

# 目录

任务：向 C 语言问好 .....	5
任务：法老的金字塔 .....	6
任务：两棵松树 .....	7
任务：打印菜单 .....	8
任务：打印矩阵 .....	9
任务：打印机器人 .....	10
任务：字符转换 .....	11
任务：求圆柱体面积与体积 .....	12
任务：求解一元一次方程 .....	13
任务：四则运算 .....	14
任务：播放音乐 .....	15
任务：等额本息房贷计算器 .....	16
任务：歌手成绩 .....	17
任务：整数“循环”左移 .....	18
任务：数字探秘 .....	19
任务：字符加密 .....	20
任务：数值排序 .....	21
任务：阶乘计算器 .....	22
任务：水仙花数 .....	23
任务：最大公约数 .....	24
任务：最小公倍数 .....	25

任务：小球落地.....	26
任务：丘比特之箭.....	27
任务：班级成绩.....	28
任务：计算求值.....	29
任务：坐标判断.....	30
任务：九九乘法表.....	31
任务：分数四则运算.....	32
任务：打印宝石.....	33
任务：猜拳游戏.....	34
任务：加减法自测.....	35
任务：数组逆置.....	36
任务：查找最值.....	37
任务：数值交换.....	38
任务：数字插入.....	39
任务：数组循环左移.....	40
任务：数组合并.....	41
任务：数组排序.....	42
任务：矩阵乘法.....	43
任务：字符串连接.....	44
任务：数据加密.....	45
任务：绘制自定义迷宫地图.....	46
任务：短信分割.....	47

任务：密码表加密 .....	48
任务：手机号验证 .....	49
任务：进制转换 .....	50
任务：字符串长度 .....	51
任务：乱飞的机器人 .....	52
任务：寻找亚军 .....	54
任务：数组排序 .....	55
任务：凯撒密码 .....	56
任务：放鞭炮 .....	58
任务：死亡游戏 .....	59
任务：年龄计算 .....	61
任务：字符串转换为整数 .....	62
任务：最大公约数（递归） .....	63
任务：两点之间距离 .....	64
任务：点到直线距离 .....	65
任务：点是否在圆内 .....	66
任务：点是否在矩形内 .....	67
任务：两圆是否相交 .....	68
任务：学生排序 .....	69
任务：学生查找 .....	70
任务：链表信息打印 .....	71
任务：链表合并 .....	72



**任务：向 C 语言问好****问题描述：**

在屏幕上打印 "Nice to meet you, C! I am Zhangfei."，将"Zhangfei"换成自己的名字。

**程序输入：**

无。

**程序输出：**

输出带有自己姓名的向 C 语言问好语句。

**输入示例：**

无。

**输出示例：**

Nice to meet you, C! I am Guanyu.

## 任务：法老的金字塔

### 问题描述：

屏幕上输出金字塔图案。

### 程序输入：

无。

### 程序输出：

字符组成的金字塔图案。

### 输入示例：

无。

### 输出示例：

```
  *
 ***
*****
*****
```

**任务：两棵松树**

**问题描述：**

编程在屏幕上打印两棵松树图案。

**程序输入：**

无。

**程序输出：**

字符组成的两颗松树图案。

**输入示例：**

无。

**输出示例：**

```
      *          *
    ***        ***
  *****    *****
  | |        | |
```

**任务：打印菜单**

**问题描述：**

打印功能菜单。

**程序输入：**

无。

**程序输出：**

打印输出示例中由字符组成的菜单。

**输入示例：**

无。

**输出示例：**

```
-----  
[1] 加法          [2] 减法  
[3] 乘法          [4] 除法  
[5] 平方          [6] 开方  
[7] 绝对值        [0] 退出  
-----
```

请输入您的选择 ( 0---7 ):



**任务：打印矩阵****问题描述：**

打印一个 4 阶矩阵。

**程序输入：**

无。

**程序输出：**

在命令行界面打印一个 4 阶矩阵，矩阵中的数字要求对齐。

**输入示例：**

无。

**输出示例：**

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

## 任务：打印机器人

### 问题描述：

在屏幕上打印一个机器人。

**程序输入：**

无。

**程序输出：**

在命令行界面打印一个由字符组成的机器人图案。。

**输入示例：**

无。

**输出示例：**

[illegible]

## **任务：字符转换**

### **问题描述：**

编写程序获取用户输入的三个小写字母的 ASCII 码以及其对应的大写字母。

### **程序输入：**

用户在命令行界面输入三个小写字母，输入时字母之间使用空格隔开。

### **程序输出：**

按照顺序输出用户输入的三个小写字母的 ASCII 码及其对应的大写字母。

### **输入示例：**

a b c

### **输出示例：**

97 98 99

A B C

## **任务：求圆柱体面积与体积**

### **问题描述：**

编写程序求给定圆柱体的表面积与体积。

### **程序输入：**

用户在命令行界面依次输入底圆半径和圆柱高度,每次输入前都会有输入提示信息提示当前输入的是什么,半径与高度均为浮点数类型。

输入示例中"圆柱体底圆半径为："、"圆柱体高度为："均为程序提示信息,其余部分为用户输入。

### **程序输出：**

根据用户输入的底圆半径和圆柱高度输出对应圆柱体表面积与体积,面积与体积均为浮点数类型,小数点后面要求保留两位有效数字。 $\pi$ 的取值使用 3.14。

### **输入示例：**

圆柱体底圆半径为：1.5

圆柱体高度为：3

### **输出示例：**

圆柱体表面积为：42.39

圆柱体体积为：21.20

## 任务：求解一元一次方程

### 问题描述：

编写程序求解方程  $ax+b=0$ ， $a$  和  $b$  为用户输入。

### 程序输入：

用户输入  $ab$ ，输入时  $ab$  之间使用空格隔开， $ab$  的值均为正整数，并且在输入时  $a$  不出现为 0 的情况。输入时要有输入提示。

输入示例中的"请输入一次方程的系数  $a$  和  $b$ ："为程序提示信息，其余部分为用户输入。

### 程序输出：

首先根据输入的  $ab$  输出方程的完整形式，然后再输出方程的解。

### 输入示例：

请输入一次方程的系数  $a$  和  $b$ ：2 6

### 输出示例：

一次方程  $2x+6=0$  的根是： $x = -3$

## 任务：四则运算

### 问题描述：

求 2 个整数的和、差、积、商、平均值。

### 程序输入：

用户输入两个数，两个数均为非 0 整数，输入时两个数之间使用空格隔开。

### 程序输出：

程序依次输出所输入两个数的和、差、积、商、平均值，其中商与平均值为浮点数类型，小数点后面保留两位有效数字。

### 输入示例：

93 88

### 输出示例：

两个数的和、差、积、商为：181、5、8184、1.06

两个数的平均值为：90.50

### 解题提示：

- 1、C 语言里加、减、乘、除运算符为：+、-、\*、/
- 2、和、差、积、商、平均值需要定义变量保存。

## 任务：播放音乐

### 问题描述：

C 程序不只是一个控制台窗口里做些简单的计算、输出，你可以用它来播放声音，但这需要调用 windows.h 里的 PlaySound 函数来播放 wav 格式音频文件。自己写一个程序播放一段自己的 wav 音频文件（不要用示例音频）。

