**第一章实验报告**

班级：

姓名：

学号：

# 一、实验目的

掌握Visual Studio控制台工程项目的建立；温习C++代码逻辑错误调试；掌握在给定代码中进行适当修改的方法；设计简单的数值计算代码。

# 二、实验内容

1.运行《实验准备(线性表)》程序，读懂程序代码；

2.运行《实验准备(graph邻接矩阵)》程序，读懂程序代码；

3.运行《求多项式A(x) 》程序，完成代码填空。

# 三、实验环境

Windows 10, Visual Studio 2022。

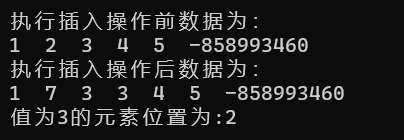
# 四、实验要求

1. 找出《实验准备(线性表)》中的逻辑错误，修改并调试。
2. 对《实验准备(graph邻接矩阵)》，修改输出边的功能：（1）在头文件.h中增加void PutArc();（2）在程序文件.cpp中实现void PutArc(); （3）在主程序中调用 PutArc()。
3. 对《求多项式A(x) 》，写出2种不同的求解代码。

# 五、报告内容

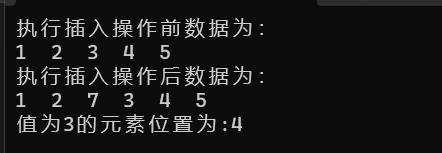
1. **修改语法错误和代码逻辑错误，调试并运行《实验准备(线性表)》代码，完成以下任务。**

