## 《程序设计语言实习》指导书

### 注意事项：

1. 修改《程序设计语言实习》模板封面上的“学号”、“姓名”和“指导老师”。
2. 修改正文区域的“页眉”部分，修改“学号”和“姓名”；中文为宋体，英文为Times New Roman，字号为五号；单倍行距，“段前”、“段后”设为0；“运行结果”可以文字输入，也可以截图；**注意排版，图片不要过大**，保证清晰度即可。
3. 根据不同模块要求任意选择题目。
4. 代码中注释率不得低于30%。
5. 根据题目要求画流程图（需要画流程图的题目已标注）。流程图可以用Visio，Axure等工具绘画，或者使用PPT的画图工具，不推荐直接使用Word画图。
6. 评分标准：

1）平时分占30%，根据平时上机考勤。注重平时上机，老师会不定期检查学生进度，学生不得以自己有私人电脑为借口而不来上机。

2）程序能按照题目要求正确编写，编译通过并正确运行（若以各种理由降低要求，将不予给予分数）。

3）面试得分占70%。面试得分为题目分值\*每题得分（满分10分），总分超过100按100计算。

3）学生能对自己的程序面对教师提问并能熟练地解释清楚。

4）提交电子稿，以“学号姓名”命名文件夹，包括《程序设计语言实习》电子稿（.doc）和题目的源文件（.c，学号+题目编号）（如：130801105.doc，13080110507.c，13080110524.c）。《程序设计语言实习》打印稿。学委统一收齐交指导教师。

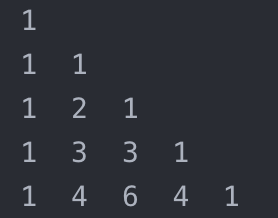
以上几项缺一不可，否则不能到得相应学分。

**模块一：选做2个题目：**

1. **统计各年龄段人数**。N个年龄通过调用随机函数获取，编写函数把0~9岁年龄段的人数放入数组下标[0]中，把10~19岁年龄段的人数放入数组下标[1]中，以此类推，把100~120岁年龄段的人数放入数组下标[10]中。
2. **求素数**。编写一个函数，将大于整数m且紧靠m的k个素数存入数组中，在命令行输入m和k。如输入17和5，输出19，23，29，31，37。（画流程图）
3. **求最大公约数和最小公倍数。**输入两个正整数m和n，求其最大公约数和最小公倍数。（画流程图）
4. **四则运算。**帮助某小学数学老师出试卷。随机生成2个800以内的正整数或零，随机计算+、-、\*或/，答案也必须是800以内的**正整数**或零。共出50道题目，不能出现重复题目（2+3和3+2算不同的题目），并将题目写入文件。（画流程图）
5. **算术运算。**输入一个算术表达式字符串，得到最后结果。算术运算符包括+、-、\*、/，注意小括号可改变优先级。例如输入“(2+3)\*4”得到20。（画流程图）
6. **两个一元多项式相加。**输入2个多项式，**例如**A17(x)=7+3x+9x8+5x17和B8(x)=8x+22x7-9x8，得到C17(x)=7+11x+22x7+5x17。
7. **玫瑰花数。**如果一个四位数等于它的每一位数的4次方之和，则称为玫瑰花数，比如：1634=1^4+6^4+3^4+4^4，编程输出所有的玫瑰花数。（画流程图）
8. **借书组合问题。**张三有五本新书，要分别借给A、B和C三位同学，如果每人每次只能借一本，则可以有多少种不同的借法，并输出所有借法。
9. **完数问题。**一个数如果恰好等于它的因子之和，这个数就称为“完数”。例如6=1＋2＋3。编程找出1000以内的所有完数。

**模块二：选做2个题目：**

1. **编写合并整数数组的函数。**数组a[M]中有m个元素（m<M），数组b[N]中有n个元素（n<N），且m+n≤M，将a、b两个数组按存放的整数升序排序并合并放入数组a中。
2. **杨辉三角问题。**输入行数n，输出对应行数的杨辉三角。