### 程序输入：

无。

### 程序输出：

无。

### 输入示例：

无。

### 输出示例：

无。

### 解题提示：

无。

### 解题提示：

1. PlaySound 是 windows 的系统函数，请查看 MSDN 或百度百科：  
<http://baike.baidu.com/view/1014003.htm>
2. 需要添加头文件：windows.h 和 mmsystem.h，链接的时候用到 WINMM.LIB 库，所以在主函数前加一句：#pragma comment(lib, "WINMM.LIB")
3. 你的 wav 文件需要和源文件放到一个目录，注意这个函数只能播放 wav 文件。
4. 可以这样调用 PlaySound 函数：  
PlaySound(TEXT("DOOR1.wav"), NULL, SND\_FILENAME | SND\_ASYNC);  
函数有三个参数，第一个参数为文件名，但文件名不是 C 语言的字符串，所以可以这样写：TEXT("DOOR1.wav")，第二参数写 0 就可以了，第三个参数一般用 SND\_FILENAME | SND\_ASYNC 就可以了

## 任务：等额本息房贷计算器

### 问题描述：

拥有一套属于自己的房子是每个年轻人的梦想，在购房时一般需要贷款，商业贷款有两种还款方式：等额本息和等额本金。等额本息还款法，在偿还初期利息支出最大，本金最少，以后利息支付逐步减少，本金逐步增加，但每月以相等的额度偿还金额（本+息）。它比较适合收入低且积蓄少的年轻人，因月供压力小不会降低生活质量。

公式为：

$$\text{monthPay} = \text{loan} * \text{monthRate} + \frac{\text{loan} * \text{monthRate}}{(1 + \text{monthRate})^{\text{monthes}} - 1}$$

monthPay 为月供

monthRate 为月利率，monthate = 年利率 / 12，为了方便计算，年利率固定为 5.56%

monthes 为贷款总月数，必须为 12 的倍数

请编写一个程序，根据指定的贷款金额和贷款总月数，计算总还款额和总利息。

### 程序输入：

依次输入贷款年限与贷款金额，每次输入需有提示信息。贷款年限以及贷款金额均为大于 0 的正整数。

输入示例中"请输入贷款年限："、"请输入贷款金额："为提示信息，其余为用户输入部分。

### 程序输出：

根据输入的贷款年限以及贷款金额，计算并输出还款总额、利息总额、月供，以上三个数值均为浮点数类型，结果小数点后面保留三位有效数字。

### 输入示例：

请输入贷款年限：20

请输入贷款金额：200000

### 输出示例：

还款总额为：331814.640

利息总额为：131814.640

月供为：1382.561

### 解题提示：

- 1、公式中分母中减 1，1 没在指数上
- 2、关于等额本息还款法详情见百度百科：

<http://baike.baidu.com/subview/1180521/1180521.htm>



**任务：歌手成绩****问题描述：**

在歌星大奖赛中,有 5 个评委为参赛的选手打分,分数为 1~100 分。选手最后得分为:去掉一个最高分和一个最低分后其余 3 个分数的平均值。请编写一个程序实现该过程。

**程序输入：**

依次输入五个分数,分数为 1~100 之间,可以出现小数点后面 1 位的小数,分数之间使用","(英文逗号)隔开。

**程序输出：**

根据打分规则,求出选手的最终的分并输出,输出时选手分数要求小数点后面保留 1 位有效数字。

**输入示例：**

93,92.5,88,96,94.1

**输出示例：**

93.2

**任务：整数“循环”左移****问题描述：**

将一个三位整数的十进制位向左循环移动 2 位输出。循环左移是指将移出的位移动到最低位。

**程序输入：**

输入一个三位正整数。

**程序输出：**

将输入的三位正整数按照规则进行循环左移两次并输出。

**输入示例：**

123

**输出示例：**

312

## **任务：数字探秘**

### **问题描述：**

求一个字符的二进制位中包含多少个 1。

### **程序输入：**

输入一个字符作为字符型进行读取。

### **程序输出：**

求所输入字符的二进制位中包含多少个 1。例如字符'A'的 ASCII 码为 65，其二进制为 0100 0001，所以最终输出 2。

### **输入示例：**

A

### **输出示例：**

2

## 任务：字符加密

### 问题描述：

从键盘输入 5 个字符，对其进行加密。

加密的规律是：用原来的字母后面第四个字母代替原来的字母。

例如，字母 A 后面第四个字母是 E，则用 E 代替。因此，若输入的是“China”，应译为“Glmre”。如果字符是 w，则加 4 为 a，x 对应 b，y 对应 c，z 对应 d。

### 程序输入：

连续输入五个字符。

### 程序输出：

将输入的五個字符按照规律加密后再输出。

### 输入示例：

China

### 输出示例：

加密后的字符是：Glmre

### 解题提示：

1、可以考虑用模运算，比如如果是大写字母 'W'，加 4 等于 91 已经大于 'Z' 了，可以用  $( 'W' + 4 - 'A' ) \% 26 + 'A'$  得到最终加密后字符 'A'，公式中 'W' 为要加密的字符。

2、字符区分大小写，1 中公式也需要考虑大写字母与小写字母的情况。

**任务：数值排序****问题描述：**

输入 3 个整数，然后按从大到小的排序输出。

**程序输入：**

由用户输入三个整数，每个数之间使用空格隔开。

**程序输出：**

将输入的三个数按照从大到小的顺序进行输出。

**输入示例：**

20 90 35

**输出示例：**

90 35 20

## **任务：阶乘计算器**

### **问题描述：**

计算一个数的阶乘。

### **程序输入：**

用户输入一个大于 0 的正整数，如果用户输入的整数小于等于 0 的话，给出错误提示，然后要求用户重新输入。

### **程序输出：**

输出用户输入数的阶乘。

### **输入示例：**

4

### **输出示例：**

24

**任务：水仙花数****问题描述：**

水仙花数是指一个  $n$  位数( $n \geq 3$ )，它的每个位上的数字的  $n$  次幂之和等于它本身。(例如： $1^3 + 5^3 + 3^3 = 153$ )。

列举出所有的三位数的水仙花数。

**程序输入：**

无。

**程序输出：**

输出所有的三位数的水仙花数，每个水仙花数之间使用空格隔开。

**输入示例：**

无。

**输出示例：**

153 370 371 407

## 任务：最大公约数

### 问题描述：

最大公约数指某几个整数共有因子中最大的一个。输入三个整数，求这三个整数的最大公约数是多少。

### 程序输入：

用户输入三个大于 0 的正整数，每个数之间使用空格隔开。

### 程序输出：

输出三个数的最大公约数。

### 输入示例：

4 8 12

### 输出示例：

4



## 任务：最小公倍数

### 问题描述：

最小公倍数如果有一个自然数  $a$  能被自然数  $b$  整除，则称  $a$  为  $b$  的倍数， $b$  为  $a$  的约数，对于  $n$  个整数来说，指该两数共有倍数中最小的一个。

输入三个整数，求这三个整数的最小公倍数是多少。

### 程序输入：

用户输入三个大于 0 的正整数，每个数之间使用空格隔开。

### 程序输出：

输出三个数的最小公倍数。

### 输入示例：

4 6 12

### 输出示例：

12

## 任务：小球落地

### 问题描述：

一球从 100 米高度自由落下，每次落地后反跳回原高度的一半；再落下。求第 10 次落地时，总共反弹了多长距离？每次反弹多高？

### 程序输入：

无。

### 程序输出：

输出小球落下前 10 次每一次的反弹高度，最后再输出前十次总共反弹高度。反弹高度为浮点数类型，要求小数点后面保留 6 位有效数字。

### 输入示例：

无。

### 输出示例：

第 1 次反弹的高度:50.000000  
第 2 次反弹的高度:25.000000  
第 3 次反弹的高度:12.500000  
第 4 次反弹的高度:6.250000  
第 5 次反弹的高度:3.125000  
第 6 次反弹的高度:1.562500  
第 7 次反弹的高度:0.781250  
第 8 次反弹的高度:0.390625  
第 9 次反弹的高度:0.195313  
第 10 次反弹的高度:0.097656  
小球反弹的总高度:99.902344

