

一、

/\*\*

\*某城市出租车计费有以下标准:

\*①起步里程 3km , 起步费 10 元

\*②10km 以内, 若超起步里程外, 每千米 2 元

\*③超过 10km 以上部分加收 50%回空补贴 , 每千米 3 元

\*④因堵车等因素发生的临时停车, 每 5 分钟 2 元, 不足 5 分不收费

\*⑤要求输入里程&等待时间 求车费

\* ⑥ 验 证 :  $\text{time}=0; 8\text{km}=10+(8-3)*2=20$  元

$12\text{km}=10+(10-3)*2+(12-10)*3=30$  元

\*  $15\text{km}=10+(10-3)*2+(15-10)*3=39$  元

\*\*/

#include <stdio.h>

int main(){

printf("请输入里程\n");

double k,yuan,time,time\_yuan; //定义公里, 金额, 等待时间

double ten\_k\_yuan=10+(10-3)\*2;//10km 的金额

scanf("%lf",&k);

printf("请输入等待时间\n");

scanf("%lf",&time);

if(time<5)

time\_yuan=0;

```
else{
    time_yuan=(int(time))/5*2;
}
if(k>0&& k<=3){
    yuan=10+time_yuan;
}
else if(k>3&& k<=10){
    yuan=(k-3)*2+10+time_yuan;
}
else if(k>10){
    yuan=(k-10)*3+ten_k_yuan+time_yuan;
}
printf("%.2lf\n",yuan);
}
```

二、

1! +2! +...+20!,求阶乘用递归。

```
/**
```

```
* 求 1! +2! +...+20!
```

```
* 记住，阶乘用递归
```

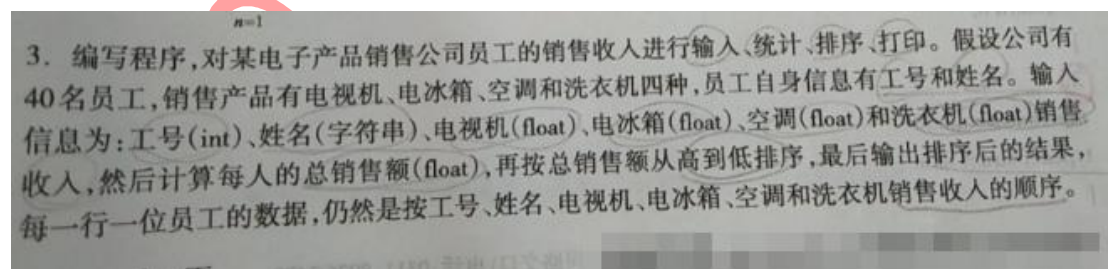
```
**/
```

```
#include <stdio.h>
```

```
int fun(int x)
```

```
{  
  
    int a;  
  
    if(x<1) a=1;  
  
    else a=x*fun(x-1); //1 1*fun(0)=1*1  
                        //2 2*fun(1)=2*1*fun(0)  
  
    return a;  
}  
  
int main(){  
  
    int num=20,sum=0;  
  
    for(int i=1;i<=num;i++) // 20 循环  
  
        sum=sum+fun(i);  
  
    printf("%d",sum);  
}
```

三、



```
/**
```

```
* 涉及到结构体、冒泡排序
```

```
**/
```

```
#include <stdio.h>
```

```
struct yg
```

```
{
```

```
int num;
```

```
char name[20];
```

```
float dsj, dbx, kt, xiyiji, zj;
```

```
};
```

```
int main()
```

```
{
```

```
struct yg arr[3], temp;
```

```
int i, j, max;
```

```
for ( i = 0; i < 3; i++ )
```

```
{ printf("请输入第%d 个员工的编号、名字、电视机金额、电冰箱金  
额、空调金额、洗衣机金额\n",i+1);
```

```
scanf("%d%s%f%f%f", &arr[i].num, arr[i].name,  
&arr[i].dsj, &arr[i].dbx, &arr[i].kt, &arr[i].xiyiji );
```

```
printf("-----  
---\n");
```

```
arr[i].zj = arr[i].dsj + arr[i].dbx + arr[i].kt + arr[i].xiyiji;
```

```
}
```

```
for ( i = 0; i < 3; i++ )
```

```
printf( "%d\t%s\t%5.2f\t%5.2f\t%5.2f\t%5.2f\t 总 金 额 : %5.2f\n",
arr[i].num,arr[i].name, arr[i].dsj, arr[i].dbx, arr[i].kt,arr[i].xiyiji, arr[i].zj );
for ( i = 0; i < 3 - 1; i++ )
{
    max = i;
    for ( j = i + 1; j < 3; j++ )
        if ( arr[j].zj > arr[max].zj )
            max = j;
    if ( max != i )
        temp = arr[i], arr[i] = arr[max], arr[max] = temp;
}
printf("-----\n");
printf( "排列:\n " );
for ( i = 0; i < 3; i++ )
    printf("%d\t%s\t%5.2f\t%5.2f\t%5.2f\t%5.2f\t      总      金
额 : %5.2f\n",arr[i].num,arr[i].name, arr[i].dsj, arr[i].dbx, arr[i].kt,
arr[i].xiyiji,arr[i].zj );
return 0;
}
```

四、

三天打鱼两天晒网，第 N 天是打鱼还是晒网？

```
/**
```

```
* 三天打鱼，两天晒网，第 N 天是打鱼还是晒网？
```

```
* 举例：第 6 天打鱼，第 9 天晒网，第 100 天晒网
```

```
**/
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int day;
```

```
    printf("请输入第几天：\n");
```

```
    scanf("%d",&day);
```

```
    if(day>0&&day<=3){
```

```
        printf("第%d 天在打鱼\n",day);
```

```
    } else if(day>3&&day<=5){
```

```
        printf("第%d 天在晒网\n",day);
```

```
    }else if(day>5){
```

```
        if(day%5==0){
```

```
            printf("第%d 天在晒网\n",day);
```

```
        }else if(day%5!=0){
```

```
            if((day%5)>=1&&(day%5)<=3){
```

```
                printf("第%d 天在打鱼\n",day);
```

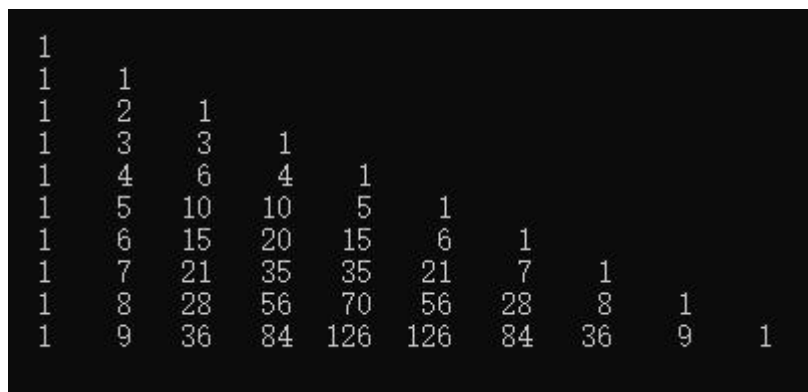
```
            }else if((day%5)>3&&(day%5)<5){
```

```
                printf("第%d 天在晒网\n",day);
```

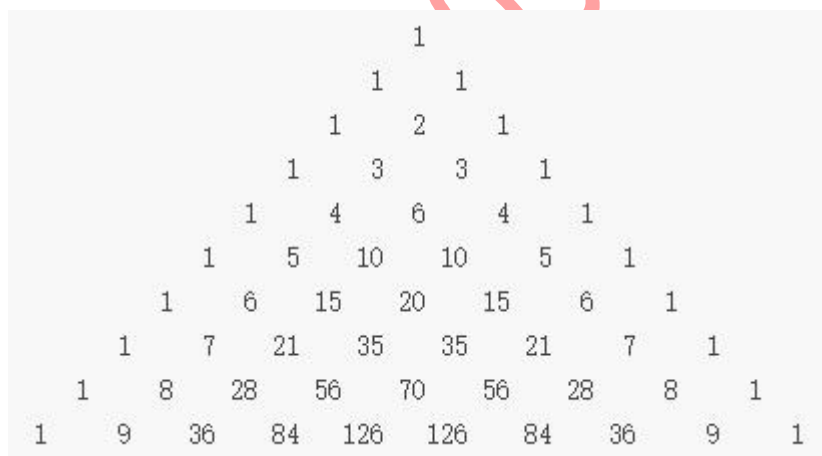
```
    }  
    }  
}  
  
return 0;  
  
}
```