杨辉三角格式如下： 

规则：1.每一行元素数等于行数；2.每一个元素值等于前一行同列元素与前一行前一列元素之和。

1. **数组插入问题。**生成一个10元素构成的一维数组，数组元素由用户随机输入。要求：先按照升序排列并输出。再输入一个数，按照升序的规律将其插入并输出。
2. **转方阵。**对一个方阵转置，就是把原来的行号变列号，原来的列号变行号。例如，图5的方阵转置后变为图6,；但如果是对该方阵顺时针旋转（不是转置），却是如图7。请编写一个函数，实现一个方阵顺时针旋转。注：方阵是一个N\*N的矩阵。

图5 图6 图7

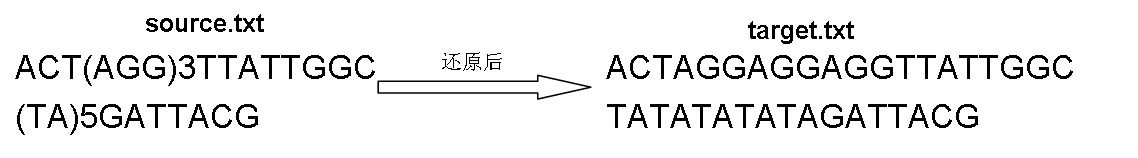
1. **链表操作。**建立单向int链表，连续输入5个结点创建链表，并实现在原链表中插入数字、删除数字、查找数字的功能。
2. **链表合并。**编写创建链表函数，链表中的结点包括学号（int）和成绩（int）。创建a、b两个链表后，再将这两个链表合并，按学号升序排序（设学号都不相同）。

**模块三：选做2个题目：**

1. **统计频率。**读入一个文件，文件中包含字母，数字，空格，标点符号等。请统计文件中的字母，数字，空格和其他符号的数目，在屏幕上显示。
2. **字符串次数问题。**编写函数统计给定字符串在文中出现的次数，如文件中内容为asd asasdfg asd as zx67 asd mklo，给定字符串为as，则函数返回6。文件名和给定字符串由命令行输入得到。
3. **统计字母个数。**读取文件中的字符串，统计从“a”到“z”26个字母各自出现的次数，并将结果放入数组中。如文件中有字符串abcdefgabcdeabc，输出33322110000000000000000000。
4. **统计单词。**要求用**命令行**读取一个英文文章（比如：文件newspap.txt），文章内有数字、空格、标点符号等。请统计出该文章中单词的个数，并按字典顺序排序存入文件新文件中。注意单词只有全部由字母组成才被计数。
5. **单词复数。**某单词本（文件）中有若干行单词单数形式，请在该单词后面给出单词复数形式，即单词本中每一行是“单数 复数”形式。注：如果单词以s、x、z、ch、或sh结尾，单词加es；如果单词以y结尾，并且y前面是一个辅音，将y改为ies；对其他单词，后面加s。
6. **猜首都。**先自己书写一个answer.txt文件，在其中存放10个国家及其首都。从该文件中随机读取出一个国家，让玩家回答出该国家的首都是什么。如回答正确则给予鼓励（如“Good Job！”），并提示是否继续。如连续3次都回答错误，则公布正确答案，并提示是否继续。回答完5道题目，给予鼓励并退出游戏。玩家输入首都时的单词不考虑大小写、空格的问题（如答案是“Beijing”，玩家输入“Bei Jing”算正确答案）。
7. **删除字符。**从源文件中读取字符，编写函数将指定字符从文件中删除，结果写入目标文件中，文件中的字母不区分大小写。源文件文件名、目标文件文件名、指定字符均从命令行得到。如源文件中内容为source.txt，指定字符为n，目标文件中内容为output.txt。注意需要用**命令行**输入参数。
8. **字符串压缩。**利用字符重复出现的次数，编写一种方法，实现基本的字符串压缩功能。比如，字符串aabcccccaaa会变为a2b1c5a3。若“压缩”后的字符串没有变短，则返回原先的字符串。可以假设字符串中只包含大小写英文字母（a至z）。