## 任务：丘比特之箭

### 问题描述：

编写一个程序，在控制台窗口打印出一只箭和一个箭靶，并实现箭射向箭靶的动画。（箭和箭靶的样式仅供参考！如果认为自己的图案更漂亮，我们鼓励创新！）

### 程序输入：

无。

### 程序输出：

打印出一只箭和一个箭靶，并实现箭射向箭靶的动画。

### 输入示例：

无。

### 输出示例：

```
##>----->
[
(
{
(
[
```

### 解题提示：

1、在程序中添加 `#include <windows.h>`，并在 `main` 函数前添加下面函数。通过此函数可设置命令行界面当前光标位置，例如设置光标位置为第一行第一个字符位置时可已通过 `setCursorPosition(0, 0)`；来实现。由于循环执行较快无法看到动画效果，可以在循环中添加 `Sleep(1000)`；来达到延时的效果，参数为毫秒数。

```
/*
 * setCursorPosition()
 * 说明：设置光标当前位置到指定位置
 * 参数： row      行
 *        column   列
 */
void setCursorPosition(int row, int column)
{
    HANDLE hOut;
    COORD pos = {column, row};
    hOut = GetStdHandle(STD_OUTPUT_HANDLE);
    SetConsoleCursorPosition(hOut, pos);
}
```

2、动画是由一幅一幅图片连续切换产生的效果，通过不停地打印箭与箭靶，箭的位置离箭靶越来越近就会产生箭射向箭靶的动画效果，每次打印完箭和箭靶使用 `Sleep(1000)`；来使程序暂停 1 秒钟，然后再使用提示 1 中的函数重新定位命令行中光标位置，通过打印空格将之前打印的箭和箭靶覆盖掉，使命令行界面清空，然后再重新定位光标位置来打印下一次的箭和箭靶。

## **任务：班级成绩**

### **问题描述：**

编写一程序，接收从键盘上输入的一个班的学生的数学成绩，直到输入的成绩为负数停止，然后求这个班学生数学平均成绩、最高分、最低分。

### **程序输入：**

输入一个班的成绩，没输入完一个成绩使用回车来完成输入，当所有人成绩都输入完成之后，输入一个负成绩，负成绩表示输入结束。前面有效成绩的取值范围为 0~100 之间的整数。

### **程序输出：**

根据输入的所有学生成绩输出这个班的平均成绩、最高分、最低分，平均成绩为浮点数类型，小数点后面保留两位有效数字。

### **输入示例：**

```
89
96
75
66
-1
```

### **输出示例：**

平均分为：81.50，最高分为：96，最低分为：66

### **解题提示：**

1、程序可以使用循环来完成，每次循环完成可以求到当前所有输入成绩的总分、个数、最高分、最低分，当输入负值时表示循环结束，此时再求出平均分。

**任务：计算求值**

**问题描述：**

编程计算  $1 \times 2 \times 3 + 3 \times 4 \times 5 + 5 \times 6 \times 7 + \dots + 99 \times 100 \times 101$  的值。

**程序输入：**

无。

**程序输出：**

输出  $1 \times 2 \times 3 + 3 \times 4 \times 5 + \dots + 99 \times 100 \times 101$  的值。

**输入示例：**

无。

**输出示例：**

13002450

**任务：坐标判断****问题描述：**

编写一程序，接收从键盘上输入的两个实数，判断这两个实数作为坐标所在的象限。

**程序输入：**

用户输入两个数，中间使用空格隔开，每个数都是浮点数类型，第一个数为 x 坐标，第二个数为 y 坐标。

**程序输出：**

根据输入的坐标，求对应点所在象限。

**输入示例：**

2.5 -5.6

**输出示例：**

第四象限

**解题提示：**

- 1、需要考虑坐标在原点的情况。
- 2、需要考虑坐标在坐标轴上的情况。

**任务：九九乘法表**

**问题描述：**

打印九九乘法表。

**程序输入：**

无。

**程序输出：**

打印九九乘法表，表格要求对齐。

**输入示例：**

无。

**输出示例：**

```
1*1=1  1*2=2  1*3=3  1*4=4  1*5=5  1*6=6  1*7=7  1*8=8  1*9=9
2*2=4  2*3=6  2*4=8  2*5=10 2*6=12 2*7=14 2*8=16 2*9=18
3*3=9  3*4=12 3*5=15 3*6=18 3*7=21 3*8=24 3*9=27
4*4=16 4*5=20 4*6=24 4*7=28 4*8=32 4*9=36
5*5=25 5*6=30 5*7=35 5*8=40 5*9=45
6*6=36 6*7=42 6*8=48 6*9=54
7*7=49 7*8=56 7*9=63
8*8=64 8*9=72
9*9=81
```

## 任务：分数四则运算

### 问题描述：

计算由用户输入的两个分数，求它们的加、减、乘、除结果。

### 程序输入：

用户输入两个分数，分数与分数之间使用空格隔开，每个分数分子与分母之间使用 '/' 隔开。

### 程序输出：

求两个分数的加、减、乘、除结果，最终结果要求进行化简。

### 输入示例：

4/3 9/8

### 输出示例：

和为：59/24，

差为：5/24

乘为：3/2

商为：32/27



## 任务：打印宝石

### 问题描述：

编写程序，用\*号打印出一个菱形，要求使用 for 循环或 while 循环。

### 程序输入：

无。

### 程序输出：

在命令行界面打印一个由 '\*' 组成的菱形图案。

### 输入示例：

无。

### 输出示例：

```
  *
 ***
*****
*****
*****
 ***
  *
```

## 任务：猜拳游戏

### 问题描述：

实现一个猜拳游戏，让你与电脑对决。你出的手势由你自己决定，电脑则随机出拳，最后判断胜负。要求使用循环实现。

### 程序输入：

用户通过输入数字表示自己的出招，1:石头 2:剪子 3:步 4:退出，其它值表示非法输入，要求重新输入，每次输入前要给出提示信息"请您先出(1:石头 2:剪子 3:步 4:退出)："。

### 程序输出：

当用户输入完出招之后，根据用户输入情况做不同的输出。当用户输入非法时需提示用户输入非法，要求重新输入；当用户输入正确时，再输出电脑的出招，然后再输出胜负情况。

### 输入示例：

- 例 1 请您先出(1:石头 2:剪子 3:布 4:退出)：1
- 例 2 请您先出(1:石头 2:剪子 3:布 4:退出)：4
- 例 3 请您先出(1:石头 2:剪子 3:布 4:退出)：5

### 输出示例：

- 例 1 您 出 石头  
电脑出 石头  
平局
- 例 2 再见！
- 例 3 输入错误，请重新输入！

### 解题提示：

1、生成随机数需要使用使用 rand 函数和 srand 函数，源文件中添加头文件 stdlib.h 和 time.h。

srand 函数是添加随机种子，保证程序每次运行时获取的随机值都是不一样的。

rand 函数会根据前面的随机数种子产生一个随机整数。

2、rand 和 srand 函数的用法如下：

srand(time(0)); // 讲当前时间设置为随机数种子，可以将本函数的调用放在 main 函数中，一个程序只需要设置一次随机数种子

int num = rand()%3+1; // 产生 1~3 的随机数赋值给 num，可以将本函数的调用放在循环里面，没执行一次 rand 函数都会重新生成一个随机数。

## 任务：加减法自测

### 问题描述：

编写加减法自测程序，程序自动生成十道两个 0~100 之间整数的加减法题目，然后由用户输入结果，程序判断结果是否正确并在所有题目完成之后输出最终总分。

如果输入答案正确，会显示“Right!”，否则显示“Not correct!”，不给机会重做，连续做 10 道题，10 道题做完后，按每题 10 分统计总得分，然后打印出总分。

