一、需求分析

二、设计思想：概要设计

三、调试分析

四、测试

五、实验总结/心得

一、需求分析

实现数据结构三元组Triplet及其增删查改

定义并实现两个三元组的加减操作

二、设计思想：概要设计

函数原型：

构造器与析构器：

Status InitTriplet(Triplet &T, ElemType v1, ElemType v2, ElemType v3);

Status DestroyTriplet(Triplet &T);

存取元素：

Status Get(Triplet T, int i, ElemType &e);

Status Put(Triplet &T, int i, ElemType e);

判断是否有序：

Status IsAscending(Triplet T);

Status IsDescending(Triplet T);

取最值：

Status Max(Triplet T, ElemType &e);

Status Min(Triplet T, ElemType &e);

两个三元组加减：#将T1与T2运算，结果赋值给T

Status Add(Triplet &T, Triplet T1, Triplet T2);

Status Minus(Triplet &T, Triplet T1, Triplet T2);

三、调试分析

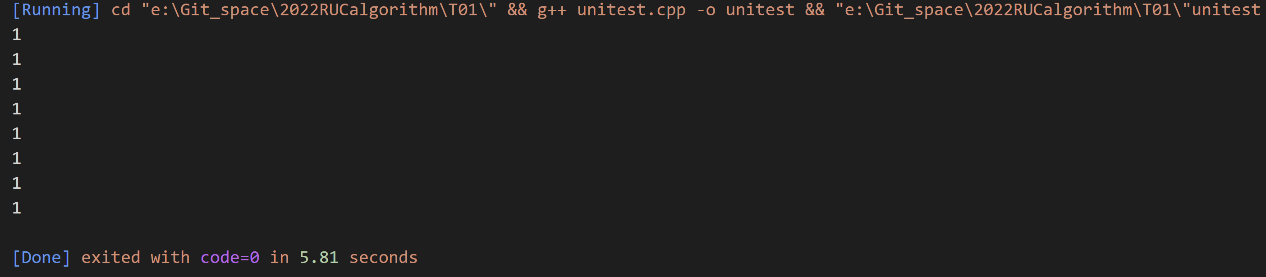
遇到了一个问题：如果在声明Triplet后不调用InitTriplet，后续函数调用则会报错

问题解决：记得在声明后立即初始化。

暂未找到这种实现方式如何自动在声明后调用默认构造器，留待后续学习完善

四、测试

单元测试写在unitest.cpp中，共8个测试点，涉及所有函数



五、实验总结/心得

了解了一个基本数据结构的实现与接口设计