

第1章 引言

1、对 C 语言来说，下列标识符中哪些是合法的，哪些是不合法的？

total, _debug, Large&Tall, Counter1, begin_

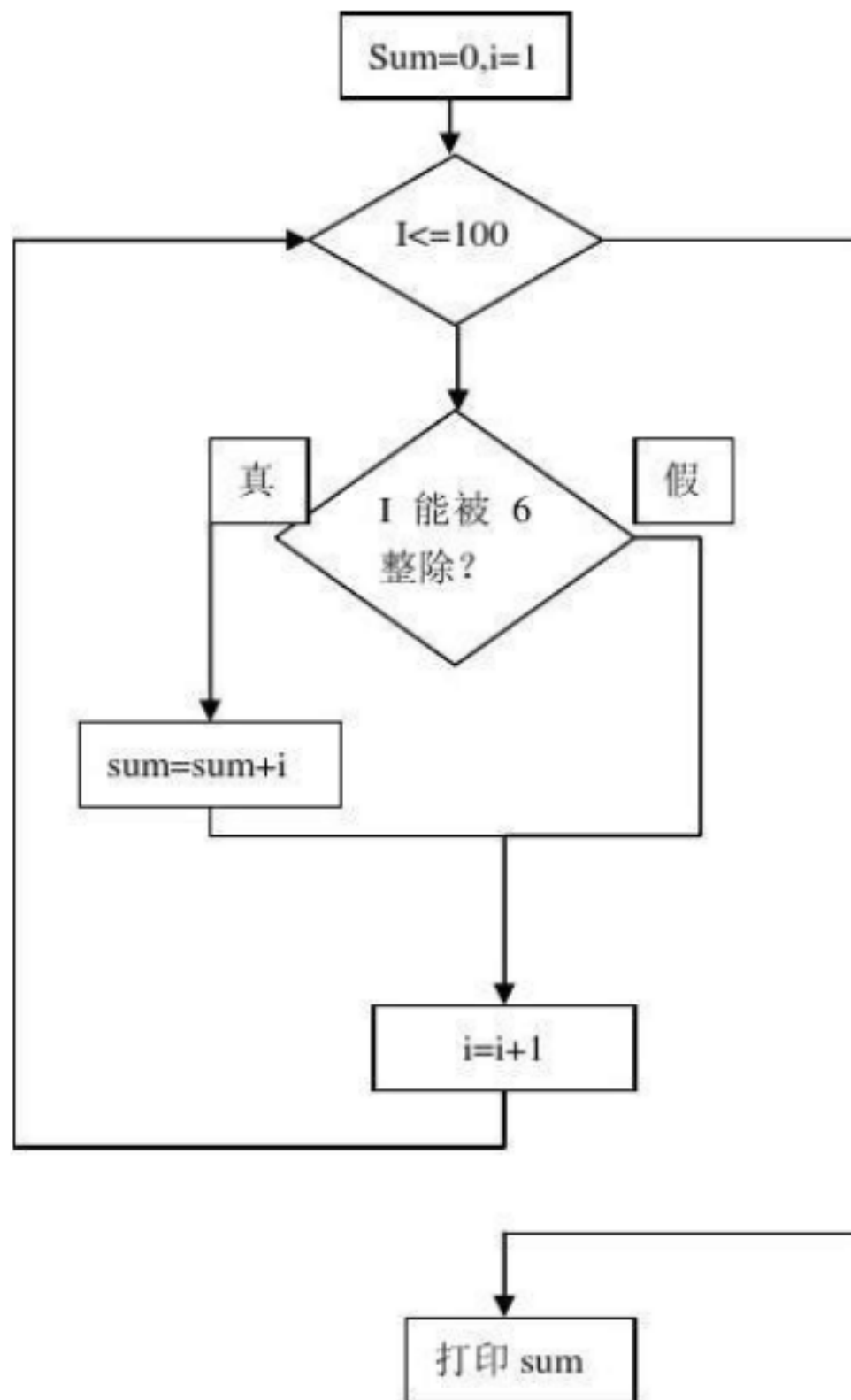
解答：合法的：total、_debug、Counter1、begin_。

不合法的：Large & Tall。

2、改写 1.4

求 1-100 中
数的和。

解答：



节中的流程图 1.2，
能被 6 整除的所有整

3、改写 1.4 节中的程序，求 1—100 中能被 6 整除的所有整数的和，并在编程环境中验证该程序的运行结果。

解答：

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

```
int main(void)
```

```
{
```

```

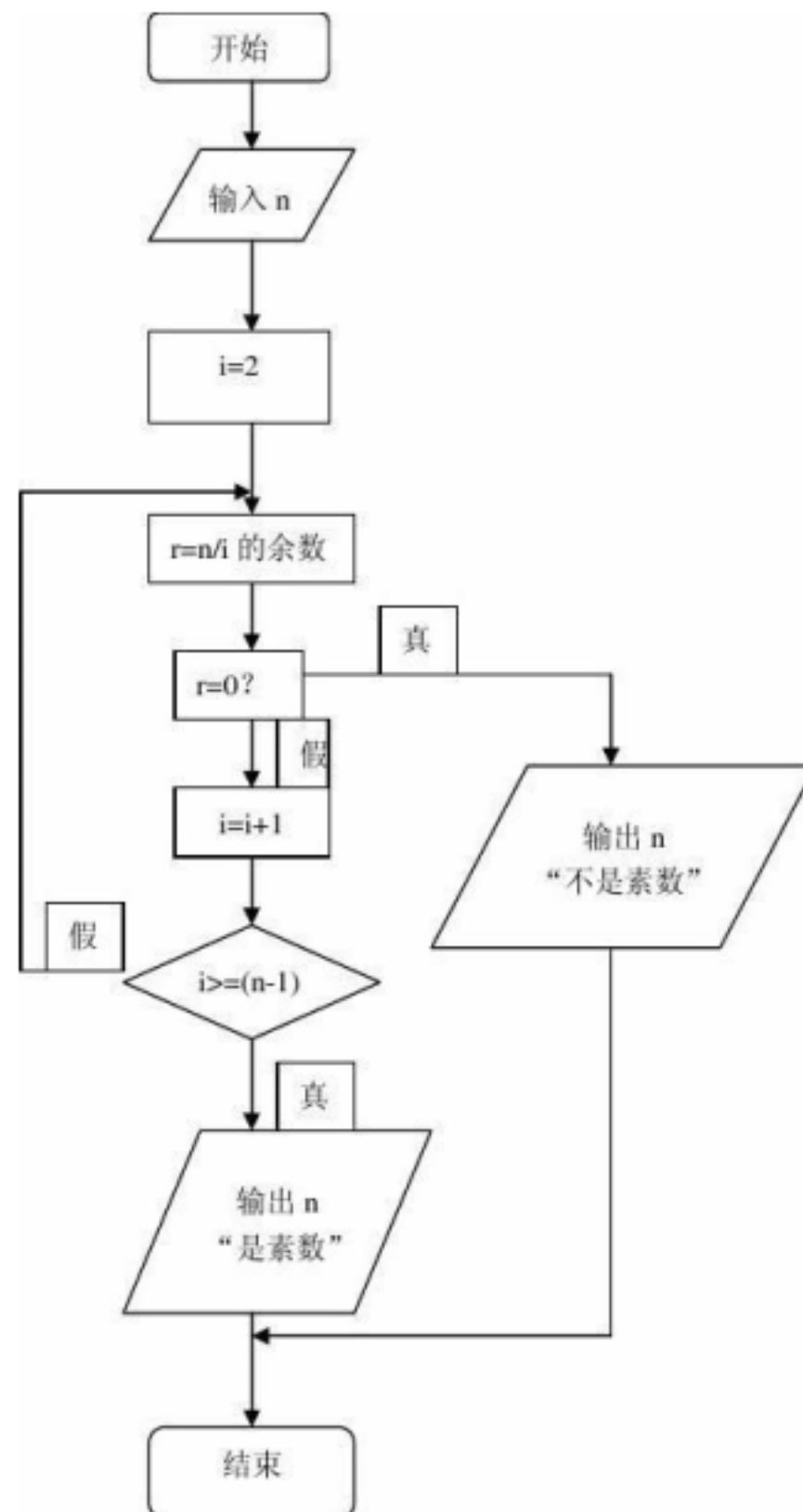
int i,sum=0;

for(i=1;i<=100;i++)
    if(i%6==0) sum=sum+i;
printf("sum=%d\n",sum);

return 0;
}

```

4、对于给定的整数
一个流程图判别 n
(只能被 1 和自己
析该流程图中哪些
是分支结构和循环



n ($n > 1$), 请设计
是否为一个素数
整除的整数), 并分
是顺序结构、哪些
结构。

第 2 章 C 语言编写程序

【练习 2-1】输出短句 (Programming in C is fun !): 在屏幕上显示一个短句
“ Programming in C is fun! ”。试编写相应程序。

解答：

```
#include <stdio.h>
int main(void)
{
    printf("Programming in C is fun!\n");

    return 0;
}
```

【练习 2-2】下列语句的运行结果是什么？与例 2-2 的运行结果有何不同？为什么？

```
printf("Programming is fun. And Programming in C is even more fun!\n");
```

解答：

运行结果： Programming is fun. And Programming in C is even more fun!