### 程序输入：

每次答题程序需显示当前是第几题，并将需要求结果的算式打印出来，再由用户输入加减法的结果，结果为整数。

### 程序输出：

正确输出"Right!"，不正确输出"Not correct!"，所有题目做完后输出测试总分。

### 输入示例：

例 1     第 1 题 45 - 37 = 8  
例 2     第 2 题 83 - 87 = 10  
例 3     第 10 题 36 - 59 = 20

### 输出示例：

例1     Right!  
例 2     Not correct!  
例 3     Not correct!  
         您的最终成绩为 80

### 解题提示：

1、本题中运算数与运算符都可以通过随机数确定，生成随机数需要使用使用 rand 函数和 srand 函数，源文件中添加头文件 stdlib.h 和 time.h。

srand 函数是添加随机种子，保证程序每次运行时获取的随机值都是不一样的。

rand 函数会根据前面的随机数种子产生一个随机整数。

2、rand 和 srand 函数的用法如下：

srand(time(0));     // 讲当前时间设置为随机数种子，可以将本函数的调用放在 main 函数中，一个程序只需要设置一次随机数种子

int num = rand()%3+1; // 产生 1~3 的随机数赋值给 num，可以将本函数的调用放在循环里面，没执行一次 rand 函数都会重新生成一个随机数。

**任务：数组逆置****问题描述：**

将给定数组里面的元素进行逆序存放，给定数组为 `int arr[6] = {1,2,3,4,5,6};`，即要求将数组 `arr` 中存放的数据变为{6,5,4,3,2,1}。本程序需使用循环完成数组逆置操作。

**程序输入：**

无。

**程序输出：**

将逆置后的数组进行打印输出，元素之间使用空格隔开。

**输入示例：**

无。

**输出示例：**

1 2 3 4 5 6

**任务：查找最值****问题描述：**

编写程序，对于一维数组 `int arr[5]`，数组元素由用户输入，输入完数组后打印输出数组中次小（第二小）元素的下标。

**程序输入：**

依次输入数组中的五个元素，元素之间使用空格隔开，每个元素都为整数。

**程序输出：**

输出第二小的元素的下标，下标从 0 开始。

**输入示例：**

11 3 5 17 9

**输出示例：**

次最小元素的下标：2

## **任务：数值交换**

### **问题描述：**

对于数组 `int arr[10]`，由用户输入的 10 个元素，然后使数组中最大的与第一个元素交换，最小的与最后一个元素交换，并输出打印交换后的数组。

### **程序输入：**

依次输入数组中的 10 个元素，元素之间使用空格隔开，每个元素都为整数。

### **程序输出：**

按顺序输出交换后的数组所有元素。

### **输入示例：**

3 6 2 7 1 9 8 0 5 4

### **输出示例：**

9 6 2 7 1 3 8 4 5 0

## 任务：数字插入

### 问题描述：

有序数组 `int arr[7] = {1, 8, 10, 16, 23, 40}`。现输入一个数，要求按原来的顺序将它插入数组中，使插入新数后的数组依然是有序的，并将插入新数后的数组按照顺序打印输出。

### 程序输入：

由用户输入一个整数。

### 程序输出：

按顺序打印输出插入数字后的数组。

### 输入示例：

20

### 输出示例：

1 8 10 16 20 23 40

### 任务：数组循环左移

#### 问题描述：

已有数组 `int arr[6] = {1,2,3,4,5,6};`，编写一程序，根据输入的整数 `i`，将数组循环左移 `i` 位，即修改数组中元素存放的位置，然后再按照顺序进行输出。

#### 程序输入：

由用户输入一个大于 0 的整数。

#### 程序输出：

将循环左移后的数组进行顺序打印。

#### 输入示例：

2

#### 输出示例：

3 4 5 6 1 2

#### 解题提示：

- 1、注意不要数组越界。
- 2、当 `i` 的值大于数组长度时如何操作。



## 任务：数组合并

### 问题描述：

将 2 个已知有序的数组合并为一个新的有序数组。两个数组为 `int arr1[4] = {3,6,9,11};` `int arr2[5] = {1,3,8,45,89};` , 要求合并时要去除数组中的重复数据。数组合并之后按顺序进行输出。

### 程序输入：

无。

### 程序输出：

按顺序输出合并后的新数组。

### 输入示例：

无。

### 输出示例：

1 3 6 8 9 11 45 89

## 任务：数组排序

### 问题描述：

已知数组 `int arr[7];`，数组元素由用户输入，然后将数组内下标为奇数的数组元素按照从小到大的顺序排序，最后将排序后的数组进行打印输出。

### 程序输入：

有用户输入数组中的七个元素，每个元素之间使用空格隔开，每个元素均为整数。

### 程序输出：

打印输出排序后的数组。

### 输入示例：

21 42 38 94 5 31 32

### 输出示例：

21 31 38 42 5 94 32

## 任务：矩阵乘法

### 问题描述：

已知有两个矩阵 A、B，编写程序计算其乘积矩阵 C，并将矩阵 C 进行打印。

$$A = \begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix}, \quad B = \begin{vmatrix} -1 & 2 & 1 \\ 2 & 1 & 1 \\ 3 & 5 & 1 \end{vmatrix}$$

### 程序输入：

无。

### 程序输出：

输出打印乘积矩阵 C，要求矩阵 C 要按行进行打印。

### 输入示例：

无。

### 输出示例：

```
12 19 6
24 43 15
36 67 24
```

## 任务：字符串连接

### 问题描述：

已知字符串 `char str1[] = "hello"; char str2[] = " world!";`,编写程序将两个字符串连接成一个新的字符串 `str3[20] = "hello world!";`,最后再将 `str3` 进行输出,不允许使用 `strcat` 函数。

### 程序输入：

无。

### 程序输出：

输出连接后的字符串。

### 输入示例：

无。

### 输出示例：

hello world !

## **任务：数据加密**

### **问题描述：**

编写程序对数据进行加密，数据是四个 1 位正整数。加密规则如下：每个数字都加上 5，然后除以 10 的余数代替该数字，再将第一个和第四个交换，第二个和第三个交换。

由用户输入待加密数据，请输出加密后的数据。

### **程序输入：**

用户输入待加密数据，每个数字之间使用空格隔开，每个数字均为 1 位正整数。

### **程序输出：**

输出加密后的数据。

### **输入示例：**

7 9 6 8

### **输出示例：**

3 1 4 2

## 任务：绘制自定义迷宫地图

### 问题描述：

用户输入 8 行 8 列的二维数组用来表示迷宫，然后根据二维数组来绘制迷宫地图。二维数组中 0 表示空地，1 表示墙，空地在绘制时使用空格，墙在绘制时使用的字符需要用户来输入。

### 程序输入：

首先提示用户输入表示墙图案的字符，字符使用字符型进行读取，提示信息为"请输入表示墙图案的字符："，然后再输入二维数组中的数据，二维数组中数据只能是 0 或 1，输入前需给出提示信息"请输入二维数组数据："。

### 程序输出：

根据用户输入的墙图案以及二维数组的数据，绘制迷宫图案。

### 输入示例：

请输入表示墙图案的字符：\*

请输入二维数组信息：

```
11111111
10000001
10000001
11110001
10000011
10000001
10000001
10000001
11111111
```

### 输出示例：

```
*****
*      *
*      *
****  *
*     **
*      *
*      *
*****
```

## 任务：短信分割

### 问题描述：

手机可以收发的短信都以 70 字为限，当用户编写超过 70 字的短信发送时，手机系统将信息自动分割为 70 字（不满 70 也算一条）一条的短信，请模拟该功能编写程序。

### 程序输入：

用户输入要发送的短信信息，信息可以定义一个足够长的字符数组来存储，例如 `char str[100];`。

### 程序输出：

将输入的信息按要求拆分成多条进行输出。

### 输入示例：

I know you can do it, but you failed at least. Now you have worked for three years and earn much money, and buy a house.