**(a) 修改语法错误后运行，给出错误结果截图**



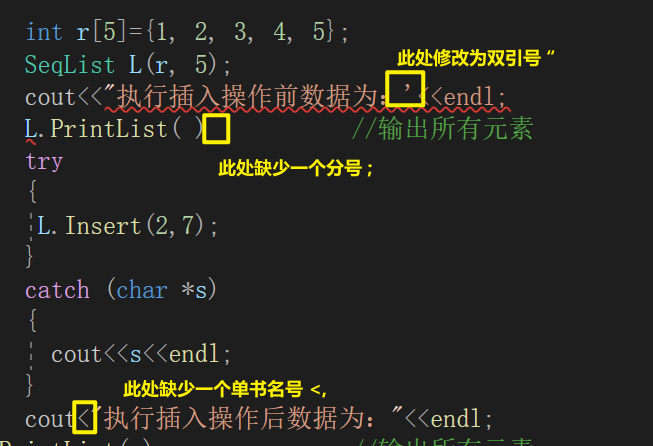
（仅修改main文件语法错误后的输出结果）

**(b) 修改代码逻辑错误，给出最终正确结果截图**

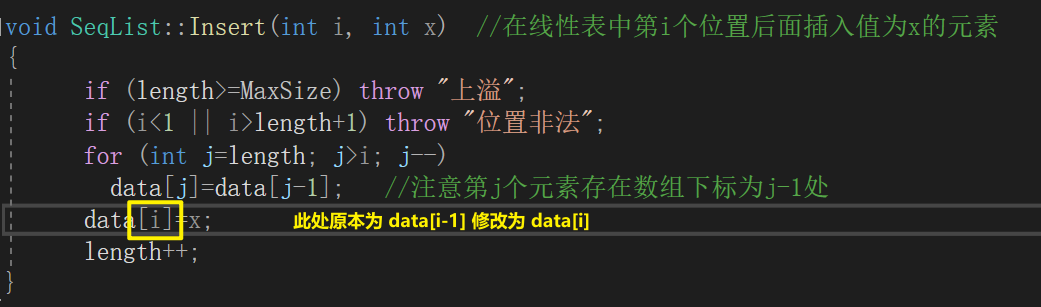


**具体说明对代码的修改：**

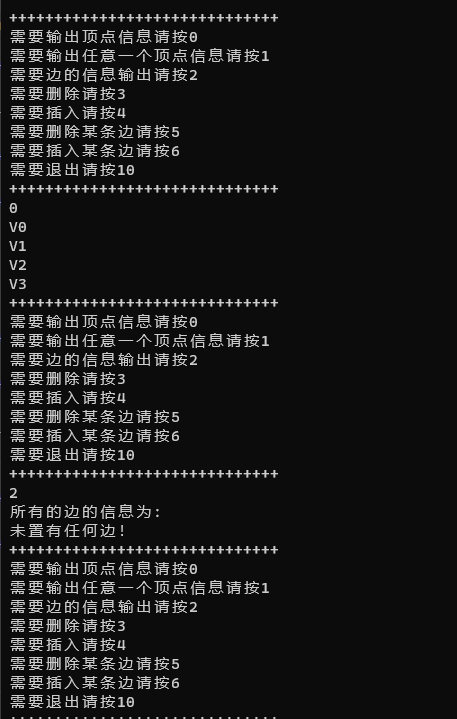
1.main.cpp修改部分



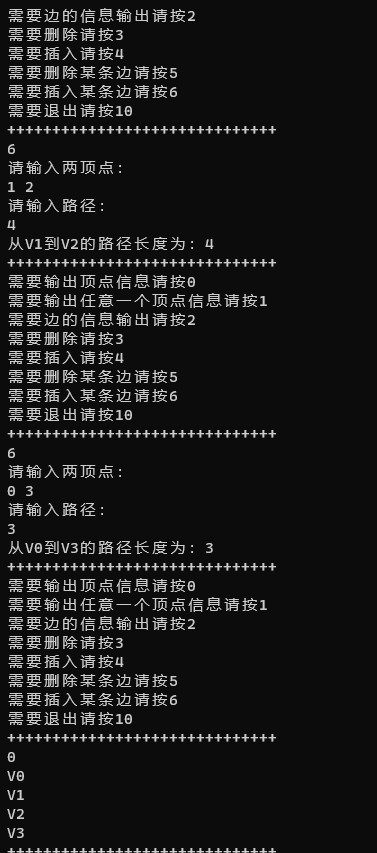
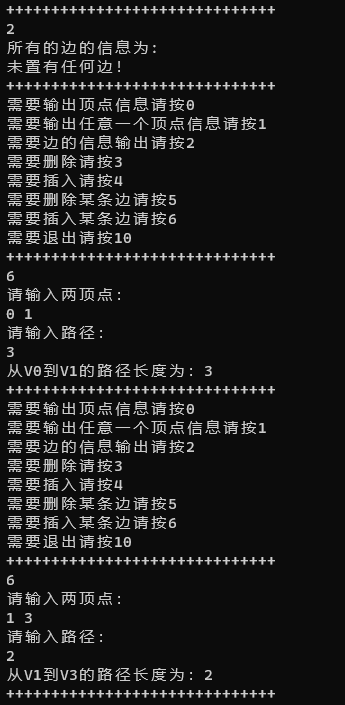
2.Seqlist.cpp修改部分



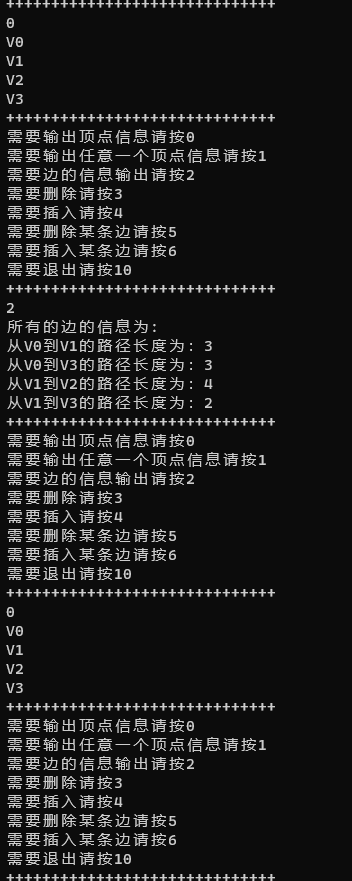
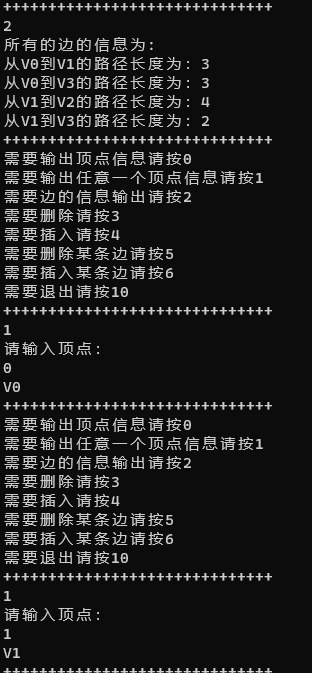
1. **《实验准备(graph邻接矩阵)》，修改输出边的功能，实现PutArc()函数，在已有顶点之间新增一条边，新增一个顶点V5，在新顶点和已有顶点间插入一条边，给出运行截图。**