例 2-2 的运行结果分 2 行显示。原因在于，例 2-2 中在每条输出语句的结尾都采用了 ‘ \n ’ 换行。

【练习 2-3】输出倒三角图案：在屏幕上显示如下倒三角图案。 试编写相应程序。

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

解答：

```
#include<stdio.h>
int main(void)
{
    printf("* * * *\n * * * \n * * \n * \n");
    return 0;
}
```

【练习 2-4】温度转换：求华氏温度 150° F 对应的摄氏温度（计算公式同例 2-3 ）。试编写相应程序。

解答：

```
#include<stdio.h>
int main(void)
{
    double celsius,fahr;

    printf("Enter fahr:\n");
    scanf("%lf",&fahr);
```

```

    celsius=5.0*(fahr-32)/9;
    printf("celsius=%.2lf\n",celsius);

    return 0;
}

```

【练习 2-5】算术表达式 $5*(fahr-32)/9$ 能改写成 $5(fahr-32)/9$ 吗？为什么？如果将其改写为 $5/9*(fahr-32)$ ，会影响运算结果吗？

解答：

$5(fahr-32)/9$ 不是合法的 C 表达式，因为不能省略运算符 $*$ ； $5/9*(fahr-32)$ 的值为 0。

【练习 2-6】一个物体从 100 米的高空自由落下，编写程序，求它在前 3 秒内下落的垂直距离。设重力加速度为 10m/s^2 。

解答：

```

#include<stdio.h>
int main(void)
{
    double h,g=10,t=3;

    h=1/2*g*t*t;
    printf("h=%f\n",h);

    return 0;
}

```

【练习 2-7】输入提示和输入语句的顺序应该如何安排？例 2-5 中，`scanf("%d%d%lf", &money, &year,&rate)` 能改写为 `scanf("%d%lf%d", &money, &year, &rate)` 吗？为什么？能改写为 `scanf("%d%lf%d", &money, &rate, &year)` 吗？如果可以，其对应的输入数据是什么？

解答：

输入提示在前，输入语句在后。不能改写为 `scanf("%d%lf%d", &money,&year, &rate)`，因为 `%lf` 与整型变量 `money` 不匹配，`%d` 与浮点型变量 `rate` 不匹配。与 `scanf("%d%lf%d", &money,&rate, &year)` 对应的输入数据为：1000 0.025 3

【练习 2-8】计算摄氏温度：输入华氏温度，输出对应的摄氏温度，计算公式同例 2-3。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    int celsius, fahr;

    Printf("Input fahr: ");
    scanf("%d", &fahr);
    celsius = 5 * (fahr - 32) / 9;
    printf("celsius = %d\n", celsius);
}

```

```

        return 0;
    }

```

【练习 2-9】整数四则运算：输入 2 个正整数，计算并输出它们的和、差、积、商。试编写相应程序。

解答：

```

#include<stdio.h>
int main (void)
{
    double m,n;

    printf("Enter m:\n");
    scanf("%lf",&m);
    printf("Enter n(n!=0):\n");
    scanf("%lf",&n);
    printf("m+n=%.0lf\n",m+n);
    printf("m-n=%.0lf\n",m-n);
    printf("m*n=%.0lf\n",m*n);
    printf("m/n=%.2lf\n",m/n);

    return 0;
}

```

【练习 2-10】计算分段函数（判断 x 是否不为 0）：输入 x，计算并输出下列分段函数 f(x) 的值（保留 1 位小数）。试编写相应程序。

$$y=f(x)=\begin{cases} \frac{1}{x} & x \neq 0 \\ 0 & x=0 \end{cases}$$

解答：