### 输出示例：

1. I know you can do it, but you failed at least. Now you have worked for
2. three years and earn much money, and buy a house.

## 任务：密码表加密

### 问题描述：

密码表加密是一种十分常用的密码加密方法，加密的原理是根据明文表和密码表，加密形成密文，根据密文和密码表解密，读出明文。

要求设计自己的密码表，不允许彼此相同密码表，明文表与密码表可创建字符数组表示，例如 `char arr1[] = "abcdefghijklmnopqrstuvwxyz1234567890"`；`char arr2[] = "18ac4y7bxuiep23hjs5ofwv0zd19gkm6nqrt"`，即 1 是 a 的密文，8 是 b 的密文。

### 程序输入：

需要用户输入待加密明文字符串数据，字符串数据需要创建字符数组存放。

### 程序输出：

输出根据密码表加密后的数据。

### 输入示例：

abcdefghijklmnopqrstuvwxyz1234567890

### 输出示例：

18ac4y7bxuiep23hjs5ofwv0zd19gkm6nqrt



## 任务：手机号验证

### 问题描述：

目前我国开放的手机号段达到 27 种。其中，移动 16 种，联通 7 种，电信 4 种。其中：

移动：134、135、136、137、138、139、147、150、151、152、157、158、159、182、187、188

联通：130、131、132、155、156、185、186

电信：133、153、180、189

请编写一个程序根据用户输入的手机号判断这个手机号是哪个公司的。要求验证手机号码是否都位数字，如果出现非数字字符，则需重新收入；验证手机号码是否为 11 位，如果不是则重新输入；取号码前 3 位，判断该号码是否为移动、电信、联通的合法号码，不是则重新输入。

### 程序输入：

由用户输入手机号，手机号作为字符串进行读取。

### 程序输出：

根据用户输入的手机号按照要求进行判断，如果不合法则给出相应提示信息，并要求重新输入，如果合法则输出手机号的运营商。

### 输入示例：

例 1      136731373k7

例 2      1367313778

例 3      12018917812

例 4      13673137389

### 输出示例：

例 1      手机号码不能含有非数字字符，请重新输入！

例 2      手机号码长度小于 11 位，请重新输入！

例 3      没有这样的号码，请重新输入！

例 4      你用的是移动的手机。

## 任务：进制转换

### 问题描述：

编写程序将输入的十进制整数转换成二进制并输出,要求程序中编写一个函数实现对于任何一个十进制整数，打印该整数的二进制形式，函数形式参数为一个十进制整数，该函数无返回值。

### 程序输入：

由用户输入一个整数。

### 程序输出：

输出输入整数的二进制，要求每四位二进制添加一个空格。

### 输入示例：

-25

### 输出示例：

1111 1111 1111 1111 1111 1111 1110 0111

## **任务：字符串长度**

### **问题描述：**

编写程序，输入一个字符串，输出该字符串的长度。要求程序中编写一个函数计算该字符串的长度，函数头为 `int myStrlen(const char * str);`，形参为要求长度的字符串，返回值为字符串长度，不允许使用标准库中的 `strlen` 函数。

### **程序输入：**

由用户输入一个字符串。

### **程序输出：**

输出输入字符串的长度。

### **输入示例：**

a fox jump fence

### **输出示例：**

16

## 任务：乱飞的机器人

### 问题描述：

编写程序在命令行界面绘制机器人，机器人出现的位置随机，每次出现停留 1 秒钟然后再出现在其他位置。要求将机器人绘制编写成一个函数，参数为机器人出现在屏幕位置的坐标 (x, y 值)，没有返回值；窗口清楚操作也编写成一个函数，完成窗口擦出工作，即将原来出现的机器人用空格代替掉，参数返回值自己确定。

在绘制机器人时要考虑命令行界面的宽度高度，不然绘制的机器人会绘制错误。

### 程序输入：

无。

### 程序输出：

按照问题描述在窗口绘制机器人，实现机器人乱飞动画，机器人形状大小没有固定要求。

### 输入示例：

无。

### 输出示例：

无。

### 解题提示：

1、在程序中添加 `#include <windows.h>`，并在 `main` 函数前添加下面函数。通过此函数可设置命令行界面当前光标位置，例如设置光标位置为第一行第一个字符位置时可通过 `setCursorPosition(0, 0)`；来实现。由于循环执行较快无法看到动画效果，可以在循环中添加 `Sleep(1000)`；来达到延时的效果，参数为毫秒数。

```
/*
 * setCursorPosition()
 * 说明：设置光标当前位置到指定位置
 * 参数： row    行
 *        column  列
 */
void setCursorPosition(int row, int column)
{
    HANDLE hOut;
    COORD pos = {column, row};
    hOut = GetStdHandle(STD_OUTPUT_HANDLE);
    SetConsoleCursorPosition(hOut, pos);
}
```

2、生成随机数需要使用 `rand` 函数和 `srand` 函数，源文件中添加头文件 `stdlib.h` 和 `time.h`。

`srand` 函数是添加随机种子，保证程序每次运行时获取的随机值都是不一样的。

`rand` 函数会根据前面的随机数种子产生一个随机整数。

`rand` 和 `srand` 函数的用法如下：

`srand(time(0));` // 讲当前时间设置为随机数种子，可以将本函数的调用放在 `main` 函数中，一个程序只需要设置一次随机数种子

`int num = rand()%3+1;` // 产生 1~3 的随机数赋值给 `num`，可以将本函数的调

用放在循环里面，没执行一次 rand 函数都会重新生成一个随机数。

## 任务：寻找亚军

### 问题描述：

已知一维整型数组 `int arr[8];`，数组元素由用户输入，编写程序打印比数组元素最大值小的次大元素的下标，有一个符合条件元素则输出一个下标，有多个符合条件元素则输出多个下标。其中，寻找次大元素值的函数原型为：`int findSecondMax(int *p, const int len);`，`p` 为整形数组，`len` 为数组长度，返回值为次大元素的值。

### 程序输入：

由用户输入数组中元素的值，每个元素在输入时用空格隔开。

### 程序输出：

输出次大元素的值，并输出次大元素出现的下标。

### 输入示例：

1 3 8 2 9 5 4 8

### 输出示例：

次大元素为 8，下标为 2 7

## 任务：数组排序

### 问题描述：

已知有一个整形数组 `int arr[7];`，数组元素由用户输入，现在我们要对数组进行从大到小排序，排序的规则如下：

当数组元素为{23,45,12,89,33,101,67}时。

第 1 趟，从这 7 个数中找出最小的数和倒数第一个数交换，排序结果为：

23,45,67,89,33,101,12

第 2 趟，从前 6 个数中找出最小的数和倒数第二个数交换，排序结果为：

101,45,67,89,33,23,12

第 3 趟，从前 5 个数中找出最小的数和倒数第三个数交换，排序结果为：

101,45,67,89,33,23,12

第 4 趟，从前 4 个数中找出最小的数和倒数第四个数交换，排序结果为：

101,89,67,45,33,23,12

第 5 趟，从前 3 个数中找出最小的数和倒数第五个数交换，排序结果为：

101,89,67,45,33,23,12

第 6 趟，从前 2 个数中找出最小的数和倒数第六个数交换，排序结果为：

101,89,67,45,33,23,12

此时数组已经有序，则排序完成。

要求将排序方法编写成一个函数，原型为：`void sort(int * p,const int len);`，`p` 为整形数组，`len` 为数组长度，无返回值，在该函数中完成对数组的排序工作。