五、

杨辉三角



1									
1	1								
1	2	1							
1	3	3	1						
1	4	6	4	1					
1	5	10	10	5	1				
1	6	15	20	15	6	1			
1	7	21	35	35	21	7	1		
1	8	28	56	70	56	28	8	1	
1	9	36	84	126	126	84	36	9	1



				1					
			1		1				
		1		2		1			
	1		3		3		1		
1		4		6		4		1	
	1	5		10		10		5	
		6		15		15		6	
	1	7		21		21		7	
		8		28		28		8	
1		9		36		36		9	1

/\*\*

- \* 杨辉三角 在初中,我们就知道,杨辉三角的两个腰边的数都是 1,
- \* 其它位置的数都是上顶上两个数之和。
- \* 这就是我们用 C 语言写杨辉三角的关键之一。
- \*

```
*/  
  
#include<stdio.h>  
  
int main()  
{  
  
    int i,j, a[10][10];  
  
    printf("\n");  
  
    for(i=0;i<10;i++){  
        a[i][0]=a[i][i]=1;  
    }  
  
    for(i=2;i<10;i++){  
        for(j=1;j<i;j++){  
            a[i][j]=a[i-1][j-1]+a[i-1][j];  
        }  
    }  
  
    for(i=0;i<10;i++){  
        for(j=0;j<=i;j++){  
            printf("%5d",a[i][j]);  
        }  
        printf("\n");  
    }  
  
    return 0;  
}
```

六、

输入多个字符串，然后输出最短的字符串。



```
/**
 * 输入多个字符串，然后输出最短的字符串。
 *
 **/

#include <stdio.h>

#include <string.h>

int main(){

    char a[10][30]; //10 行，每行最多放 30 个字符
    int i,count[10],temp;//count[10]是储存每一行字符串个数
    printf("请输入 10 串字符串\n");
    printf("\n");
    printf("-----\n");
    for (i=0;i<10;i++){

        printf("请输入第%d 串字符串\n",i+1);
        printf("\n");
        gets(*(a+i)); //列指针*(a+i)和 a[i]等价,这行代码就是一行行复
赋值
        count[i]=strlen(*(a+i));//提取每行字符串的个数，存入数组

        /*每个字符串的长度已经放到数组里，接下来
        采用冒泡排序

        */

        printf("-----\n");
```

```
}

for(i=0;i<9;i++){ //变量 i 代表比较的趟数

    for(int j=0;j<10-i;j++){//变量 j 代表每趟两两比较的次数

        if(count[j]<count[j+1]){

            temp=count[j]; //利用中间变量实现两值互换

            count[j]=count[j+1];

            count[j+1]=temp;

        }

    }

}

printf("以下是字符串长度排序之后的结果\n");

for(i=0;i<10;i++){ //测试：冒泡排序，把每个串的长度排大
小

    printf("%d\n",count[i]);

}

printf("\n");

printf("-----\n");

//找字符串长度最短的数组

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

    if(strlen(*(a+i))==count[9]){

        printf("最短的字符串为第%d 次输入的字符串\n",i+1);
```

```

        break;

    }

}

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

    if(strlen(*(a+i))==count[9]){

        printf("%s\n",a[i]);

        break;

    }

}

return 0;

}

```

冒泡排序:

```

for(i=0;i<9;i++){ //变量i代表比较的趟数
    for(int j=0;j<10-i;j++){//变量j代表每趟两两比较的次数
        if(count[j]<count[j+1]){
            temp=count[j]; //利用中间变量实现两值互换
            count[j]=count[j+1];
            count[j+1]=temp;
        }
    }
}

```

```

23 | for(i=0;i<9;i++){ //变量i代表比较的趟数
24 |     for(int j=0;j<10-i;j++){//变量j代表每趟两两比较的次数
25 |         if(count[j]<count[j+1]){
26 |             temp=count[j]; //利用中间变量实现两值互换
27 |             count[j]=count[j+1];
28 |             count[j+1]=temp;
29 |         }
30 |     }
31 | }

```

七、

三个球，大小一样，其中有一个质量和其他球质量不同，请找出不同

的球。

```
/**
```

```
* 三个球，大小一样，其中有一个质量和其他球质量不同，请找出  
不同的球。
```

```
* 也就是 3 个数，有两个数一样，找另一个数
```

```
**/
```

```
#include <stdio.h>
```

```
int main(){
```

```
    int arr[3]={1,3,1},temp,i,j;
```

```
    for(i=0;i<2;i++){
```

```
        for(j=0;j<3-i;j++){
```

```
            if(arr[j]<arr[j+1]){
```

```
                temp=arr[j];
```

```
                arr[j]=arr[j+1];
```

```
                arr[j+1]=temp;
```

```
            }
```

```
        }
```

```
    for(i=0;i<3;i++){//测试
```

```
        printf("%d\n",arr[i]);
```

```
    }
```

```
printf("\n=====\\n");

if(arr[0]==arr[1]){

    printf("%d\\n",arr[2]);

} else if(arr[2]==arr[1]){

    printf("%d\\n",arr[0]);

}

return 0;

}
```

八、

/\*\*

\* 有 1000 人捐款，捐到 10 万就停止捐款。统计捐款人数，再求平均数

\*\*/

```
#include <stdio.h>
```

```
int main(){
```

```
    float money,ave,sum=0;
```

```
    int i;
```

```
    for (i=1,sum=0;i<=3;i++){
```

```
        printf("请输入金额:");
```

```
        scanf("%f",&money);
```

```
        sum= sum+money;
```

```
        if (sum>=100000) break;

    }

    if(i<3) {

        ave=sum/i ;

    }else if(i>3)

        ave=sum/(i=i-1);

    printf("捐款人数: %d\t 平均数: %.2f\n",i,ave);

    return 0;

}
```

九、

采用递归，把整形转化成字符串

递归：外层需要用到内层算出的结果，每次进入更深一层递归时，问题规模相比上一次递归都应有所减少。

```
/**
 * 有采用递归，把整形转化成字符串
 **/

#include <stdio.h>

void toString(int n){    // 例如传入，即 n=1238

    if(n/10!=0){        // 1238/10!=0

        toString(n/10);    /* 执行 toString(1238/10=123)即 n=123
```

123/10!=0

执行 toString(123/10=12)即 n=12

12/10!=0

执行 toString(12/10)即 n=1

1/10=0

不执行 toString(1/10)

\*/

```
printf("\n====%d====\n",n/10);//    测试
}
printf("%c",n%10+'0');
}
int main(){
    int n;
    printf("输入一个整数: \n");
    scanf("%d",&n);
    printf("字符串是: ");
    if(n<0){
        printf("-");
        n=-n;
    }
    toString(n);
    return 0;
```

```
}
```

十、

```
/**
```

\* 根据火车的出发时间和到达时间，编写程序，计算整个旅途时间

\* 输入两个 4 位数 hhmm hhmm--> hh:mm 表示时间

\* 取值范围[0~23] mm 取值范围[0~59]

\* 出发和到达在同一天内

```
**/
```

```
#include <stdio.h>
```

```
int main(){
```

```
    int startTime,endTime,hourTime,minuteTime;
```

```
        printf("请输入出发时间（例如 12:52 输入 1252）\n");
```

```
        scanf("%d",&startTime);
```

```
        printf("请输入到达时间（例如 22:52 输入 2252）\n");
```

```
        scanf("%d",&endTime);
```

```
    /*既然是同一天，结束时间的小时数肯定大于开始时间
```

```
    *结束小时 21，开始小时 12
```

```
    *题目要求输入 4 位数，比如 22:52-->2152,2252/100=22,可以求出
```

