桂林电子科技大学2019-2020学年第2学期

**程序设计与问题求解实验报告**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 实验名称 | 实验三 有序数据程序设计 | | | | | | | |  | 课号：  辅导教师意见：  成绩教师签名： |
| 院系 | **计算机与信息安全学院** | | | 专业 | | **计算机大类** | | |
| 学号 | **1900301517** | | | 姓名 | | **陆洪业** | | |
| 实验日期 | **2020** | 年 | **04** | | 月 | | **10** | 日 |
|  |  | | | | | | | |

### 一．实验目的：

1. 掌握循环程序设计思路；
2. 能熟练使用3种循环结构对问题进行求解；
3. 掌握穷举法、递推法等在工程问题中的应用。

|  |
| --- |
| 二、实验内容和步骤: |

1. 改正下列程序错误，使之完成规定的功能：

错误代码：该程序统计数组元素之和并输出。

#include <stdio.h>

int main(void)

{

int a[5]={5,4,3,2,1};

int i;

for (i=1;i<=5;i++)

{

sum=sum+a[i];

}

printf("sum=%f\n",Sum);

return 0;

}



错误原因:1.C程序中使用变量之前得先声明, 并且使用变量的值之前得先初始化,否则会出现不可预料的结果,所以题中需要对sum进行声明和初始化.

2.C语言数组引索是从0开始的,所以累加的并不是题目中的a[1]到a[5]而应该是a[0]到a[4].

3.题目错认为变量不区分大小写,C语言变量是区分大小写的,所以printf里面的Sum应改为sum.

4.printf中,引号里面的格式字符需要和变量有一对应, 数据类型也需要一一对应,而sum是整数型,应该将%f改为%d.

2. 编写程序实现以下功能

(1) 输入N 个不大于100的正整数（Ν ≤10 ）存入数组 a 中，输出 a 中的最大值，最小值及求出其平均值（保留两位小数），其中N的值由用户输入。



(2) 餐饮服务质量调查打分

某公司的主管需要了解一年来公司的营业状况，比较一下各月份的销售收入状况。如果仅给出一大堆数据，这显然太不直观了，如果能将这些数据以条形图（直方图）的形式表示，将会大大增加这些数据的直观性，也便于数据的分析与对比。下面以顾客对餐饮服务打分为例，练习这方面的程序编写方法。假设有40个学生被邀请来给自助餐厅的食品和服务质量打分，分数划分为1～10这10个等级（1表示最低分，10表示最高分），试统计调查结果，并用\*打印出如下形式的统计结果直方图。

Grade Count Histogram

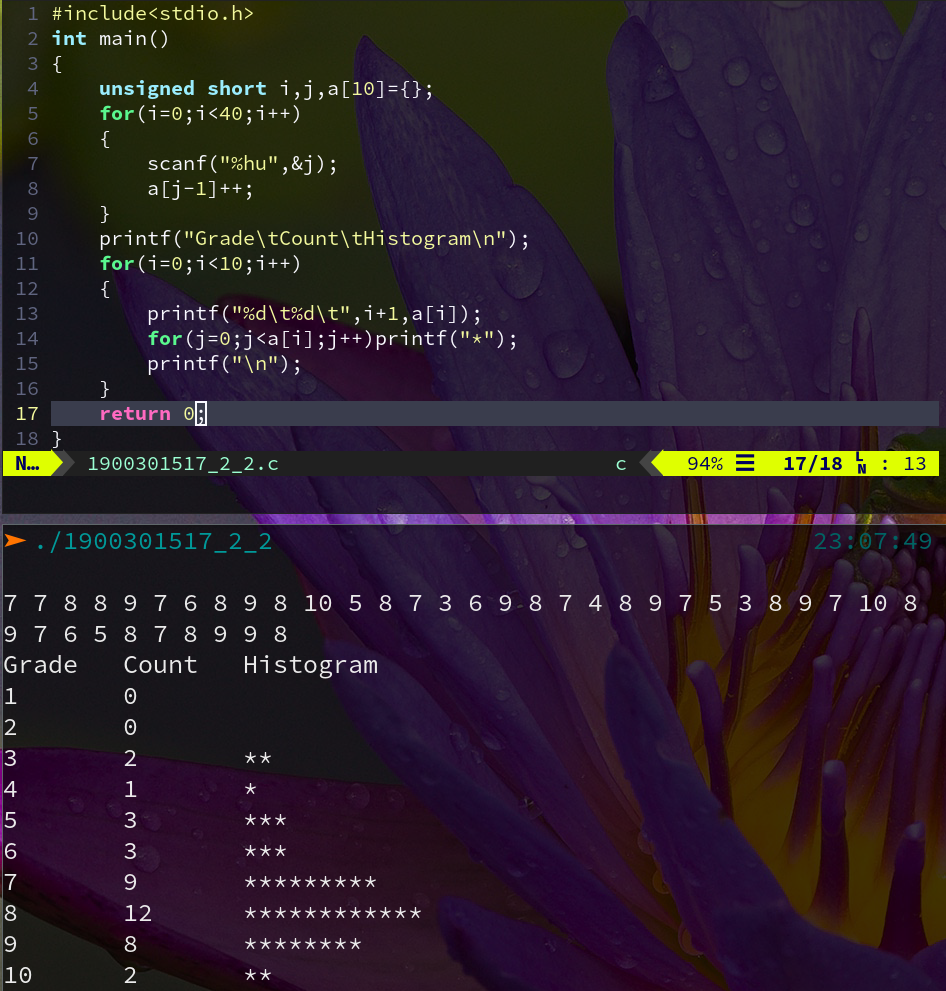
1 5 \*\*\*\*\*

2 10 \*\*\*\*\*\*\*\*\*\*

3 7 \*\*\*\*\*\*\*

...