### 程序输入：

由用户输入数组中的元素，每个元素之间使用空格隔开，每个元素均为正数。

### 程序输出：

按照顺序输出排好序的数组。

### 输入示例：

23 45 12 89 33 101 67

### 输出示例：

101 89 67 45 33 23 12

### 解题提示：

1、在数组每一趟排序中，找到最小值元素后，记录下当前最小元素的下标，交换时，只需交换指定下标的元素即可。查找最小元素的下标函数原型为：`int findMinIndex(int * p,const int len);`，`p` 为要查找的数组，`len` 为数组长度，返回值为最小值出现的下标。

## 任务：凯撒密码

### 问题描述：

凯撒密码的使用最早可以追溯到古罗马时期，《高卢战记》有描述恺撒曾经使用密码来传递信息，即所谓的“恺撒密码”，它是一种替代密码。通过将字母按顺序推后起 3 位起到加密作用，如将字母 A 换作字母 D，将字母 B 换作字母 E。因据说恺撒是率先使用加密函的古代将领之一，因此这种加密方法被称为恺撒密码。这是一种简单的加密方法，这种密码的密度是很低的，只需简单地统计字频就可以破译。现今又叫“移位密码”，只不过移动的为数不一定是 3 位而已。已知下面的文字是经过“凯撒”密码加密的，请将其解密，并将解密后的内容输出到屏幕上。

```
Rqfh pruh lqwr wkh euhdfk,ghdu iulhqgv,rqfh pruh;  
Ru forvh wkh zdoo xs zlwk rxu Hqjolvk ghdg.  
Lq shdfh wkhuh'v qrwklqj vr ehfrphv d pdq  
Dv prghvw uwlooqhvv dqg kxplolwb:  
Exw zkhq wkh eodvw ri zdu eorzv lq rxu hduv,  
Wkhq lplwdwh wkh dfwlrq ri wkh wljhu;  
Vwliihq wkh vlqhzv,vxpprq xs wkh eorrg.
```

统计英文字符出现的频率函数原型为 `void countFrequency(char *ch,int *frequency);`，编写完成本函数，`char * ch` 为要统计字符串，`int * frequency` 为存放每个字符的出现频率的数组（总共 26 个字符，所以不用给出长度了），没有返回值。

### 程序输入：

无。

### 程序输出：

输出解密后的信息。

### 输入示例：

无。

### 输出示例：

```
Once more into the breach,dear friends,once more;  
Or close the wall up with our English dead.  
In peace there's nothing so becomes a man  
As modest rtillness and humility:  
But when the blast of war blows in our ears,  
Then imitate the action of the tiger;  
Stiffen the sinews,summon up the blood.
```

### 解题提示：

- 1、本任务第 1 个难点在于查找移位个数是多少位，如 'a' 由 'c' 代替，则表示移位个数是 2。
- 2、查找移位个数的方法可以为：先统计当前文本中字符次数出现最多的前几个字符，从网上搜索英文文本中，26 个字母中频次关系，推理求得移位的个数。
- 3、移位时，若移位个数是 2，则 'a' 由 'c' 代替，'b' 由 'd' 代替，.....，'y' 由 'a' 代替，'z' 由 'b' 代替，即实现循环移位。
- 4、文本中的字符，只考虑大写或小写字母的加密操作，其它字符（如标点符号）不考虑。



5、解密成功后，打印原文（一首英文诗）。

## 任务：放鞭炮

### 问题描述：

过年了,刘备、关羽、张飞一起放鞭炮,每一个人一支香,每人初始都为  $N$  个鞭炮(散),刘备每一秒燃放一个,关羽每两秒燃放一个,张飞每三秒燃放一个,在不考虑燃烧时间跟安全的情况下,鞭炮即点即炸,开始的时候大家一起开始放鞭炮,问大家都放完时,一共可以听到多少次响声?一共花了多长时间?

本程序需定义函数实现,函数原型为 `void firecrackers(int n);`,编写完成本函数, `intn` 为每个人初始鞭炮个数,函数中需打印鞭炮响声次数与所需时间,没有返回值。

鞭炮燃放情况可以参考下面示例(鞭炮数量为 3)

时间	刘备	关羽	张飞
1	放	放	放
2	放		
3	放	放	
4			放
5		放	
6			
7			放

### 程序输入：

输入初始鞭炮数。

### 程序输出：

输出鞭炮响声与燃放完所需时间。

### 输入示例：

3

### 输出示例：

time = 7  
count = 6

## 任务：死亡游戏

### 问题描述：

据说著名犹太历史学家 Josephus 有过以下的故事：在罗马人占领乔塔帕特后，39 个犹太人与 Josephus 和他的朋友躲到一个洞中，39 个犹太人决定宁愿死也不要被敌人抓到，于是决定了一个自杀方式，41 个人排成一个圆圈，由第 1 个人开始报数，每报数到第 3 人该人就必须自杀，然后再由下一个重新报数，直到所有人都自杀身亡为止。然而 Josephus 和他的朋友并不想遵从。请你编写一个函数，计算一下 Josephus 和他的朋友应该安排在什么位置，才能免于一死（最后剩下他们两人）。

要求人的总数与报数均由键盘输入，不能指定成 41 和 3。

### 程序输入：

由用户输入总人数与报数值，分两次完成输入，输入的值均为大于 2 的整数，输入时要有提示信息。

要求将求最后剩余人的方法封装成一个函数，p 为一维数组，n 为总人数，m 为报数，函数头为 `void josephus(int * p, const int n, const int m);`，在此函数中求最终幸存者的位置编号并进行打印（位置编号从 1 开始）。

### 程序输出：

输出最后剩余人的号码。

### 输入示例：

请输入人的总数：41

请输入报数值：3

### 输出示例：

Josephus 和他的朋友的位置：16 31

### 解题提示：

1、可以使用数组表示所有人，数组元素值为零，就表示人未出局，如果出局对应值为 1，数组下标可以表示人的编号（编号=下标+1）。数组里最后为 0 的元素所对应的下标+1 就应该是 Josephus 和他的朋友的位置。

数组可以使用 `calloc` 函数进行分配内存，例如。用户输入的总人数为 n，则可使用 `int * arr = (int *)calloc(n, sizeof(int));` 在堆区分配一块内存，指针 arr 当做数组来使用。推荐使用 `calloc` 函数，因为 `calloc` 函数在分配完内存之后会对内存做置 0 操作，相当于 arr 里每一个元素均设置为 0；而 `malloc` 函数则不会进行置 0 操作。使用 `calloc` 函数前需包含 `stdlib.h` 头文件。

2、总人数为 41，报数值为 3 时，出列顺序为：



3、main 函数中的参考代码：

```
int main(void)
{
    int * arr;
    int n, m;
    printf("请输入人的总数：");
    scanf("%d", &n);
    printf("请输入报数值：");
    scanf("%d", &m);
    arr = (int *)calloc(n, sizeof(int));
    josephus(arr, n, m);

    return 0;
}
```

**任务：年龄计算****问题描述：**

有五个学生坐在一起，问第五个人多少岁？答比第四个大 2 岁，第四个人说他比第三个人大 2 岁，...，第一个人说他 10 岁。请写一个程序求第 5 个人的年龄，要求使用递归函数求第 5 个人的年龄。

**程序输入：**

无。

**程序输出：**

输出第 5 个人的年龄。

**输入示例：**

无。

**输出示例：**

18

## **任务：字符串转换为整数**

### **问题描述：**

编写程序，将一个由数字组成的字符串转换成一个整数，例如有字符串 `char * str = "1200"`，使其转换为整型 `1200`。本题目要求使用递归解决，函数原型为 `void StringToInteger(char * str, int * number);`，如果 `str = "1200"`，则调用后 `*number = 1200`。待转换的数字组成的字符串由用户输入。