小时差

```
    *分钟差：22:52-->2152，如何获得 52 呢？2152%100 正好得分钟
```

数



\*但是 22:13-12:52 怎么算呢? 小学的时候学过, 不够就向小时借 1 (60 分钟)

\*即为 21:73-12:52 就可以算分钟差

\*例如 22:13-12:52

\*先小时相减,  $22-12=10$ , 然后分钟相减  $13-52=-39$

\*为了保证时间的准确性: 小时-1= $10-1=9$ ; 分钟+60= $-39+60=21$

\*/

//分别算 22:52-12:52 和 22:12-12:52

```
hourTime=endTime/100-startTime/100;
```

```
minuteTime=endTime%100-startTime%100;
```

```
if(minuteTime<0){
```

```
    hourTime=hourTime-1;
```

```
    minuteTime=60+minuteTime;
```

```
}
```

```
printf("%02d:%02d",hourTime,minuteTime);
```

//2 是宽度很简单。如果整数不够 2 列就补上 0

```
//printf("%02d",3);结果就是 03
```

```
return 0;
```

```
}
```

十一、

输入一行字符串，求数字，大小写字母，空格，其他字符各多少

```
/**
```

\*输入一行字符串，求数字，大小写字母，空格，其他字符各多少

```
**/
```

```
#include <stdio.h>
```

```
int main(){
```

```
    char ch;
```

```
    int upper=0, lower=0, number=0, space=0, others=0;
```

```
    printf("请输入一行字符:");
```

```
    while((ch=getchar())!='\n')
```

```
        if (ch>='0' && ch<='9')
```

```
            number++;
```

```
        else if (ch>='A' && ch<='Z')
```

```
            upper++;
```

```
        else if (ch>='a' && ch<='z')
```

```
            lower++;
```

```
        else if (ch==' ')
```

```
            space++;
```

```
        else
```

```
            others++;
```

```
    printf("数字字符: %d\n 大写字母: %d\n 小写字母: %d\n 空  
格: %d\n 其他字符: %d\n", number, upper, lower, space, others);
```

```

    return 0;
}

```

十二、

编写 `char *str (char *s, char x)`，其功能：在给定的字符串 `s` 寻找一个特定的字符 `x`，如果找到，返回 `x` 在 `s` 的第一次出现的地址，在主函数中把 `s` 中该字符串和该字符串之前的字符逆序输出

### 指针函数

```
int *fun(int, int);
```

由于“\*”的优先级低于“()”的优先级，因而 `fun` 首先和后面的“()”结合，也就意味着，`fun` 是一个函数。即：`int *(fun(int, int));`

其返回值是一个 `int` 类型的指针，是一个地址。

```
/**
```

\*十二、编写 `char *str (char *s, char x)`，其功能：在给定的字符串 `s`

\*寻找一个特定的字符 `x`，如果找到，返回 `x` 在 `s` 的第一次出现的地址，

\*在主函数中把 `s` 中该字符串和该字符串之前的字符逆序输出

```
**/
```

```
#include <stdio.h>
```

```
char *str(char *s,char x){
```

```
    int ij=-1; // i 用来做循环，j 用来做判断是否找到相应的字符
```

```
    for(i=0;s[i]!='\0';i++){ //abcd'\0'    x=c
```

```
        if(x==s[i]){ //开始匹配，逐个对比，找相同
```

```
            ++j;        //找到相同 j 不为-1
```

```
        break;

    }

}

if(j==-1){    //没找到, ++j 不执行, 返回 NULL

    return NULL;

}else{

    return &(s[i]); // 返回地址

}

}

int main(){

    char s[40], *p, x;

    int i, j;

    printf("请输入一串字符\n"); //abcd'\0'

    gets(s);

    printf("请输入一个字符\n");

    x=getchar();

    p=str(s,x);

    if(p==0){

        printf("\n 你输入的字符未找到");

    }

    else{

        printf("你输入的字符串为: \n");//abcd'\0'
```

```
for(j=0;s[j]!='\0';j++)

    printf("%c",s[j]);

for(i=0;s[i]!='\0';i++){

    if(x==s[i]){

        printf("\n");

        printf("\n 找到匹配的字符,它在第%d 位,值为: %c\n",i+1,s[i]);

        printf("地址为: %d\n",p);

        printf("逆序为\n");

        for(i;i>=0;i--){      //abcd'\0'

            printf("%c",s[i]);

        }

        break;

    }

}

return 0;

}
```

十三、

求分段函数  $f(x)$  ,  $1/x, x \neq 0; 0, x = 0;$

$f(100) = 1/100; f(7) = 1/7; f(0) = 0$

```
/**
 * 十求分段函数 f (x) , 1/x, x!=0;0,x=0;
 **/

#include <stdio.h>

int main(){

    float x;

    printf("请输入一个值: \n");

    scanf("%f",&x);

    if(x==0){

        printf("%d",0);

    }

    else{

        printf("%.4f",1/x);

    }

    return 0;

}
```

十四、

水仙花数,  $153=1*1*1+5*5*5+3*3*3$

```
/*
*水仙花数,  $153=1*1*1+5*5*5+3*3*3$ 
*
```

```
*/  
  
#include <stdio.h>  
  
int main(){  
    int i,j,k,n;  
  
    printf("water flower number is:");  
  
    for(n=100;n<1000;n++){  
        i=n/100; /*分解出百位*/  
        j=n/10%10; /*分解出十位*/  
        k=n%10; /*分解出个位*/  
        if(i*100+j*10+k==i*i*i+j*j*j+k*k*k)  
            printf("%d\t",n);  
    }  
  
    return 0;  
}
```

十五、

给一个数，求它的素数因子。

$$\begin{array}{r} 2 \overline{) 10} \\ \underline{5} \end{array} \quad 2, 5$$

---

$$\begin{array}{r} 2 \overline{) 24} \\ \underline{24} \\ 2 \overline{) 12} \\ \underline{12} \\ 2 \overline{) 6} \\ \underline{6} \\ 3 \end{array} \quad 2, 2, 2, 3$$

```
#include <stdio.h>
```

```
//判断是否为素数
```

```
int fun(int num){
```

```
    int a=0; // 素数的个数
```

```
    for(int i=2;i<num;i++){
```

```
        if(num%i==0){
```

```
            a++; // 如果 if 执行, 证明不是素数
```

```
        }
```

```
    }
```

```
    if(a==0){
```

```
        printf("%d 是素数, 不用因式分解\n", num);
```

```
        return 1;
```

```
    }else{
```

```
        printf("%d 不是素数。 \n", num);
```

```
        return 0;
```

```
    }
```



```
}
```

```
int main(){
```

```
    int n,i,j,k=0,x;
```

```
    printf("输入一个大于 2 的整数: ");
```

```
    scanf("%d",&n);
```

```
    i=fun(n);
```

```
    printf("\n-----测试 i 返回值 i=%d-----\n",i);
```

```
    //一个数（素数除外），均可由若干素数相乘组成。
```

```
    //例如 15=3*5    28=2*2*7
```

```
    if(i==0){
```

```
        int j;
```

```
        printf("\n 则%d 素数为: \n",n);
```

```
        while(n){
```

```
            for(j=2;j<=n;j++)
```

```
                if(n%j==0){                //n=9 当j=3 时，if 执行
```

```
                    printf("%d ",j);
```

```
                    n=n/j;                //n=9/3=3
```

```
                    break;
```

```
                    //在多层循环中，一个 break 语句只向外跳一层
```

```
            }
```

```
        }
```

```
    }  
  
    return 0;  
  
}
```

