存储结构、操作特性及实现方法。

**二、实验题目**

1.设从键盘输入一个整数序列：a1, a2, …，an，编写程序实现：采用**链栈**结构存储输入的整数，当ai ≠-1时，将ai进栈；当ai=-1时，输出栈顶整数并出栈。

2.设以**不带头结点的循环链表**表示**队列**，并且只设一个指针指向队尾结点，

但不设头指针。编写相应的入队和出队程序。

**程序范例：**以顺序表作为存储结构，实现线性表的就地逆置。

#include<iostream>

using namespace std;

const int MaxSize=100;

class SeqList

{ private:

int data[MaxSize];

int length;

public:

SeqList() { length=0;}

SeqList(int a[],int n);

void print();

void Reverse();

};

SeqList::SeqList(int a[],int n)

{ if(n>MaxSize)throw"参数非法";

for(int i=0;i<n;i++)

{ data[i]=a[i];

}

length=n;

}

void SeqList::print()

{ for(int i=0;i<length;i++)

{ cout<<data[i]<<" ";

}

cout<<endl;

}

void SeqList::Reverse(){

int temp=0;

for(int i=0;i<length/2;i++)

{ temp=data[i];

data[i]=data[length-1-i];

data[length-1-i]=temp;

}

int main()

{ int a[10]={1,2,3,4,5,6,7,8,9,10};

SeqList m(a,10);

m.print();

m.Reverse();

m.print();

}