**暨南大学本科实验报告专用纸**

课程名称 高级语言程序设计实验 成绩评定

实验项目名称 指针验证实验 指导教师 张鑫源

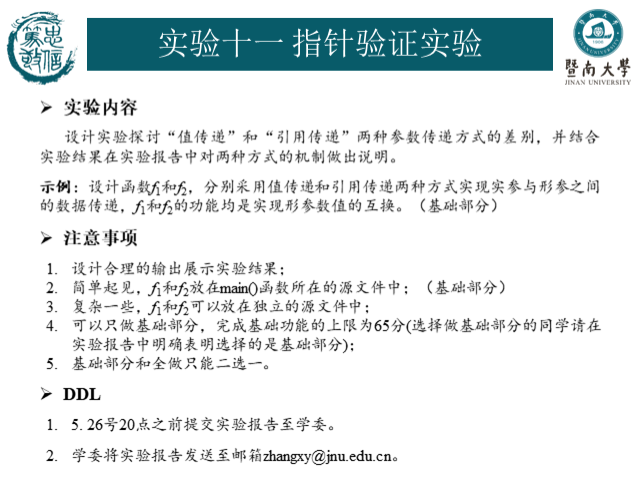
实验项目编号 110 实验项目类型编程实验 实验地点

学生姓名 林晓旭 学号 2019051121

智能工程与科学学院 系 信息安全 专业

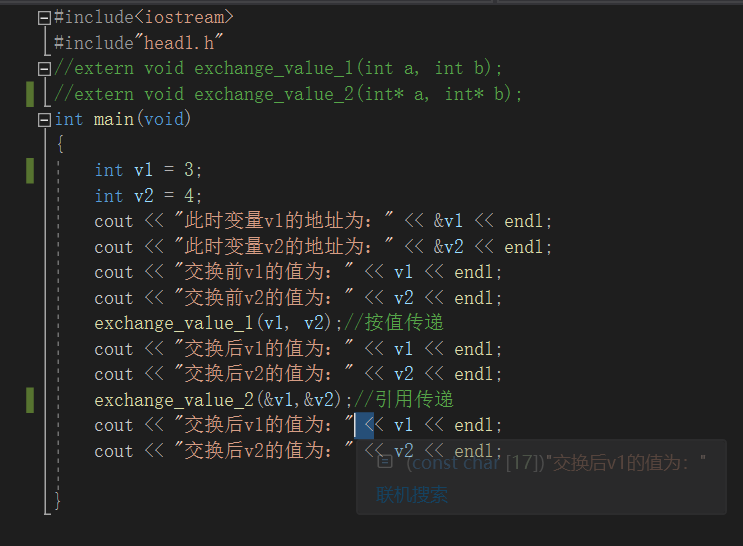
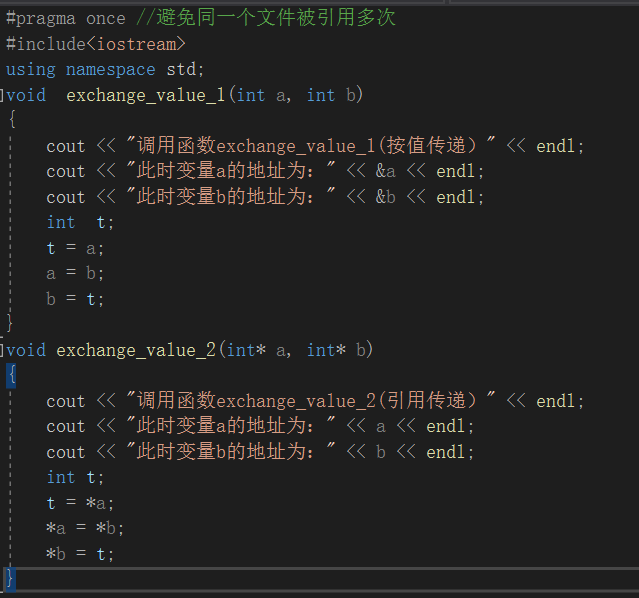
实验时间2020 年—5月20日 上午～5月20日下午 温度 ℃湿度

1. **实验目的： 领会c语言中值传递和引用传递的区别**
2. **实验内容和要求**

****

1. **主要仪器设备：计算机**
2. **源程序**

**1.C2.cpp**

**2.head1.h**

1. **步骤与调试**

**1.在head1.h头文件中定义了两个函数分别是exchange\_value\_1和exchange\_value\_2；exchange\_value\_1是通过按值转递来传递参数，作用是交换形参的值；exchange\_value\_2是通过“引用”传递（指针实现）来传递参数，作用同上。**