十六、

对两个数整数进行加减乘除取余，且除取余分母不为 0；

/\*\*

\* 对两个数整数进行加减乘除取余，且除取余分母不为 0；

\*/

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int a,b;
```

```
char c;
```

```
printf("请输入你的操作，例如 100+200\n");
```

```
scanf("%d%c%d",&a,&c,&b);
```

```
switch(c){
```

```
    case '+':
```

```
        printf("%d\n",a+b);
```

```
        break;
```

```
    case '-':
```

```
        printf("%d\n",a-b);
```

```
        break;
```

```
    case '*':
```

```
        printf("%d\n",a*b);
```

```
        break;
```

```
    case '/':
```

```
        if(b!=0){
```

```
            printf("%d\n",a/b);
```

```
            break;
```

```
        }
```

```
    case '%':
```

```
        if(b!=0){
```

```
            printf("%d\n",a%b);
```

```
            break;
```

```
        }
```

```
    default :
```

```
        printf("分母不能为 0\n");
```

```
        break;
```

```
return 0;
}
}
```

十七、

求  $s=a+aa+aaa+aaaa$  四个数相加，其中  $a$  和几个数相加由用户决定。

```
/*
```

\*求  $s=a+aa+aaa+aaaa$  四个数相加，其中  $a$  和几个数相加由用户决定。

```
*/
```

```
#include <stdio.h>
```

```
int main(){
```

```
    //例如 22 是由  $2+2*10$     第二个数
```

```
    //222 是由  $2+2*10+2*100$  第三个数
```

```
    //主要是对 2 进行处理，定义一个变量  $num=num+scanInNumb$ ，先对 2 存储，
```

```
    第一次对 2 求和， $sum=sum+ num$ 
```

```
    思路：主要是要求和结果
```

```
    之后的输入的 2 依次乘以 10，可以得到 20,200,2000....(用  $scanInNumb$  记)
```

```
     $scanInNumb$  然后加到  $num$  上，第二次  $num=2+20=22$ ，之后  $sum=2+22$ 。以此类推
```

```
    */
```

```
    int scanInNumb,count,i,num=0,sum=0;    //定义输入的数和个数
```

```
        //i 为控制循环， $num$  为对每个数操作
```

```
        //sum 是累加
```

```
    printf("请输入数： \n");
```

```
    scanf("%d%d",&scanInNumb,&count); // 例如 2,个数： 3-->2+22+222
```

```
    for(i=1;i<=count;i++){
```

```
        num=num+scanInNumb;    //num= 0+2① num=2+20②
```

```
        sum=sum+num;    //sum=0+2① sum=22+2②
```

```
         $scanInNumb=scanInNumb*10$ ; //scanInNumb=2*10=20① scanInNumb=200
```

```
    }
```

```
    printf("sum=%d\n",sum);
```

```
    return 0;
```

```
}
```

十八、

已经定义 `void swap(*int a,*int b)`（功能是交换），实现输入  $n$  个整数，然后升序排列。

```
/*
```

\*已经定义 `void swap(*int a,*int b)`（功能是交换）

\*实现输入  $n$  个整数，然后升序排列。

```
*
```

```
*/
```

```
#include<stdio.h>
```

/\*思考, 输入一串整数, 考虑到了定义数组  
然后给处理数组, 以后见到数组, 就想到指针 (地址)  
既然是排序, 首先想起冒泡排序  
关于 void swap(\*int a,\*int b)函数, 没有 return 语句,  
因为地址是关联着, 在 swap () 里的变化, 也就在 main () 一一对应  
\*/

```
void swap(int *a,int *b){
    int temp;
    temp=*a;
    *a=*b;
    *b=temp;
}

int main(){

    int array[50],num,i,j;
    printf("请输入数据的个数(<50): ");
    scanf("%d",&num); //要进行 num 个数的输入
    printf("请输入%d 个元素的值:\n",num);
    for(i=0;i<num;i++) //数组初始化
        scanf("%d",&array[i]);
    for(i=0;i<num;i++) //冒泡排序
        for(j=i+1;j<num;j++)
            if(array[j]<array[i])
                swap(&array[j],&array[i]); //if 语句进行交换, 调用 swap ()
    printf("升序排序的结果:\n");
    for(i=0;i<num;i++) //输出
        printf("%d ",array[i]);
    printf("\n");
}
```

十九、

高速公路达到或超出限速的 10%处罚 200 元, 输出 Exceed x%.Ticket 200

达到或超出限速的 50%则吊销驾驶证。输出 Exceed x%.License Revoked

其中 x 是超速的百分比, 精确到整数。

正常行驶输出 OK, , , 分别输入车速和限速

/\*\*

\* 高高速公路达到或超出限速的 10%处罚 200 元, 输出 Exceed x%.Ticket 200

\* 达到或超出限速的 50%则吊销驾驶证。输出 Exceed x%.License Revoked

\* 其中 x 是超速的百分比, 精确到整数。

\* 正常行驶输出 OK, , , 分别输入车速和限速

\*\*/

#include <stdio.h>

int main(){

/\*分析, 限速 100, 意味车子不能达到 110 或超过 110

超过 100 的 10%就是  $100 \times 110\% \rightarrow 100 \times (1+0.1) = 110$  得出具体速度

超过 50% 类似算法

此时拿当前车速与超速分别比较。没超过的输出 OK

当车速超出 10%，超出的百分比（数学题）： $(\text{车子速度} - \text{限速}) / \text{限速}$

\*/

```
int carSpeed,speedLimit,n,m;//车速、限速
scanf("%d%d",&carSpeed,&speedLimit);
n=speedLimit*(1+0.1);//达到或者超速 10% 小于这个速度，不被处罚
m=speedLimit*(1+0.5);//达到或者超速 50%
if(carSpeed<n)
    printf("OK\n");
else if(carSpeed<m) //超速范围[10%,50%)，罚款
    printf("Exceed %.0f%%. Ticket 200\n",1.0*(carSpeed-speedLimit)/speedLimit*100);
else //超速范围[50%,+∞)，吊销驾驶证
    printf("Exceed %.0f%%. License Revoked\n",1.0*(carSpeed-speedLimit)/speedLimit*100);
return 0;
}

二十、
输入全班 30 个学生的成绩，成绩由高到低排序，主函数要求输入和输出，子函数排序
/*
*输入全班 30 个学生的成绩，成绩由高到低排序
*主函数要求输入和输出，子函数排序
*
*/
#include<stdio.h>
void swap(int *a,int *b){
    int temp;
    temp=*a;
    *a=*b;
    *b=temp;
}
int main(){
```

```
    int array[30],num,i,j;
    printf("要给多少个学生录入成绩? ");
    scanf("%d",&num);//要进行 num 个数的输入
    for(i=0;i<num;i++){
        printf("请输入第%d 个学生的分数:\n",i+1);
        scanf("%d",&array[i]);
    } //数组初始化
    for(i=0;i<num;i++) //冒泡排序
        for(j=i+1;j<num;j++)
```

```

        if(array[j]<array[i])
            swap(&array[j],&array[i]); //if 语句进行交换，调用 swap ()
    printf("升序排序的结果:\n");
    for(i=num-1;i>=0;i--) //输出
        printf("%d ",array[i]);
    printf("\n");
}

```

二十一、

编写函数，int IsSquare (long int n) 功能判定完全平方数，若一个数能表示成某个数平方形式，则为完全平方数

/\*

\*求编写函数，int IsSquare (long int n) 功能判定完全平方数，

\*若一个数能表示成某个数平方形式，则为完全平方数

\*/

#include <stdio.h>

#include <math.h>

/\*

\*分析：现在需要 math.h 头文件，调用 sqrt () 函数（算术平方根）

\*如果一个数开根号，前提是强制类型转换为 int，例如 i=8，int（根号下 8）

结果为 2，然后 2\*2!=8，则 8 不是完全平方数。

\*/

```

int IsSquare(long int n);

```

```

int main(){

```

```

    long n,p;

```

```

    scanf("%d",&n);

```

```

    if(p=IsSquare(n))

```

```

        printf("%d 是平方数,%d*d=%d\n",n,p,p,n);

```

```

    else

```

```

        printf("%d 不是平方数\n",n);

```

```

    return 0;

```

```

}

```

```

int IsSquare(long int n){

```

```

    int m;

```

```

    m=(int)sqrt(n);

```

```

    if(m*m==n)

```

```

        return m;

```

```

    else

```

```

        return 0;

```

```

}

```

二十二、

生活问题，交水费 y 元，可以用 x 吨水，x 不超过 15 吨时，y=4x/3，超过后，y=2.5-17.5。

/\*\*

\* 生活问题，交水费 y 元，可以用 x 吨水，x 不超过 15 吨时，y=4x/3，

\* 超过后,  $y=2.5x-17.5$ 。

\*/

