========= **实验三：数组的定义与使用** =========

实验目的：

1、掌握一维数组及二维数组的声明、初始化和遍历的方法；

2、掌握冒泡排序、选择排序、折半查找的方法；

3、掌握字符数组的常见用法；

4、掌握常见的字符串处理函数；

5、掌握getline及gets函数的用法。

6、掌握调试的基本方法。

实验要求：

1. 实验报告及cpp文件命名、打包要求，与前期实验相同；
2. **上次布置的课后作业11和13题，将cpp与实验内容一同打包上交；**
3. **材料提交截止时间为11月29日晚10点，过期不收；**
4. **带“C++调试方法和技巧(VC6) ”PPT，方便上机时查看。**

实验题目：

1. 分析以下代码，并解释str1的最终输出结果。

注意：答案应包含**内存状态截图或内存占用效果图（画清楚就可以）**，**并辅以文字说明**。

#include <iostream>

using namespace std;

int main()

{ char str1[]={"I really love you!"};

char str2[10]={"Me too?"};

strcpy(str2,str1);

cout<<str2<<endl;

cout<<str1<<endl;

return 0;

}

1. 有一部小说，一共有3行文字，每行最多有80个字符。要求分别统计出其中英文字母、数字、空格以及其他字符的个数。

要求：无。

1. 输入n个字符串，将它们按字母由小到大的顺序排列并输出。

提示：例如string应排在student之前。不要想太多，书上就有很容易的对比方法。

1. 找出一个二维数组中的“鞍点”。即该位置上的元素在该行上最大，在该列上最小。

提示：一个二维数组中如果存在鞍点，则鞍点唯一。

要求：请思考“鞍点唯一”的原因，并简要说明。

1. 有n个数**按序**存放在一个一维数组中。现输入一个数x，用“折半查找”法找出该数位于数组的哪个位置（输出“x是数组中的第i个数”）。若该数不在数组中，则输出“查无此数”。

提示：将需要查找的x，与数组中间值对比，根据比较结果，决定从数组的哪一半中继续查找x，而另外一半数据就可以舍弃了。

要求：数组大小n通过常变量或字符常量定义，数组中数据从终端读入。

1. （提高）分析以下代码（打印杨辉三角），为何存在数组越界，却能输出正确的结果。

#include <iostream>

using namespace std;

int main()

{ static int a[10][10];

int i, j;

a[0][0]=1;

for(i=1;i<10;++i)

for(j=0;j<=i;++j)

a[i][j]=a[i-1][j-1]+a[i-1][j]; //越界

for (i=0;i<10;++i)

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

cout<<a[i][j]<<" ";

cout<<endl; }

return 0;

}

说明：本题为提高题（建议大家都尝试做一下，不用全部都做完，能做到什么程度就做到什么程度）。

要求：

1. 解释为何静态数组越界仍能产生正确结果？越界后结果正确是偶然还是必然？
2. 若a[10][10]定义为普通数组，在发生越界时，输出的所谓“乱码”是什么？乱码是完全没有逻辑随意输出的吗？
3. 写出测试用例，并辅以文字+运行结果截图或debug截图，说明以上问题。

提示：全局数据区(静态区)：全局变量和静态变量的存储是放在一块的，初始化的全局变量和静态变量在一块区域，未初始化的全局变量和未初始化的静态变量在相邻的另一块区域。测试时应注意数据初始化问题。