### **程序输入：**

由用户输入待转换的数字组成的字符串，必须作为字符串进行读取，不允许作为整型读取。

### **程序输出：**

输出转换后的数字。

### **输入示例：**

1200

### **输出示例：**

1200

### 任务：最大公约数（递归）

#### 问题描述：

编写程序求两个正整数的最大公约数。要求使用递归，函数原型为 `int gcd(int m, int n);`，`m`、`n` 为要求最大公约数的两个整数，返回值为最大公约数。

#### 程序输入：

有用户输入两个大于 1 的整数，输入时两个数之间使用空格隔开。

#### 程序输出：

输出用户所输入两个整数的最大公约数。

#### 输入示例：

9 6

#### 输出示例：

3

#### 解题提示：

- 1、如果  $m > n$ ，则  $m$  和  $n$  的最大公约数与  $m - n$  和  $n$  的最大公约数相同。
- 2、如果  $n > m$ ，则  $m$  和  $n$  的最大公约数与  $m$  和  $n - m$  的最大公约数相同。
- 3、如果  $n == m$ ，则  $m$  和  $n$  的最大公约数即为  $m$  或  $n$ 。

## 任务：两点之间距离

### 问题描述：

编写函数求两个坐标点之间的距离。要求自己定义结构体表示点 ( Point ), 然后编写函数求两个点之间的距离, 参数为两个 Point ( 变量 ), 函数返回这两个点之间的距离。

### 程序输入：

由用户分两次输入两个坐标点的信息, 每次输入包括点的 x、y 坐标, 中间使用空格隔开, 坐标均为浮点数。

### 程序输出：

输出两个点之间的距离, 要求小数点后面保留两位有效数字。

### 输入示例：

```
0 0
3 4
```

### 输出示例：

```
5.00
```

### 解题提示：

- 1、表示一个点的 x 坐标与 y 坐标就可以表示这一点。
- 2、关于程序中需要用到的一些数学函数, 需要在代码中添加 math.h 头文件, 具体函数使用方法请参考 math.h 百度百科 <http://baike.baidu.com/subview/673537/673537.htm>。



## 任务：点到直线距离

### 问题描述：

编写函数求一个坐标点到一条直线的距离。要求自己定义结构体表示点 ( Point )、线 ( Line )，然后编写函数求点到直线的距离，参数 1 为 Point ( 变量 )，参数 2 为 Line ( 变量 )，函数返回点到直线的距离。

### 程序输入：

由用户分两次分别输入点信息与直线信息，点信息包括点的 x、y 坐标，输入时中间添加空格隔开；直线信息包括直线二元一次方程中的三个参数，每个参数之间使用空格隔开。点信息与直线信息均为浮点数类型。

### 程序输出：

输出点到直线的距离，要求小数点后面保留 2 位有效数字。

### 输入示例：

```
0 0
5 5 5
```

### 输出示例：

```
0.71
```

### 解题提示：

- 1、表示一个点的 x 坐标与 y 坐标就可以表示这一点。
- 2、表示一条直线，可以使用直线的二元一次方程  $Ax+By+C=0$  中的三个参数来表示一条直线。
- 3、关于程序中需要用到的一些数学函数，需要在代码中添加 math.h 头文件，具体函数使用方法请参考 math.h 百度百科 <http://baike.baidu.com/subview/673537/673537.htm>。

## 任务：点是否在圆内

### 问题描述：

编写函数求一个坐标点是否在一个圆内。要求自己定义结构体表示点 ( Point )、圆 ( Circle )，然后编写函数求点是否在圆内，参数 1 为 Point( 变量 )，参数 2 为 Circle( 变量 )，函数返回一个整数表示点是否在圆上，0 代表在圆外，1 代表在圆上，2 代表在圆内。

### 程序输入：

由用户分两次分别输入点信息与圆信息，点信息包括点的 x、y 坐标，输入时中间添加空格隔开；圆信息包括圆心点的 x、y 坐标，以及圆的半径长度，共三个数值，每个数值之间使用空格隔开。点信息与圆信息均为浮点数类型。

### 程序输出：

根据点信息与圆信息输出点与圆的位置关系。

### 输入示例：

```
0 0
0 0 1
```

### 输出示例：

圆内

### 解题提示：

- 1、表示一个点的 x 坐标与 y 坐标就可以表示这一点。
- 2、表示一个圆，可以使用圆心的点信息与半径长度来表示一个圆。
- 3、关于程序中需要用到的一些数学函数，需要在代码中添加 math.h 头文件，具体函数使用方 法 请 参 考 math.h 百 度 百 科

<http://baike.baidu.com/subview/673537/673537.htm>

## 任务：点是否在矩形内

### 问题描述：

编写函数求一个坐标点是否在一个矩形内。要求自己定义结构体表示点( Point )、矩形( Rect )，然后编写函数求点是否在矩形内，参数 1 为 Point ( 变量 )，参数 2 为 Rect ( 变量 )，函数返回点一个整数表示点是否在矩形内，0 代表在矩形外，1 代表在矩形上，2 代表在矩形内。

### 程序输入：

由用户分两次分别输入点信息与矩形信息，点信息包括点的 x、y 坐标，输入时中间添加空格隔开；矩形信息包括矩形左上角与右下角两个角的点的 x、y 坐标共四个数值，每个数值之间使用空格隔开。点信息与矩形信息均为浮点数类型。

### 程序输出：

根据点信息与矩形信息输出点与矩形的位置关系。

### 输入示例：

```
0 0
0 0 1 -1
```

### 输出示例：

矩形上

### 解题提示：

- 1、表示一个点的 x 坐标与 y 坐标就可以表示这一点。
- 2、表示一个矩形，可以使用矩形左上角与右下角两个角的点来表示一个矩形。
- 3、关于程序中需要用到的一些数学函数，需要在代码中添加 math.h 头文件，具体函数使用方 法 请 参 考 math.h 百 度 百 科

<http://baike.baidu.com/subview/673537/673537.htm>

## 任务：两圆是否相交

### 问题描述：

编写函数求两个圆是否相交。要求自己定义结构体表示点 ( Point ) 圆 ( Circle ) , 然后编写函数求两个圆是否相交, 参数 1 与参数 2 均为 Circle ( 变量 ), 函数返回一个整数表示两个圆是否相交, 0 代表不相交, 1 代表相交。

### 程序输入：

由用户分两次分别输入两个圆的信息, 圆信息包括圆心点的 x、y 坐标, 以及圆的半径长度, 共三个数值, 每个数值之间使用空格隔开。圆信息均为浮点数类型。

### 程序输出：

输出两个圆的位置关系。

### 输入示例：

```
0 0 1
1 0 1
```

### 输出示例：

相交

### 解题提示：

1、表示一个圆, 可以使用圆心的点信息与半径长度来表示一个圆。

2、关于程序中需要用到的一些数学函数, 需要在代码中添加 math.h 头文件, 具体函数使用方法请参考 math.h 百度百科  
<http://baike.baidu.com/subview/673537/673537.htm>

## 任务：学生排序

### 问题描述：

现有一个班级的学生成绩,要求自定义结构体存储学生信息,学生信息包括学生姓名与学生成绩。然后编写函数完成对学生信息按照成绩从大到小的排序,函数原型为 `void sort(STU * aStu, int n);`, `aStu` 代表数组的头指针, `n` 代表数组的长度。

学生数组里面的学生数据信息自己填写 ( ), 数组长度自定义, 学生信息结构体如下:

```
typedef struct student
{
    char name[20];
    int score;
} STU;
```

