

	七、编程题																																			
1	<p>输入正整数 m 和方阵的阶次 n，且 $m \leq n \times n$，参照输出样例的格式，按照蛇形方式在方阵中填充从 0 到 m- 1 的数字，数字用完后用“*”代替，每行数字或者'*'之间用 Tab 键分割，行末无 Tab 键。</p> <p>样例</p> <p>输入 1:</p> <p>5</p> <p>3</p> <p>输出 1:</p> <table><tr><td>2</td><td>1</td><td>0</td></tr><tr><td>3</td><td>4</td><td>*</td></tr><tr><td>*</td><td>*</td><td>*</td></tr></table> <p>输入 2:</p> <p>9</p> <p>3</p> <p>输出 2:</p> <table><tr><td>2</td><td>1</td><td>0</td></tr><tr><td>3</td><td>4</td><td>5</td></tr><tr><td>8</td><td>7</td><td>6</td></tr></table>	2	1	0	3	4	*	*	*	*	2	1	0	3	4	5	8	7	6																	
2	1	0																																		
3	4	*																																		
*	*	*																																		
2	1	0																																		
3	4	5																																		
8	7	6																																		
2	<p>编写程序，定义一个 5×5 的二维数组（数组元素为整数），按行读入数组元素，计算数组最外层元素的平均数与次外层元素的平均数的差值（结果保留小数点后三位）。</p> <p>例如：读入 a 数组中的值为：</p> <table><tr><td>0</td><td>1</td><td>2</td><td>7</td><td>2</td></tr><tr><td>1</td><td>9</td><td>7</td><td>4</td><td>8</td></tr><tr><td>2</td><td>3</td><td>8</td><td>3</td><td>9</td></tr><tr><td>4</td><td>5</td><td>6</td><td>8</td><td>7</td></tr><tr><td>7</td><td>8</td><td>5</td><td>5</td><td>6</td></tr></table> <p>其最外层元素为：</p> <table><tr><td>0</td><td>1</td><td>2</td><td>7</td><td>2</td></tr><tr><td>1</td><td></td><td></td><td></td><td>8</td></tr></table>	0	1	2	7	2	1	9	7	4	8	2	3	8	3	9	4	5	6	8	7	7	8	5	5	6	0	1	2	7	2	1				8
0	1	2	7	2																																
1	9	7	4	8																																
2	3	8	3	9																																
4	5	6	8	7																																
7	8	5	5	6																																
0	1	2	7	2																																
1				8																																

	<div>2 9</div> <div>4 7</div> <div>7 8 5 5 6</div> <div>其次外层元素为：</div> <div>9 7 4</div> <div>3 3</div> <div>5 6 8</div> <div>则：a 数组最外层元素的平均数与次外层元素的平均数的差值：- 1.000</div> <div>输入输出样例：</div> <div>输入：</div> <div>0 1 2 7 2</div> <div>1 9 7 4 8</div> <div>2 3 8 3 9</div> <div>4 5 6 8 7</div> <div>7 8 5 5 6</div> <div>输出：</div> <div>- 1.000</div>
3	<div>写一个函数，其功能是对一个字符串内的不重复字符按降序排序，并返回结果。在主函数中输入一个不带空格的、长度小于 50 个字符的字符串，调用上述函数排序，并输出结果。</div> <div>样例</div> <div>输入：tianjin</div> <div>输出：tnjia</div> <div>输入：kensington</div> <div>输出：tsonkige</div>
4	<div>(在子函数中要使用指针，完全不用指针的扣分 20%)</div> <div>在主函数中，输入整数 n(<20)，再输入 n 个点的坐标 (float 型)。再利用下面函数计算这些点中任意两点间最大、最小距离，通过 max, min 返回主函数。并在主函数输出这两个值。</div>

	<pre>// x - - 横坐标, y - - 纵坐标, n - - 点数 void dist2(float *x, float *y, int n, float *max, float *min); 样例输入: 3 0 0 0 3 3 4 样例输出: 5.000000 3.000000</pre>
5	<p>某单位组织年底考核,对 5 位员工的 5 个方面进行打分,人事部门从键盘依次输入 5 位考核人员的信息 (编号、姓名和年龄) 和各方面得分,编写程序按总分从高到低原则对所有员工进行排序,并屏幕输出名次、员工的编号和总分。</p> <p>(1) 定义考核人员信息的结构体 STAFF, 包括编号 (整数)、姓名 (由不含空格的 26 个英文字母的大小写构成, 不超过 10 个字符)、年龄、5 方面得分和总分, 其中得分和总分均为浮点数;</p> <p>(2) 定义 void input(struct STAFF *pt)函数, 用来输入一位员工的信息 (编号、姓名和年龄) 和各项得分, 计算总分并保存;</p> <p>(3) 定义 void output(struct STAFF *pt)函数, 用来输出一位员工的编号、各项得分和总分 (其中浮点数保留小数点后 2 位);</p> <p>(4) 定义 void sort(struct STAFF *pt)函数, 对所有员工按总分由高到低进行排序 (得分相同保持原顺序);</p> <p>(5) 在主函数中调用 input、sort 和 output 函数。</p> <p>示例输入 (分隔符为空格)</p> <pre>101 Louise 28 83 73 86 75 87 102 Tom 26 85 82 93 86 90 103 Cyan 25 73 81 68 86 76 104 Indira 25 86 82 84 96 93 105 Muriel 27 75 76 83 73 86</pre> <p>示例输出 (各数据项用英文空格隔开, 英文冒号后也有一个空格, 数值数据保留两位小数, 每行最后一个数据项没有空格)</p> <pre>Number1: 104 86.00 82.00 84.00 96.00 93.00 441.00 Number2: 102 85.00 82.00 93.00 86.00 90.00 436.00</pre>

	Number3: 101 83.00 73.00 86.00 75.00 87.00 404.00 Number4: 105 75.00 76.00 83.00 73.00 86.00 393.00 Number5: 103 73.00 81.00 68.00 86.00 76.00 384.00