**2.在C2.cpp中声明两个变量v1,v2并初始化它们并打印v1，v2的地址。**

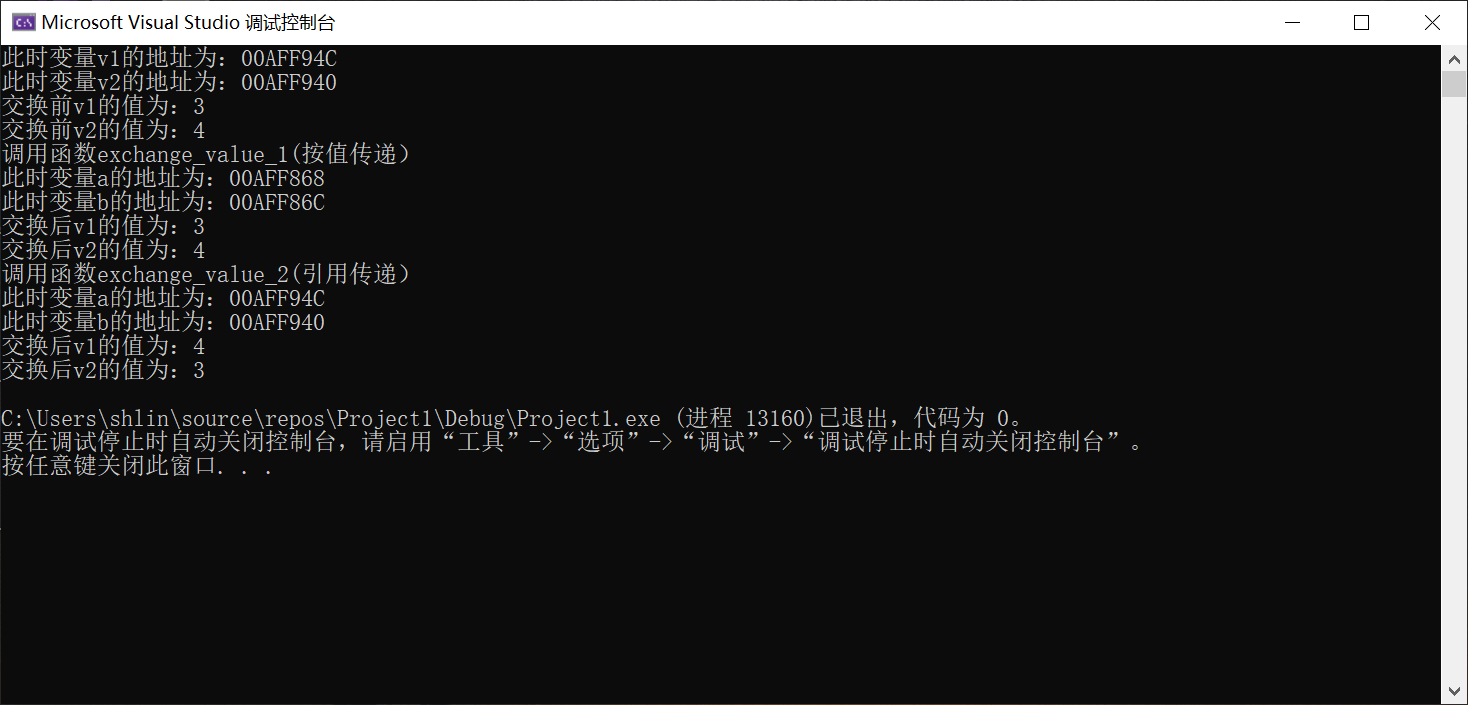
**3.随后调用exchange\_value\_1来尝试交换v1,v2的值，并在exchange\_value\_1函数中打印对应的形参的地址。**