### 程序输入：

无。

### 程序输出：

将排序好的学生信息按顺序输出,每行输出一条学生信息,每条学生信息包括学生姓名、学生成绩,中间使用空格隔开。

### 输入示例：

无。

### 输出示例：

```
关羽 95
张飞 90
刘备 80
```

## 任务：学生查找

### 问题描述：

现有一个班级的学生成绩，要求自定义结构体存储学生信息，学生信息包括学生姓名与学生成绩。然后编写函数完成对指定姓名的学生信息的查找，找到后打印学生信息，函数原型为 `STU * find(STU * aStu, int n, char * aName);`，`aStu` 代表数组的头指针，`n` 代表数组的长度，`aName` 代表要查找学生的名字，如果找到则返回该学生的结构体的指针，没有找到返回 `NULL`。

学生数组里面的学生数据信息自己填写，数组长度自定义，不存在重名学生，学生信息结构体如下：

```
typedef struct student
{
    char name[20];
    int score;
} STU;
```

### 程序输入：

由用户输入学生姓名，按照字符串读取。

### 程序输出：

将查找到的学生信息输出，学生信息包括学生姓名、学生成绩，中间使用空格隔开。如果没有找到则打印“没有找到”。

### 输入示例：

例 1     张飞  
例 2     李逵

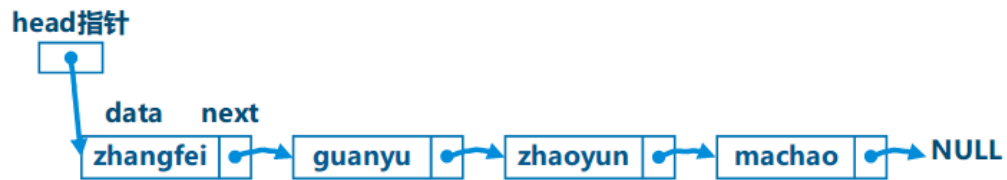
### 输出示例：

例 1     张飞 95  
例 2     没有找到

## 任务：链表信息打印

### 问题描述：

现有如下链表，链表中每个节点均保存有信息，编写程序定义出次链表，并编写函数将链表中的 data 部分全部打印出来。



节点结构体如下：

```
typedef struct node
{
    char data[20];
    struct node * next;
} Node;
```

打印函数原型为：

```
void printList(Node * head);
```

### 程序输入：

无。

### 程序输出：

输出链表中的 data 数据。

### 输入示例：

无

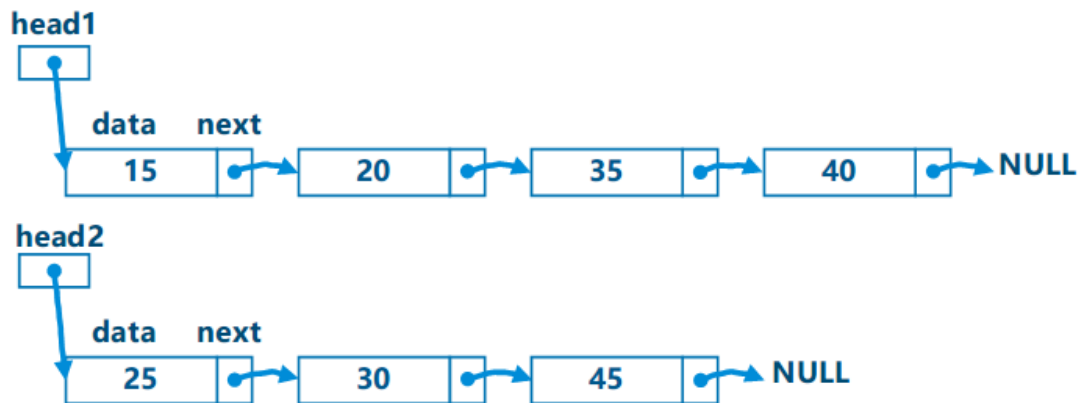
### 输出示例：

```
zhangfei
guanyu
zhaoyun
machao
```

## 任务：链表合并

### 问题描述：

现有如下两个按照从小到大排好顺序的链表，链表中每个节点均保存有信息，编写程序定义出次链表，并编写函数将两个链表中的数据组合成一个新的链表，要求新链表中数据满足从小到大的顺序。



节点结构体如下：

```
typedef struct node
{
    int data;
    struct node * next;
} Node;
```

函数原型为：

```
Node * mergeList(Node * h1, Node * h2);
```

### 程序输入：

无。

### 程序输出：

输出合并后链表中的数据。

### 输入示例：

无

### 输出示例：

10 20 25 30 35 40 45



## 任务：学生信息管理

### 问题描述：

编写程序完成一个简单的学生信息管理系统，该系统能够完成简单的学生信息增、删、改、查、打印功能，根据用户的选择来完成不同的功能。系统要由良好的信息提示，帮助用户正确使用系统。本系统需按照解题提示中要求完成。

本系统运行效果参考如下（粗体部分为用户输入部分）：

```
=====
                        学生信息管理系统
=====
1 增加学生信息          2 删除学生信息          3 修改学生信息
4 查找学生信息          5 打印学生信息          0 退出系统
请输入您要进行的操作：1
请输入新增学生姓名及成绩（使用空格隔开）：张飞 80
增加成功！
请输入您要进行的操作：1
请输入新增学生姓名及成绩（使用空格隔开）：关羽 90
增加成功！
请输入您要进行的操作：1
请输入新增学生姓名及成绩（使用空格隔开）：刘备 70
增加成功！
请输入您要进行的操作：2
请输入要删除的学生姓名：赵云
没有赵云信息！
请输入您要进行的操作：2
请输入要删除的学生姓名：刘备
删除刘备信息成功！
请输入您要进行的操作：3
请输入要修改的学生姓名：刘备
没有刘备信息！
请输入您要进行的操作：3
请输入要修改的学生姓名：张飞
请输入要修改的学生成绩：75
信息修改成功！
请输入您要进行的操作：4
请输入要修改的学生姓名：张飞
张飞 75
请输入您要进行的操作：5
张飞 75
关羽 90
```

请输入您要进行的操作：0

再见！

**程序输入：**

程序输入如问题描述示例中一致，当输入学生姓名与成绩时，中间使用空格隔开。

**程序输出：**

程序输入如问题描述示例中一致。

**输入示例：**

无。

**输出示例：**

无。

**解题提示：**

1、程序中使用链表存储学生信息，使用的学生节点信息结构体如下：

```
typedef struct student
{
    char name[20];
    int score;
    struct student * next;
} STU;
```

2、创建带头结点的链表的函数原型为：

STU \* createdList(char \* aName, int score)。aName 是要插入学生的名字，score 是要插入学生的分数，返回值为链表的头结点地址。

3、在链表末尾添加新结点的函数原型为：

int insertRecordFromBack(STU \* aHead, char \* aName, int score)。

aHead 是链表的头指针，aName 是要插入学生的名字，score 是要插入学生的分数，添加成功返回值为 1，否则为 0。

4、删除链表中指定结点的函数原型为：

int deleteRecordFromList(STU \* aHead, char \* aName)。aHead 是链表的头指针，aName 代表要删除学生的名字，删除成功返回值为 1，否则为 0。

5、在链表中查找节点并打印该结点信息的函数原型为：

int findAndPrint(STU \* aHead, char \* aName)。aHead 代表链表的头指针，aName 代表要查找学生的名字，找到该学生的个人信息并打印成功返回 1，未找到返回 0。

6、在链表中查找节点并修改该结点的函数原型为：

int findAndEdit(STU \* aHead, char \* aName, int aScore)。aHead 代表链表的头指针，aName 代表要查找学生的名字，int aScore 代表该学生的新成绩，找到该学生的个人信息并修改成功返回 1，未找到返回 0。

7、打印链表中所有学生信息的函数原型为：

void printList(STU \* aHead)。aHead 代表链表的头指针。