

四川大学电气工程学院

实验报告（本科）



提交人学号 2024141440057

提交人姓名 张美杰

专 业 电气类

日 期 2024/12/5

实验题目 变量的存储类别

组员（按贡献大小排序）：

张美杰

算 法 描 述

实训 1

输入学生学号stuNo
while (stuNo[i] != '\0' && stuNo[i] >= '0' && stuNo[i] <= '9')
k = stuNo[i] - 48 转换为整型数字
number = 10 * number + k 将整数累加到number中
i++
输出转换后的整数

实训 2

依次输入成绩并保存到数组score中:	
for (i = 0; i < 10; i++)	
输入第i个数据	
初始化最大最小的静态变量	
maxScore = minScore = score[0]	
for (i = 0; i < 10; i++)	
score[i] > maxScore?	
是	否
maxScore = score[i]	
score[i] < minScore?	
是	否
minScore = score[i]	
输出最大值maxScore和最小值minScore	

实训 3

请输入20个整数
初始化数组x用于储存数据
for (i = 0; i < 20; i++)
存入第i+1个数据
初始化最大值maxNum最小值minNum
总和sum
for (i = 0; i < 10; i++)
是 score[i] > maxScore? 否
maxScore = score[i]
是 score[i] < minScore? 否
minScore = score[i]
累加得总和sum += x[i]
求平均值ave=sum/n
输出最大值、最小值、平均值

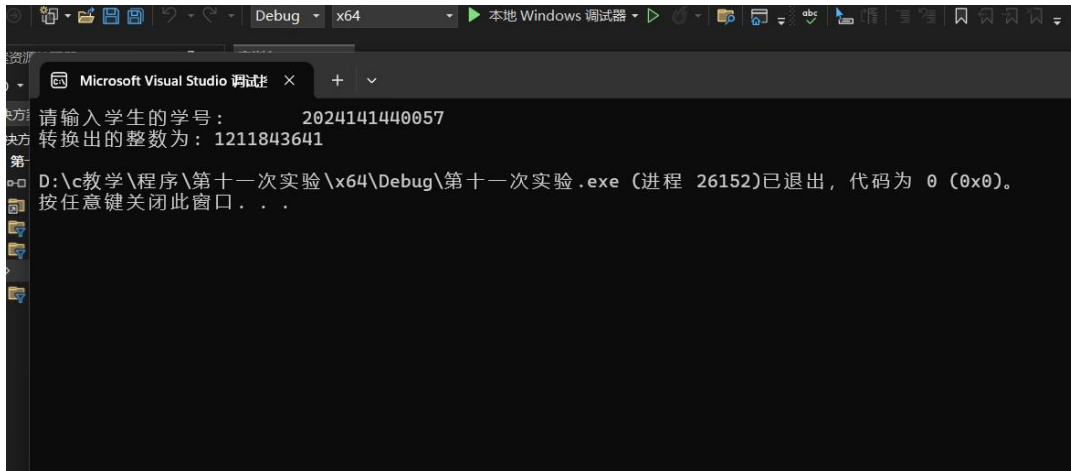
实训 4

初始化最大公约数gysh和最小公倍数qbsh		
输入两个整数: n1, n2		
n1==n2? yes no		
gysh=n1	n1 < n2? yes no	
gbsh=n2	交换n1、n2的值	
for (int i = n1; i > 1; i--)		
!(n2 % i)? yes no		
gysh = i		
break		
for (int i = 1;;i++)		
n2 *= i		
!(n2 % n1)? yes no		
gbsh=n2		
break		
gysh==0? yes no		
gysh = 1		
输出最大公约数为: gysh		
最小公倍数为: gbsh		

实验结果（实训 1~4 的执行界面截图）

实训 1

2)



3)

k“未定义标识符”。

4)

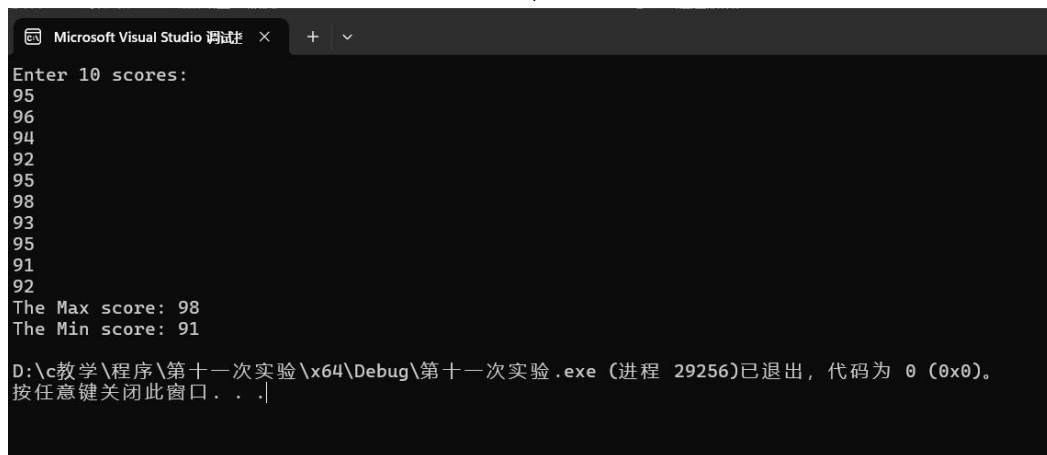
k 起初为较大的-858993460，在执行 15 行后，k 的值依次为输入的数。