```
#include <stdio.h>
```

```
int main(){
    float x,y;
    scanf("%f",&x);
    if(x>=0 && x<=15)
        y=4*x/3;
    else
        y=2.5*x-17.5;
    printf("%.2f\n",y);
    return 0;
}
```

二十三、

一条蠕虫长 1 寸, 其 1 分钟可以向上爬 U 寸, 但是他得休息, 休息的时候下滑 D 寸, 问什么时候爬出去? 不足 1 分钟按照 1 分钟计算, 蠕虫头部到达井的顶部, 则完成任务, 开始蠕虫趴在井底 (高度为 0)

/\*

\*一条蠕虫长 1 寸, 其 1 分钟可以向上爬 U 寸, 但是他得休息, 休息的时候下滑 D 寸,

\*问什么时候爬出去? 不足 1 分钟按照 1 分钟计算, 蠕虫头部到达井的顶部,

\*则完成任务, 开始蠕虫趴在井底 (高度为 0)

\*

\*/

```
#include <stdio.h>
```

/\*

\*分析: 可以用循环和判断来解决这道题。

\*蠕虫每向上爬一次, 离井口就越近,

\*而井长 N 就越小, 当  $N \leq 0$  的时候, 就输出。

\*当爬到最后的寸数小于 U 时, 这时候只需再爬一次就到井口了, 而不用休息

\*/

```
int main(){
    int N,U,D;
    int time=0;
    scanf("%d%d%d",&N,&U,&D); //井长、上爬距离、下滑距离
    while(N>0){
        if(N>U){ //若井长大于上爬距离
            time=time+2; //爬一分钟, 休息一分钟
        }
        else{
            time=time+1; //爬一分钟, 不休息
        }
        if(N<=U){ //若井长小于等于爬的距离
            N = N-U; //虫子头部到井口的距离
        }
    }
}
```

```

    }
    else{
        N = N-U+D;//虫子头部到井口的距离+下滑距离
    }
}
printf("%d",time);
return 0;
}

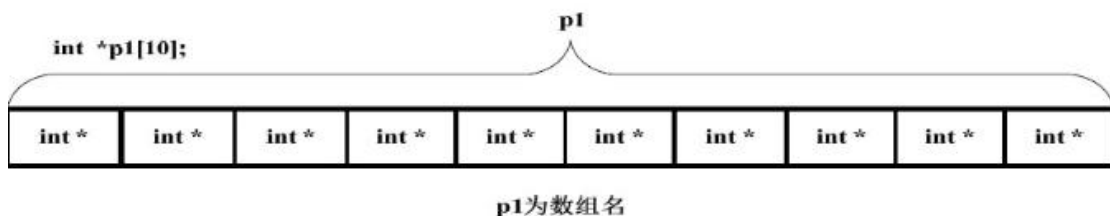
```

二十四、

编写函数 `void sort(char *words[], int n)`，对一组单词进行字典排序，在主函数输出原序列和排序后的序列。

注：指针数组：首先它是一个数组，数组的元素都是指针，数组占多少个字节由数组本身决定。它是“储存指针的数组”的简称

`int *p1[10]; sizeof strlen`



`strcmp` 函数是 `string compare`(字符串比较)的缩写，用于比较两个字符串并根据比较结果返回整数。基本形式为 `strcmp(str1,str2)`，若 `str1=str2`，则返回零；若 `str1<str2`，则返回负数；若 `str1>str2`，则返回正数。

abc ef

```

/*
*求编写函数 void sort(char *words[], int n),
*对一组单词进行字典排序，在主函数输出原序列和排序后的序列
*/
#include <stdio.h>
#include <string.h>
void sort(char *words [], int n);
int main(){
    char *wString[]={"cde","cab","foo", "fab","cba "};
    int i, n=5;
    printf("原序列:\n");
    for (i=0; i<n; i++)
        printf ("\twString[%d]=%s\n", i, wString[i]);
    printf("排列之后的序列:\n");
    sort(wString,n); /* 调用函数，对指针数组 wString 中的 n 个字符串排序 */
    for (i=0; i<n; i++)
        printf ("\twString[%d]=%s\n", i, wString[i]);
}

```



/\* 对指针数组 s 中的 n 个字符串按字典排序 \*/

```
//{"cde","cab","foo","fab","cba"}
void sort(char *s[], int n){
    char *temp;
    int i,j,k;
    for (i=0; i<n-1; i++){
        k=i;
        for (j=i+1; j<n; j++)
            if (strcmp(s[k],s[j])>0)
                k=j;
        if (k!=i){
            temp=s[i];
            s[i]=s[k];
            s[k]=temp;
        }
    }
}
```

二十五、

生活问题，50 度（含）电以内，0.53 元/度，超过 50 度，电价上调 0.05 元/度

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int dl;
```

```
    float df;
```

```
    scanf("%d",&dl);
```

```
    if(dl<0) printf("Invalid Value!");
```

```
    else if(dl<=50) df=0.53*dl;
```

```
    else df=0.53*50+(dl-50)*(0.53+0.05);
```

```
    printf("%.2f",df);
```

```
return 0;
```

```
}
```

二十六、

任何比 2 大的偶数，总能表示两个素数，比如 24=5+19；5 和 19 都是素数

```
/*
```

\*任何比 2 大的偶数，总能表示两个素数，比如 24=5+19；5 和 19 都是素数

```
*
```

```
*/
```

```
#include<stdio.h>
```

```
#include<math.h>
```

```
int fun(int num){
```

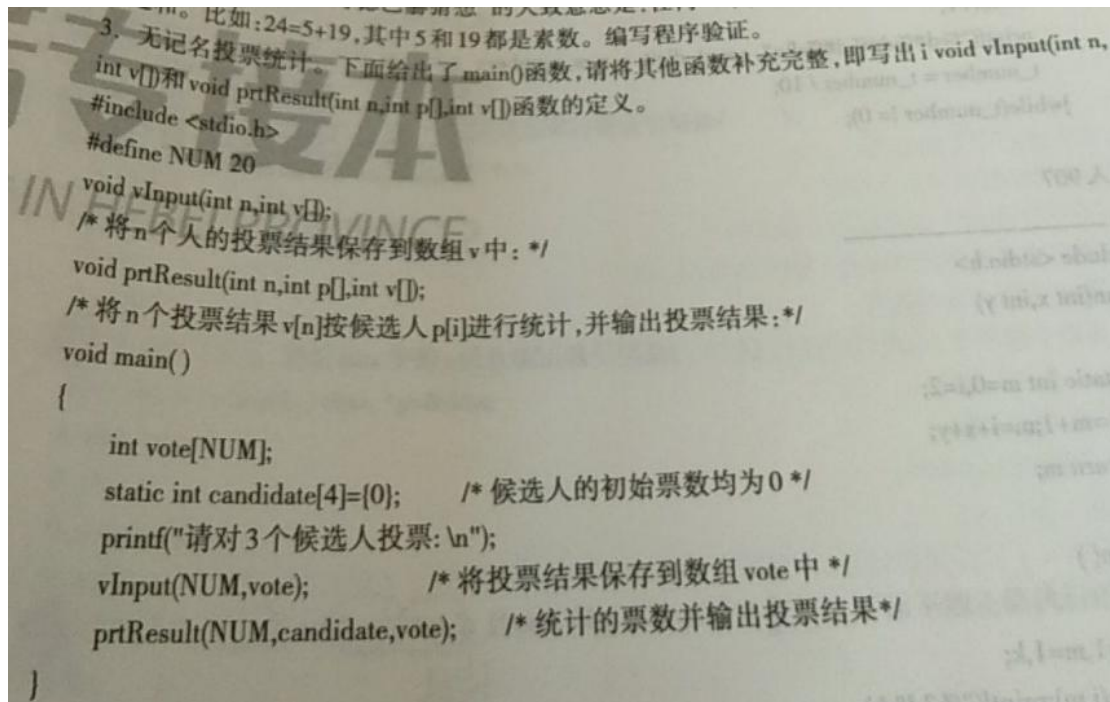
```
    int a=0;
```

```
    // 判断是否为素数
```

```
for(int i=2;i<num;i++){
    if(num%i==0){
        a++; // 如果 if 执行，证明不是素数
    }
}
if(a==0){
    return 1;//返回 1，是素数
}else{
    return 0;//返回 0，不是素数
}
}
int main(){
    int n,i,a,b;
    scanf("%d",&n);
    for(i=2;i<=n;i++){
        //返回 1，是素数
        //返回 0，不是素数
        //8=3+5
        //测试代码
        if((a=(fun(i)))==1&&(b=(fun(n-i)))==1){
            printf("%d=%d+%d\n",n,i,n-i);
            printf("a=%d\tb=%d\n",a,b);
            break;
        }
        //返回 1，是素数
        //返回 0，不是素数
        //8=3+5
        //源代码 for(i=2;i<=n;i++)

        if(fun(i)==1&&fun(n-i)==1){
            printf("%d=%d+%d\n",n,i,n-i);
            break;
        }
    }
    return 0;
}
```

