**实验1 顺序表实现和应用**

**实验目的：**

掌握顺序表的基本操作；

熟悉C++程序设计

**实验内容：**

利用顺序表实现大整数的加法和乘法，并分析算法时间复杂度。

**实验要求：**

1. 顺序表类用C++的模板实现
2. 算法用流程图和伪代码的形式分别描述，并分析算法时间复杂度
3. 输入大数据位数以及每一位的数字（从高位到低位），输出结果位数和每一位的数字（从高位到低位）。大整数位数不超过1000。

输入数据示例如下：

3 1 2 3 //3位数，数值为123

5 1 2 3 4 5 //5位数，数值为12345

加法结果输出数据如下：

5 1 2 4 6 8 //5位数，数值为12468

乘法结果输出数据如下：

7 1 5 1 8 4 3 5 //7位数，数值为1518435

1. 实验报告和源程序（.h和.cpp）请在实验结束一周内上传到网络教学平台

**必做实验：**

大整数加法、大整数乘法：

**选做实验：**

数组奇偶调整：将整型数组A[n]调整为左右两部分，左边所有元素为奇数、右边所有元素为偶数，并且左右两边都是递增顺序。请编写程序实现并分析算法复杂度。

**实验提示：**

**1、模板类的使用（课本P32-37）**

（1）定义：

Template <typename DataType>//模板类的定义，DataType为抽象类型，具体的类型可以在主函数中指定

class SeqList

{

SeqList(DataType a[], int n)//形参类型为DataType类型

DataType Delete(int i)//函数返回值为DataType类型

DataType data[MaxSize]//变量类型为DataType

…

}

（2）使用：

int a[MaxSize],n;

SeqList<int> A(a,n)//指定SeqList类中DataType类型均为int型

int k = Delete(i)//函数返回值DataType类型为int型，可以直接赋值给int变量

**2、程序编写**

（1）顺序表的定义和基本操作的实现，参照课本P32—36；

（2）课本上的子函数中所有变量均未定义，需要自己添加定义；

（3）自主编写主函数；

（4）自主编写加和乘函数（算法参考实验指导P182）

参考定义：将两个SeqList A和B求和或乘积，结果放到SeqList C中返回。

template <class DataType>

SeqList<DataType> Add(SeqList<DataType> A, SeqList<DataType> B)

{ SeqList<int> C;

Int lengthA=A.Length(), lengthB=B.length();

Int len=lengthA;

If(lengthA>lengthB) len=length;

For(i=0;i<len;i++)

C.Insert(i+1,A.Get(i)+B.Get(i));//把A[i]和B[i]相加，结果为C[i]

}

template <class DataType>

SeqList<DataType> Mult(SeqList<DataType> A, SeqList<DataType> B)

{……}

（5）注意类中private变量的访问

（6）注意数据高低位的存储顺序和进位问题，加法、乘法的计算和进位均从低位向高位进行。