**模块四：选做1个题目：**

1. **哥德巴赫猜想验证。**验证1000以内的正偶数都能够分解为两个素数之和。（画流程图）
2. **分数比较问题。**比较两个分数的大小。（画流程图）
3. **年月日搜索问题。**中国俗话说“三天打渔两天晒网”。张三从2000年1月1号起开始“三天打渔两天晒网”，请问张三在以后的某一天是“打渔”还是“晒网”。
4. **DNA序列还原。**某生物学文件内容为多条长度不等DNA序列，为书写方便其中有些片段用小括号括起并跟一个int数字，表示该片段需要重复多次。如“(AGG)3”表示“AGG”片段需要重复3次，请按要求将源文件还原后写入新的文件中。注意：括号内的DNA片段长度不定长，重复次数可能是个多位数。



**模块五：选做1个题目：**

1. **饭卡管理程序。**
2. 建立饭卡信息：添加若干人的饭卡号、姓名、金额，要求饭卡号是唯一的；
3. 买饭：要求用户输入饭卡号、饭费，系统自动从该人的饭卡中减去饭钱，并分别显示买饭前后的金额，如果原来饭卡中的余额不足5元，则不能买饭，显示“余额不足，请充值”；
4. 充值：输入饭卡号、充值金额，充值完成后显示充值前后的金额。
5. **歌曲排行榜管理系统。**
6. 每首歌曲包括编号、歌曲名称、歌手姓名、点播次数（如001，告白气球，周杰伦， 123）；
7. 要求提供输入歌曲信息、删除歌曲信息、修改歌曲信息。（注删除、修改可按照歌曲名称查找）。
8. 程序要求有菜单，根据需求选择功能。
9. **学生成绩管理系统。**
   1. 输入n个学生的学号，姓名，高等数学、大学物理、英语和C语言四门课程的成绩。
   2. 计算每门课平均分数并输出；
   3. 按照某类课程的成绩高低排序；
   4. 找出每门课最高分的同学学号和姓名（如果最高分有多名同学，都输出）。

**代码格式：**

如下所示，其中“**[代码]**”要有适当的注释，注释量不低于代码量的30%，可以用中文。“**[运行结果]**”中要能够体现测试结果，边界值、特殊值的结果等。“**[小结]**”可以写遇到的困难，解决的方案，收获和体会等。

**[题目]**N个人的成绩存放在score数组中，编写函数，用于返回低于平均分的人数，并将低于平均分的分数存放在below数组中。

**[代码]**

#include <stdio.h>

#define N 9

/\* 找出低于平均分的人数，并返回 \*/

int fun(int score[ ], int m, int below[])

{

int i, k=0, aver=0; /\* aver保存总分和平均值 \*/

while(i<m) /\* 循环遍历所有分数 \*/

{

aver+=\*(score+i); /\* 累加求总分 \*/

i++;

}

aver/=m; /\* 求平均分 \*/

i=0; /\* 再次遍历数组 \*/

while(i<m)

{

if(\*(score+i)<aver) /\* 寻找低于平均分的分数 \*/

{

below[k++]=\*(score+i); /\* 保存到数组below中 \*/

}

i++;

}

return k; /\* 返回低于平均分的分数个数 \*/

}

int main(void)

{

int i, n, score[N], below[N];

printf("input students' score:\n"); /\* 输入学生分数 \*/

for(i=0; i<=N-1; i++)

{

scanf("%d", &score[i]);

}

n=fun(score, N, below); /\* 调用fun函数，求得低于平均分的人数 \*/

printf("\nBelow the average score are:\n");

for(i=0; i<=n-1; i++) /\* 输出低于平均分的人 \*/

{

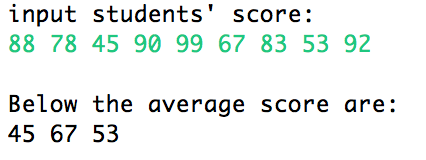
printf("%d ", below[i]);

}

return 0;

}

**[运行结果]**

****

**[小结]**此题涉及到了数组、循环语句、查找算法、函数返回等内容。必须注意平均分aver的赋值情况。如不能在一开始赋值为0，则结果将不正确。我一开始遇到了XXXX问题，通过XXXX方法解决了。