二十七、



```
#include <stdio.h>
#define NUM 6
//6 个人给 3 个人投票
//初始化数组
void vInput(int n, int v[]);
//统计票数, 输出结果
void prtResult(int n, int p[], int v[]);
int main(){
    int vote[NUM];
    int candidate[3]={0}; // 候选人的初始票数均为 0
    printf("请对 3 个候选人投票: \n");
    vInput(NUM, vote); //初始化数组, 将投票结果保存到数组 vote
    prtResult(NUM, candidate, vote); //统计的票数并输出投票结果
}
/* 将 6 个人的投票结果保存到数组 v 中: */
void vInput(int n, int v[]){
    int i;
    for(i=0; i<NUM; i++){
        scanf("%d", &v[i]); //初始化数组, 对候选人投票
        if(v[i]<1 || v[i]>3) //三个人的编号为 1,2,3, 输入之外的编号为无效
            v[i]=0; /* 统计无效票数 */
    }
}
/* 将 n 个投票结果 v[n]按候选人 p[i]进行统计, 并输出投票结果: */
void prtResult(int n, int p[], int v[]){
    //n=6, p[]为三个竞选者, v[]为所投的票数
    int i;
```

```

for(i=0;i<n;i++)
    ++p[v[i]]; // i 的范围 0-6, (正常投票) v[i] 的值就是 1,2,3
    // ++p[1], ++p[2], ++p[3]
printf("候选人\t\t 得票数\n");
printf("-----\n");
for(i=1;i<4;i++)
    printf(" %d\t\t %d\n", i, p[i]);
printf("弃权票:\t\t %d\n", p[0]);
}

```

二十八、

现在 90#汽油 6.95 元/升, 93#汽油 7.44 元/升, 97#汽油 7.93 元/升, 加油站推出了“自助服务”和“协助服务”两个服务等级, 分别可得到 5%和 3%的折扣, 根据输入顾客的加油量 a, 汽油品种 b (90、93、97) 和服务类型 c (m-自助, e-协助), 计算应付金额。

/\*

现在 90#汽油 6.95 元/升, 93#汽油 7.44 元/升,

97#汽油 7.93 元/升, 加油站推出了

“自助服务”和“协助服务”两个服务等级,

分别可得到 5%和 3%的折扣, 根据输入顾客的加油量 a, 汽油品种 b (90、93、97) 和服务类型 c (m-自助, e-协助), 计算应付金额

\*/

```
#include <stdio.h>
```

```
int main(){
```

```
    float je, jg;
```

```
    int a, b;
```

```
    char c;
```

```
    // a 代表油量 b 代表品种 c 代表服务类型
```

```
    scanf("%d,%d,%c", &a, &b, &c);
```

```
    switch(b){
```

```
        case 90: jg=6.95; break;
```

```
        case 93: jg=7.44; break;
```

```
        case 97: jg=7.93; break;
```

```
    }
```

```
    if(c=='m')
```

```
        je=jg*a-jg*a*0.05;
```

```
    else
```

```
        je=jg*a-jg*a*0.03;
```

```
    printf("%.2f\n", je);
```

```
    return 0;
```

```
}
```

二十九、

编写程序, 求两个给定正整数的最大公约数和最小公倍数。

/\*

\*编写程序，求两个给定正整数的最大公约数和最小公倍数

\*如果数 a 能被数 b 整除，a 就叫做 b 的倍数，b 就叫做 a 的约数

\*/

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int a,b,i,min,max;    //a、b 存储读入的两个正整数
```

```
    scanf("%d%d",&a,&b);
```

```
//举两个例子 10 和 11 10 和 20
```

```
    max = 1;
```

```
    for(i=2;i<=a;i++)
```

```
    {
```

```
        if(a%i==0 && b%i==0)
```

```
            max = i;
```

```
    }
```

```
    min = a*b;
```

```
    for(i=a*b-1;i>=a;i--)
```

```
    {
```

```
        if(i%a==0 && i%b==0)
```

```
            min = i;
```

```
    }
```

```
    printf("最大公约数: %d\t 最小公倍数: %d\n",max,min);    //先后输出最大公因数和最
```

```
小公倍数
```

```
    return 0;
```

```
}
```

三十、

用结构体来描述一个学生的成绩情况，内容包括：姓名、学号、C 成绩、微机原理成绩、对一个班 30 同学统计。

求同学平均分、分数从高到低排名，打印出名字，打印出 90 以上和不及格的同学名字

```
#include <stdio.h>
```

```
#define N 4
```

```
struct stu{
```

```
    char name[20];
```

```
    int no;
```

```
    float C;
```

```
    float WJYL;
```

```
    float avg;
```

```
};
```

```
int main(){
```

```
    int i,j;
```

```
struct stu a[N],t;
//输入信息，求平均分
for(i=0;i<N;i++){
    scanf("%s%d%f",a[i].name,&a[i].no,&a[i].C,&a[i].WJYL);
    a[i].avg=(a[i].C+a[i].WJYL)/2;
}
//排序
for(i=0;i<N-1;i++)
    for(j=0;j<N-i-1;j++)
        if(a[j].avg<a[j+1].avg){
            t=a[j];
            a[j]=a[j+1];
            a[j+1]=t;
        }
//输出排序后
for(i=0;i<N;i++)
    printf("%s,%f\n",a[i].name,a[i].avg);
//输出 90 分以上
for(i=0;i<N;i++)
    if(a[i].avg>90)
        printf("90 分以上的: %s\n",a[i].name);
//输出不及格的
for(i=0;i<N;i++)
    if(a[i].avg<60)
        printf("不及格的: %s\n",a[i].name);
return 0;
}
```

三十一、

给定平面上任意三个点的坐标  $(x_1,y_1)$ 、 $(x_2,y_2)$ 、 $(x_3,y_3)$  编写程序，是否能构成三角形，不能构成输出 No，如果可以，输出三角形周长和面积

/\*

\*了解两点之间的距离公式

\*了解求三角形的公式

\*/

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int main(){
```

```
    float x1,x2,x3,y1,y2,y3;
```

```
    float a,b,c,s,l,area;
```

```
    scanf("%f%f%f%f%f%f",&x1,&y1,&x2,&y2,&x3,&y3);
```

```
    a=sqrt((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2));
```

```
    b=sqrt((x1-x3)*(x1-x3)+(y1-y3)*(y1-y3));
```

```
    c=sqrt((x3-x2)*(x3-x2)+(y3-y2)*(y3-y2));
```

```
if(a+b>c && a+c>b && b+c>a){
    s=(a+b+c)/2;
    area=sqrt(s*(s-a)*(s-b)*(s-c));
    l=a+b+c;
    printf("L = %.2f, A = %.2f\n",l,area);
}
else printf("No\n");
return 0;
}
```