**4.在调用完exchange\_value\_1后打印v1,v2的值。**

**5. 随后调用exchange\_value\_2来尝试交换v1,v2的值，并在exchange\_value\_2函数中打印对应的形参的地址.**

**6. 在调用完exchange\_value\_1后打印v1,v2的值。**

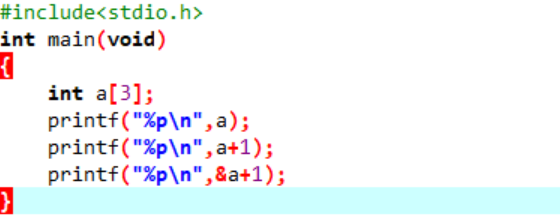
**(六)、实验结果与分析**

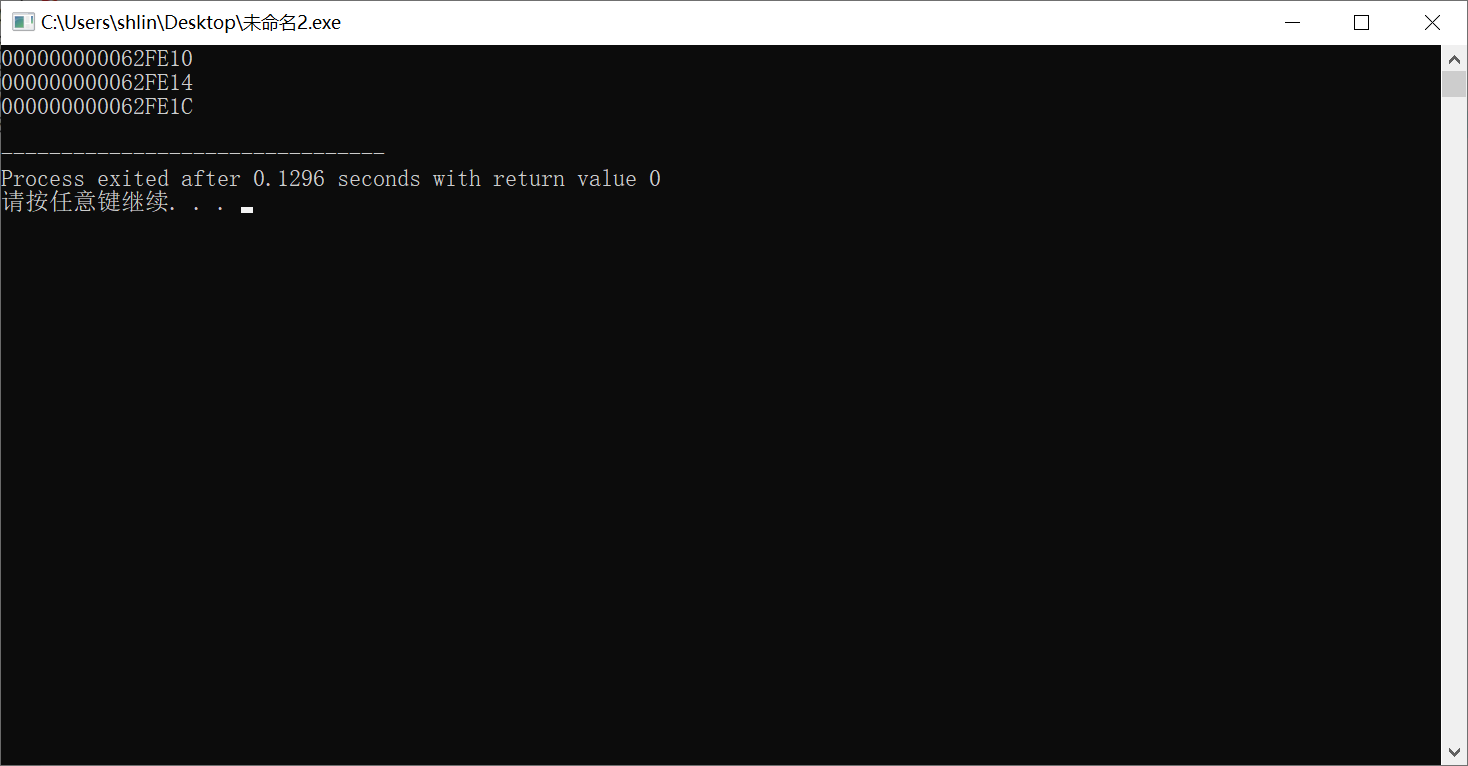
****

**可以发现，在从head1.h调用的exchange\_value\_1中，实参v1,v2与对应的形参a，b的地址不相同，说明在按值传递中，传递的是实参的一份拷贝，传递的是右值；而在exchange\_value\_2中，实参v1,v2与对应的形参a，b的地址相同，说明按引用传递实际上传递的实参的地址，在本程序中通过exchange\_value\_2中的形参指针取接受实参地址，然后用过\*符，来获取实参地址中储存的值并进行运算，所以可以成功的交换在C2.cpp中mian函数中的v1，v2的值。**

**顺便说一下，地址是地址，指针是指针，两个息息相关却有着本质上的不同。地址是一个右值，指针则是储存地址的一种数据类型，有着自己的独立的内存空间。当定义了一个指针int\* p，对p进行取值的操作&p，返回的是储存指针的地址，而不是指针指向的地址，这是值得注意的点。**

**如果定义了一个数组int a[3]，然后对数组名取地址&a会发生什么呢？数组名就是数组的首地址，是一个右值，取一个数组名产生的是一个该数组数据类型的指针常量，其值是该数据的首地址；但是对数组名取地址，产生的是一个指向该数组的指针(例如：(int\*) [5])，并不是指向一个指针常量的指针**

****

****

**通过上述程序可以验证，当我们定义了一个int a[3]数组时，然后进行a+1的操作，通过指针运算的性质可知，会对a的地址加上一个该指针所指的类型的长度（即int类型长度4字节）；当进行&a+1的操作时，发现是对a加上了整个数组的长度，通过指针运算的性质，可以反推出这是是对一个指向长度为2的整型数组的指针。**

**虽然a的值与&a的值相同，但是却有本质上的不同（类型不同）**

**从编译的角度来讲，编译器通过用户是否给出&，来决定指针变量的类型，进而翻译为相应的汇编码。 或者换句话说，&符只是用来表明变量a取地址后得到的值，被看作什么类型的指针，而不是用来表示对a进行取地址操作。**

**暨南大学本科实验报告纸(附页)**