5)

k 的作用域只限于 `while (stuNo[i] != '\0' && stuNo[i] >= '0' && stuNo[i] <= '9')` 后的 `{}` 内。

实训 2

2)



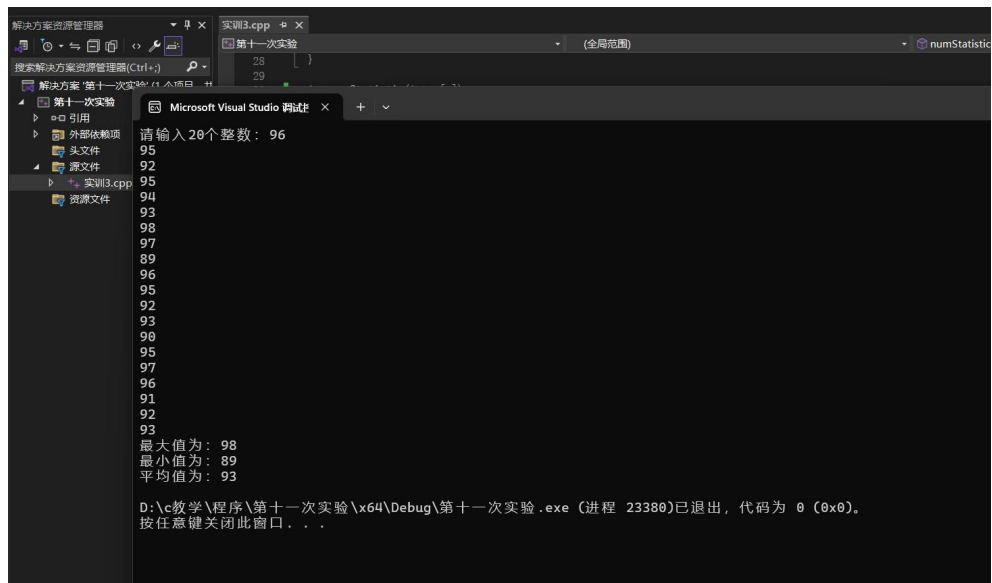
3)

`score`、`minScore` 和 `maxScore` 作用于整个主程序，`score` 是静态全局数组，`minScore` 和 `maxScore` 是静态全局变量。

4)

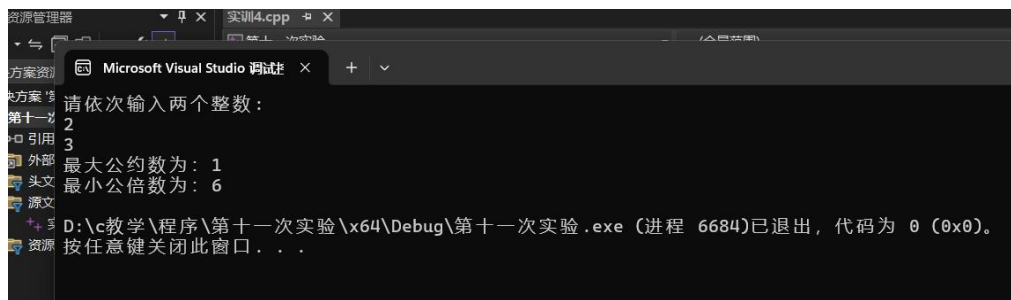
会有变化，应为去掉 `static` 后 `score`、`minScore` 和 `maxScore` 变为自动变量，在函数体内发挥作用后恢复在主程序调用函数之前的值，即 `score[0]`，不会记录得到的最大值和最小值。

实训 3



实训 4

如果两个数互为质数



否则



总结（出错信息、出错原因、修改方法、体会等。这部分需包含每个组员自己的总结体会）

出错信息、出错原因、修改方法

无

体会（张美杰）

局部变量，在函数、结构以及主函数中定义，仅限于定义它的函数或代码块{}内，函数调用结束后释放内存，不能在其他函数中访问。

全局变量，在所有函数外定义，整个程序，所有函数均可访问，生命周期贯穿整个程序。

静态变量，使用 `static` 关键字声明，作用于局部或全局，局部静态变量仅在函数内；全局静态变量仅限文件内，值在多次调用间保留，生命周期贯穿程序。

选择合适的变量类型有助于优化程序性能和可读性。局部变量占用内存小，避免全局变量干扰，适合短期使用的数据；全局变量方便多函数共享，但易增加耦合；静态变量保持值的持久性，适合记录状态。