三十二、

编写函数 `int strcmp (char *p1,char *p2)` ,实现两个字符串的比较

/\*

\*编写函数 `int strcmp (char *p1,char *p2)` ,实现两个字符串的比较

\*/

```
#include <stdio.h>
```

```
int strcmp(char *p1,char *p2){
    int i=0;
    while(*(p1+i)==*(p2+i))// *(p+i)=a[p+i]=a[i]
        if (*(p1+i++)=='\0')
            return 0;
    return 1;
}
```

```
int main(){
    int m;
    char str1[20],str2[20];
    printf("输入两个串:\n");
    scanf("%s",str1);
    scanf("%s",str2);
    m=strcmp(str1,str2);
    if(m==0)
        printf("字符串相同");
    else
        printf("字符串不同");
    return 0;
}
```

三十三、

10 个评委打分，采用百分制，先计算评委的分数之和，减去最高分和最低分，再除以 8 得到成绩，输入 1-10 个评委的分数，计算学生的最终得分。请写出 `input()`、`ssum()`、`smax()`、`smin()` 函数的定义

```
#include
```

```
int main()
```

```
{ int score[10],max,min,sum;
```

```
float aver;
```

```
input(score, 10); //输入 10 个评委的评分
```

```
sum=ssum(score,10); //计算 10 个评委的评分之和
max=smax(score,10); //计算最高分
min=smin(score,10); //计算最低分
aver=(sum-max-min)*1.0/(10-2); //计算学生的最终得分
printf("%f\n",aver); //输出学生的最终得分
return 0;
}
```

---

//代码:

```
#include <stdio.h>
```

```
//10 个人的分数传入 10
```

```
void input(int s[], int n){
```

```
    int i;
```

```
    for(i=0;i<n; i++) scanf("%d",&s[i]);
```

```
}
```

```
int ssum(int s[], int n){
```

```
    int st=0,i;
```

```
    for(i=0; i<n; i++)
```

```
        st+=s[i];
```

```
    return st;
```

```
}
```

```
int smax(int s[], int n){
```

```
    int max, i;
```

```
    max=s[0];
```

```
    for(i=1;i<n;i++){
```

```
        if(max<s[i]) max=s[i];
```

```
    }
```

```
    return max;
```

```
}
```

```
int smin(int s[], int n){
```

```
    int min, i;
```

```
    min=s[0];
```

```
    for(i=1;i<n;i++){
```

```
        if(min>s[i])
```

```
            min=s[i];
```

```
    }
```

```
    return min;
```

```
}
```

```
int main(){
```

```
    int score[10],max,min,sum;
```

```
    float aver;
```

```
    input(score, 10); //输入 10 个评委的评分
```

```
    sum=ssum(score,10); //计算 10 个评委的评分之和
```



```

max=smax(score,10); //计算最高分
min=smin(score,10); //计算最低分
aver=(sum-max-min)*1.0/(10-2); //计算学生的最终得分
printf("%f\n",aver); //输出学生的最终得分
return 0;
}

```

三十四、

求一元二次方程的根， $ax^2+bx+c=0$ ,  $a, b, c$  自己输入

```
#include <stdio.h>
```

```
#include <math.h>
```

```
/*
```

思路，首先判断根与系数的关系

$<0$  无解

$>0$  两个不同的解

$=0$  两个相同的解

如果大于等于 0  $x1=-b/2a+判别式/2a$   $x2=-b/2a-判别式/2a$

```
*/
```

```
int main (){
```

```
double a,b,c,disc,x1,x2,p,q;
```

```
scanf("%lf%lf%lf",&a,&b,&c);
```

```
disc=b*b-4*a*c;
```

```
if (disc<0)
```

```
printf("无根\n");
```

```
else{
```

```
p=-b/(2.0*a);
```

```
q=sqrt(disc)/(2.0*a);
```

```
x1=p+q;
```

```
x2=p-q;
```

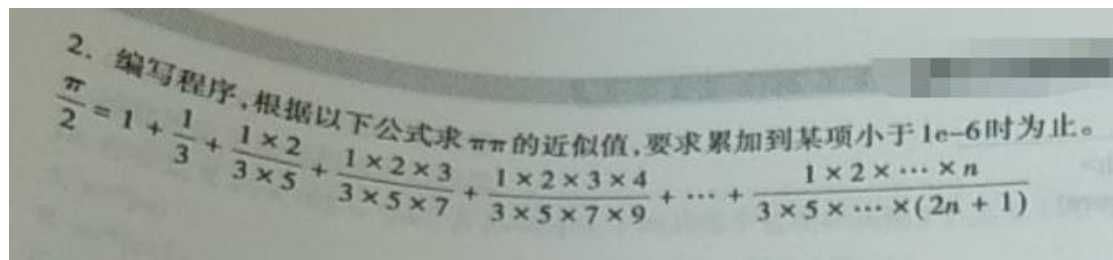
```
printf("根为:\nx1=%7.2f\nx2=%7.2f\n",x1,x2);
```

```
}
```

```
return 0;
```

```
}
```

三十五、



```
#include <stdio.h>
```

```
int main(){
```

```
float i,j,n,m,sum;
```

```
n=1.0;
m=1.0;
i=0.0;
j=1.0;
sum=0.0;
while(n/m>=1e-6){
    sum+=n/m;
    i++;
    j=j+2;
    n=n*i;
    m=m*j;
}
printf("pi=%f\n",2*sum);
return 0;
}
```

三十六、

定义一个函数 `s(char *s1, char *s2)`, 把两个字符串连接起来, 不能用 `strcat` () 函数

```
#include<stdio.h>
```

```
void s(char *s1, char *s2){
    int i=0,j=0;
    while (s1[i]!='\0')
        i++;
    while(s2[j]!='\0')
        s1[i++]=s2[j++];
    s1[i]='\0';
}
```

```
int main(){
    char s1[80],s2[40];
    printf("input string1:");
    scanf("%s",s1);
    printf("input string2:");
    scanf("%s",s2);
    s(s1,s2);
    printf("\nThe new string is:%s\n",s1);
    return 0;
}
```

三十七、

编写程序, 要求输出的百分制分数段, A 为 85 分以上, B 为 70~84 分, C 为 60~69 分, D 为 60 以下, 成绩由键盘输入

```
#include <stdio.h>
```

```
int main(){
    char grade;
    scanf("%c",&grade);
```

```
printf("Your score:");
switch(grade){
    case 'A': printf("85~100\n");break;
    case 'B': printf("70~84\n");break;
    case 'C': printf("60~69\n");break;
    case 'D': printf("<60\n");break;
    default: printf("enter data error!\n");
}
return 0;
}
```

三十八、

先在有 10 个城市名以及春夏秋冬的平均温差，编写程序，定义结构体，输入城市名称、四季温度以及全年平均温度。

```
#include <stdio.h>
struct aa{
    char city[10];
    float spring;
    float summer;
    float autumn;
    float winter;
    float avgtem;
};
int main(){
    struct aa x[10],t;
    int i,j;
    for(i=0;i<10;i++)
        scanf("%s%f%f%f%f",x[i].city,&x[i].spring,&x[i].summer,&x[i].autumn,&x[i].winter);
    for(i=0;i<10;i++)
        x[i].avgtem=(x[i].spring+x[i].summer+x[i].autumn+x[i].winter)/4;
    for(i=0;i<9;i++){
        for(j=0;j<9-i;j++){
            if(x[j].avgtem>x[j+1].avgtem){
                t=x[j];
                x[j]=x[j+1];
                x[j+1]=t;
            }
        }
    }
    for(i=0;i<10;i++)

    printf("%s,%f,%f,%f,%f,%f\n",x[i].city,x[i].spring,x[i].summer,x[i].autumn,x[i].winter,x[i].avgtem);
    return 0;
}
```