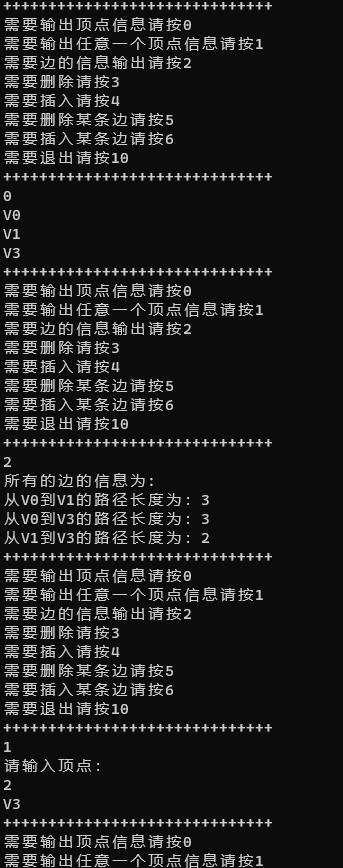
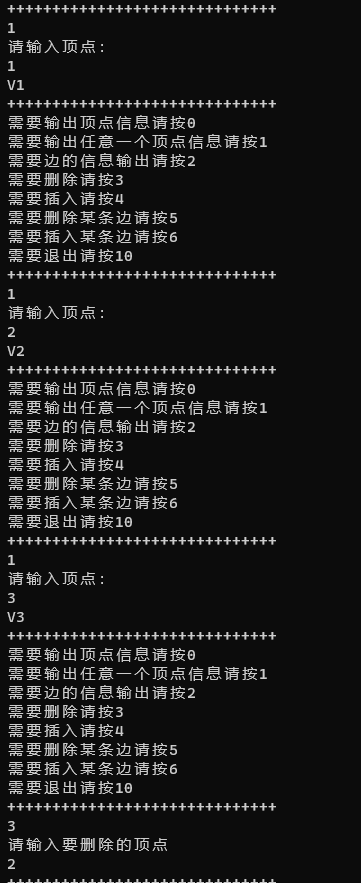
(1)



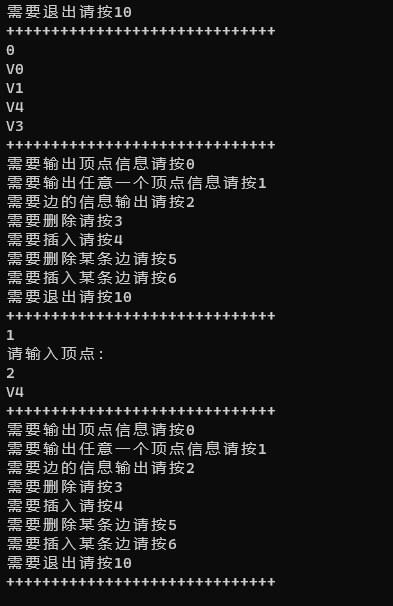
(2) (3)



(4) (5)



(6) (7)



(8)

**void MGraph<T>::PutArc()实验代码：**

template<class T>

void MGraph<T>::PutArc()

{

int i, j;

int flag = 0;

for (i = 0; i < vertexNum; i++) {

for (j = i + 1; j < vertexNum; j++){

if (arc[i][j] != 0) {

cout << "从" << vertex[i] << "到" << vertex[j] << "的路径长度为：" << arc[i][j] << "\n";

flag=1;

}

}

}

if (flag == 0) {

cout << "未置有任何边！"<< "\n";

}

}

**新增顶点和边的实现代码：**

**在main函数中修改如下**

case 7: // 插入新顶点V5

try {

g.InsertVex(4, "V5"); // 在索引4插入

cout << "已插入顶点V5" << endl;

} catch(const char\* msg) {

cout << "错误: " << msg << endl;

}

break;

case 8: // 在V5和V0间插入边

try {

g.InsertArc(4, 0, 7); // 索引4(V5)和0(V0)，权重7

cout << "已插入V5到V0的边" << endl;

} catch(const char\* msg) {

cout << "错误: " << msg << endl;

}

1. **对《求多项式A(x) 》，设置n=10，x=2，并设系数，给出运行截图。**

**第一种实现方式float Ax1(float A[],int n, float x)代码：**

float Ax1(float A[],int n, float x)

{

float result=0.0;

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

{

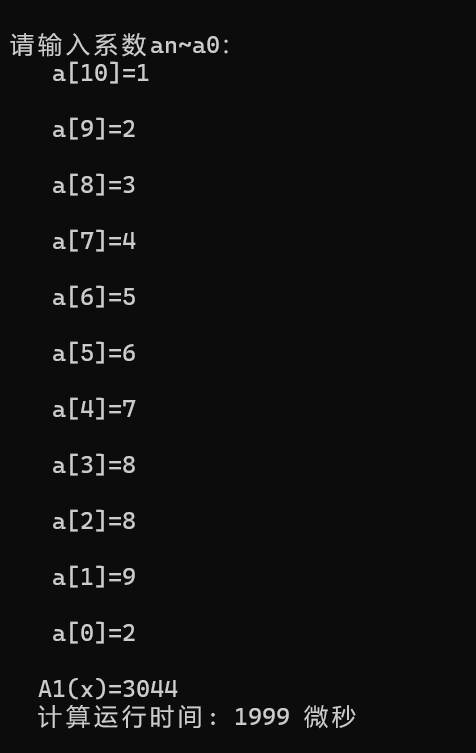
result+= A[i] \* pow(x, i);

}

return result;

}

**第一种实现方式运行截图，给出运行时间：**



**第二种实现方式float Ax2(float A[],int n, float x)代码：**

float Ax2(float A[],int n, float x)

{

float result = 0;

float X = 1;

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

{

result += A[i] \* X;

X = X \* x;

}

return result;

}

**第二种实现方式运行截图，给出运行时间：**