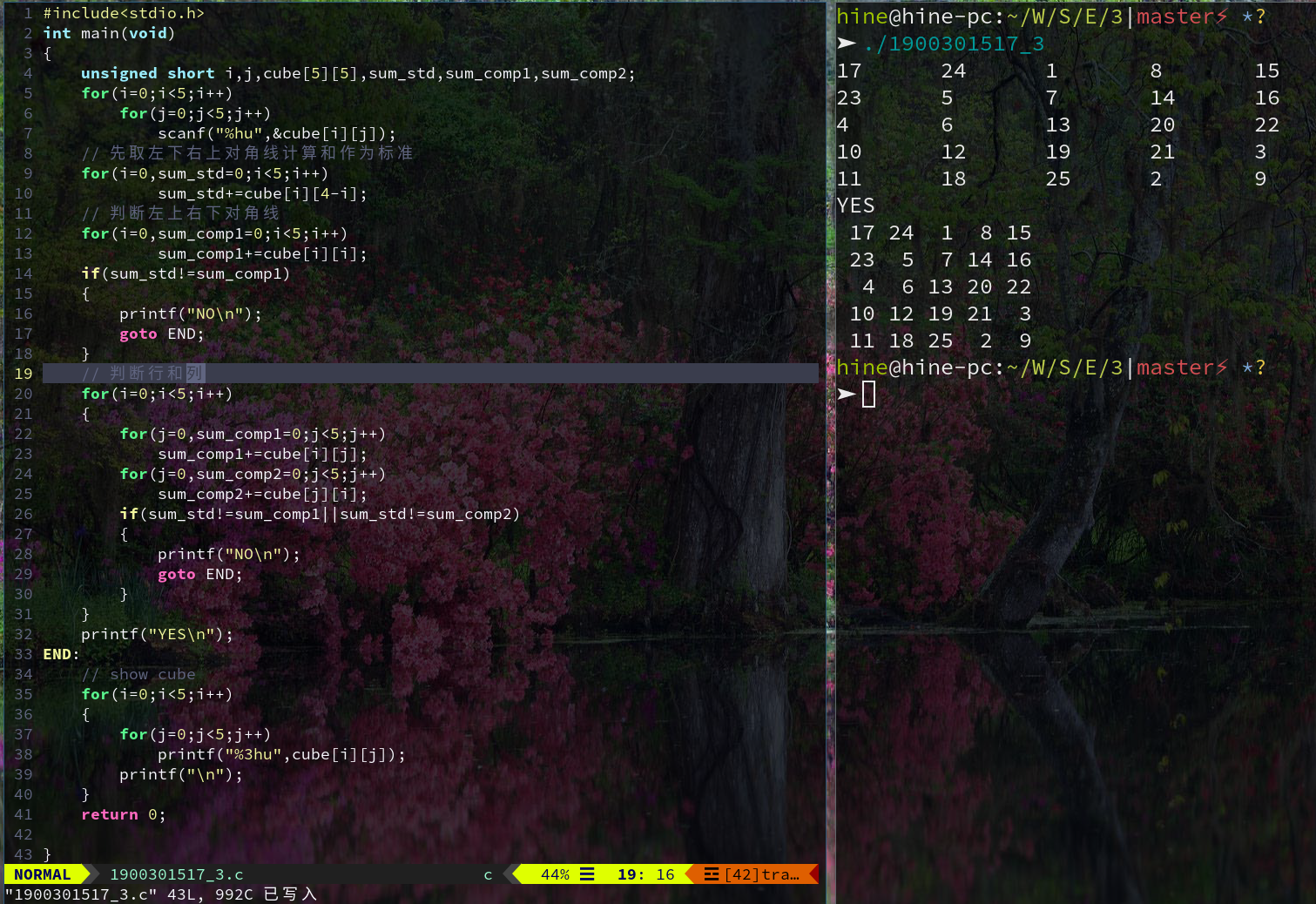
10 3 \*\*\*



3．检验并打印魔方矩阵

在下面的5×5阶魔方矩阵中，每一行、每一列、每一对角线上的元素之和都是相等的，试编写程序将这些魔方矩阵中的元素读到一个二维整型数组中，然后检验其是否为魔方矩阵，并将其按如下格式显示到屏幕上。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 17 | 24 | 1 | 8 | 15 |
| 23 | 5 | 7 | 14 | 16 |
| 4 | 6 | 13 | 20 | 22 |
| 10 | 12 | 19 | 21 | 3 |
| 11 | 18 | 25 | 2 | 9 |



### 三、实验小结

用函数之间二维数组传参问题,二维数组传参形式参数不能是(以int为例)int[][]或者int\*\*,因为二维数组实际储存还是线性的,如果不知道一列有多少个元素它就不知道该如何换行,也就无法确定位置了.

解决方法:1.固定每行元素,让它知道怎么定位,如function(int array[][10]),如果全部元素个数都确定,那么可以直接写上数组行列长度function(int array[5][5])

1. 也可以传参时,将数组行列数也传到函数里面去,这是需要使用指针自己定位:

函数定义:function(int \*\*p,m,n){...}在...里面使用数组,可以使用(\*(\*p+i)+j)或者(\*\*(p+i\*m+j))调用数组