三十九、

设计一个函数 `int isprime (int x)`，判断是否为素数，是返回 1，否返回 0，在主函数中调用 `isprime (int x)` 函数，输出 100-999 之间的各位数为 7 的所有素数之和。

```
#include<stdio.h>
int isprime(int num){
    int a=0; // 最后判断素数
    for(int i=2;i<num;i++){
        if(num%i==0){
            a++; // 如果 if 执行，证明不是素数
        }
    }
    if(a==0){
        return 1;//素数返回 1
    }else{
        return 0;
    }
}
int main(){
    int i,s=0;
    for(i=100;i<=999;i++){
        if(i%10==7){
            if(isprime(i))
                s=s+i;
        }
    }
    printf("s=%d\n",s);
    return 0;
}
```

四十、

编写 N 个学生的成绩（百分制），统计 N 个学生的等级分布。

大于等于 90 的是 A，小于 90 大于等于 80 的是 B，小于 80 大于等于 70 的是 C，小于 70 大于等于 60 的是 D，小于 60 的是 E。

```
#include <stdio.h>
int main(){
    int N,score,A,B,C,D,E,i;
    A=B=C=D=E=0;
    scanf("%d",&N);
    for(i=1;i<=N;i++){
        scanf("%d",&score);
        switch(score/10){
            case 10:case 9: A++; break;
            case 8: B++;break;
```

```

        case 7: C++;break;
        case 6: D++;break;
        default: E++;break;
    }
}
printf("%d %d %d %d %d\n",A,B,C,D,E);
return 0;
}

```

四十一、

```

    *  *
  *    *
 *      *
 *      *
    *  *

```

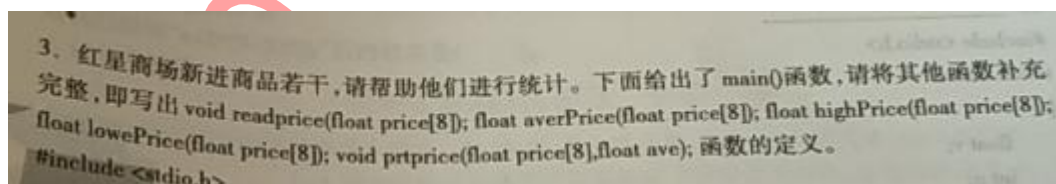
输出空心菱形。

```

#include<stdio.h>
#include<string.h>
int main(){
    char a[5][14]={"    **    ","    *    *    ","    *    *    ","    *    *    ","    *    *    "};
};
for(int i=0;i<5;i++){
    printf("%s",a[i]);
    printf("\n");
}
}

```

四十二、



```

#include <stdio.h>
void readprice(float price[8]); /*输入商品的价格*/
float averPrice(float price[8]); /*计算商品的平均价格*/
float highPrice(float price[8]); /*找出最高价的商品*/
float lowPrice(float price[8]); /*找出最低价的商品*/
void prtprice(float price[8],float ave); /*输出高于平均价格的商品*/
int main(){
    float price[8];
    float average,highestP,lowestP;
    readprice(price); /*输入商品的价格*/
}

```

```
        average=averPrice(price); /*计算商品的平均价格*/
        highestP=highPrice(price); /*找出最高价的商品*/
        lowestP=lowePrice(price); /*找出最低价的商品*/
        printf("最高价格的物品=%6.2f\n",highestP);
        printf("最低价格的物品=%6.2f\n",lowestP);
        printf("平均价格=%6.2f\n",average);
        prtprice(price,average); /*输出高于平均价格的商品*/
        return 0;
    }
    /*-----*/
    /*输入商品的价格*/
    void readprice(float price[8]){
        int i;
        printf("请输入 8 个物品的价格:\n");
        for (i=0;i<8;i++)
            scanf("%f",&price[i]);
        printf("这个 8 个物品的价格分别是 :\n");
        for (i=0;i<8;i++)
            printf("%6.2f\t",price[i]);
        printf("\n");
        return;
    }
    /*-----*/
    /*计算商品的平均价格*/
    float averPrice(float price[8]){
        float sum=0.0;
        float average;
        int i;
        for (i=0;i<8;i++)
            sum=sum+price[i];
        average=sum/8;
        return average;
    }
    /*-----*/
    /*找出最高价的商品*/
    float highPrice(float price[8]){
        float highest;
        int i;
        highest=price[0];
        for (i=0;i<7;i++)
            if (highest<price[i+1])
                highest=price[i+1];
        return highest;
    }
```

```
/*-----*/
/*找出最低价的商品*/
float lowePrice(float price[8]){
    float loweest;
    int i;
    loweest=price[0];
    for (i=0;i<7;i++)
        if (loweest>price[i+1])
            loweest=price[i+1];
    return (loweest);
}
/*-----*/
/*输出高于平均价格的商品*/
void prtprice(float price[8],float average){
    int i;
    printf("高出平均价格的物品是:\n");
    for (i=0;i<8;i++)
        if (price[i]>average)
            printf("%.2f\t",price[i]);
    return;
}
```

四十三、

编写一段程序，计算某年某月某日是该年的第几天？

```
#include<stdio.h>
int main(){
    int y,m,d,day;
    printf("请输入年月日，例如 2020 年 1 月 1 日输入 2020-01-01\n");
    scanf("%d-%d-%d",&y,&m,&d);
    switch (m){
        case 1:day=0; break ;
        case 2:day=31; break ;
        case 3:day=59; break ;
        case 4:day=90; break ;
        case 5:day=120; break ;
        case 6:day=151; break ;
        case 7:day=181; break ;
        case 8:day=212; break ;
        case 9:day=243; break ;
        case 10:day=273; break ;
        case 11:day=304; break ;
        case 12:day=335; break ;
    }
    day=day+d;
}
```

```
if ((m%400==0)|| (m%4==0)&&(m%100==0))
    day=day+1;

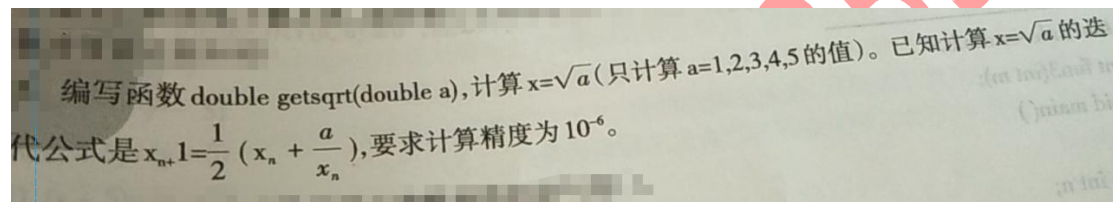
printf("\n%d-%d-%d 是%d 年得第%d 天\n",y,m,d,y,day);

}
```

四十四、

一对兔子，从出生后的第 3 个月起，每个月都要生一对兔子，小兔子长到 3 个月之后每个月又生一对兔子，假如兔子都不死，请问第一个月出生的一对兔子，至少繁衍到几个月时兔子总数才可以达到 N 对？

四十五、



编写函数 double getsqrt(double a), 计算  $x=\sqrt{a}$  (只计算  $a=1,2,3,4,5$  的值)。已知计算  $x=\sqrt{a}$  的迭代公式是  $x_{n+1}=\frac{1}{2}(x_n+\frac{a}{x_n})$ , 要求计算精度为  $10^{-6}$ 。

迭代公式就是指用现在的值，代到一个公式里面，算出下一个值

再用下一个值代入公式，如此往复地代。

比如

$$x=(x+2/x)/2$$

你随便拿一个  $x=10$  代入，得  $x=(10+2/10)/2=5.1$

再代进去  $x=(5.1+2/5.1)/2=2.746$

再代入得 1.737

再代得 1.444

再代得 1.414525655

再代得 1.414213597

再代得 1.414213562



```
#include <stdio.h>
#include <math.h>
double getsqrt(double a);
int main(){
    double x[5]={1,2,3,4,5},xsqrt[5];
    int i;
    for(i=0;i<5;i++){
        xsqrt[i]=getsqrt(x[i]);
        printf("%12.9f\n",xsqrt[i]);
    }
    return 0;
}
double getsqrt(double a){
    double x1=a,x2;
    while(1){
        x2=(x1+a/x1)/2;
        if(fabs(x1-x2)<1e-6)
            break;
        x1=x2;
    }
    return x1;
}
```