```

#include<stdio.h>
int main(void)
{
    double x,y;

    printf("Enter x:\n");
    scanf("%lf",&x);
    if(x!=0)
        y=1/x;
    else
        y=0;
    printf("y=f(%lf)=%.1lf\n",x,y);
}

```

```

    return 0;
}

```

【练习 2-11】计算分段函数（判断 x 是否小于 0）：输入 x ，计算并输出下列分段函数 $f(x)$ 的值（保留 2 位小数）。可包含头文件 `math.h`，并调用 `sqrt()` 函数求平方根，调用 `pow()` 函数求幂。试编写相应程序。

$$y=f(x)=\begin{cases} 2+2x+\frac{1}{x} & x < 0 \\ \sqrt{x} & x \geq 0 \end{cases}$$

解答：

```

#include <stdio.h>
#include <math.h>
int main(void)
{
    double x, y;

    printf("    Input x:    ");
    scanf("%lf", &x);
    if(x >= 0)
        y = sqrt(x);
    else
        y = pow(x,5) + 2 * x + 1 / x;
    printf("f(%.2f) = %.2f\n", x, y);

    return 0;
}

```

【练习 2-12】输出华氏 - 摄氏温度转换表：输入两个整数 `lower` 和 `upper`，输出一张华氏 - 摄氏温度转换表，华氏温度的取值范围是 `[lower, upper]`，每次增加 2°F ，计算公式同例 2-6。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    int lower, upper;
    double celsius, fahr;

    printf("    Input lower and upper:    ");
    scanf("%d%d", &lower, &upper);
    if(lower<=upper){
        printf("    fahr    celsius\n    ");
        for(fahr = lower; fahr <= upper; fahr = fahr + 2){
            celsius = 5 * (fahr - 32.0)/9 ;

```

```

        printf("%3.0f %6.1f\n", fahr, celsius);
    }
    else
        printf("Invalid Value!\n");

    return 0;
}

```

【练习 2-13】求给定序列前 n 项和 ($1+1/2+1/3+ \dots$): 输入一个正整数 n , 计算序列 $1+1/2+1/3+ \dots$ 的前 n 项之和。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    int i, n;
    double sum;

    printf("Input n: ");
    scanf("%d", &n);
    sum = 0;
    for(i = 1; i <= n; i++)
        sum = sum + 1.0 / i;
    printf("sum = %.6f\n", sum);
}

return 0;
}

```

【练习 2-14】求给定序列前 n 项和 ($1+1/3+1/5+ \dots$): 输入一个正整数 n , 计算序列 $1+1/3+1/5+ \dots$ 的前 n 项之和。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    int i, n;
    double sum;

    printf("Input n: ");
    scanf("%d", &n);
    sum = 0;
    for(i = 1; i <= n; i++)
        sum = sum + 1.0/(2*i-1);
    printf("sum = %.6f\n", sum);

    return 0;
}

```

【练习 2-15】求给定序列前 n 项和 ($1-1/4+1/7-1/10 \dots$): 输入一个正整数 n , 计算序列 $1-1/4+1/7-1/10+1/13-1/16 \dots$ 的前 n 项之和。试编写相应程序。

解答：

```
#include <stdio.h>
int main(void)
{
    int flag, i, n, denominator;
    double item, sum;

    printf("    Input n:    ");
    scanf("%d", &n);
    sum = 0;
    flag = 1;
    denominator = 1;
    for(i = 1; i <= n; i++){
        item = flag * 1.0 / denominator;
        sum = sum + item;
        flag = -flag;
        denominator = denominator + 3;
    }
    printf("sum = %.3f\n", sum);

    return 0;
}
```

【练习 2-16】执行下列程序段后，sum 的值_____。

```
for (i = 1; i <= 10; i++){
    sum = 0;
    sum = sum + i;
}
```

解答：sum 的值是 10。

2.5 生成乘方表与阶乘表

【练习 2-17】生成 3 的乘方表：输入一个正整数 n , 生成一张 3 的乘方表，输出 3 的 0 次方到 3 的 n 次方的值。可包含头文件 `math.h`, 并调用幂函数计算 3 的乘方。试编写相应程序。

解答：

```
#include <stdio.h>
#include <math.h>
int main(void)
{
    int i, n;
    double mypow;

    printf("    Input n:    ");
```



```

scanf("%d", &n);
for(i = 0; i <= n; i++){
    mypow = pow(3, i);
    printf("pow(3,%d) = %.0f\n", i, mypow);
}

return 0;
}

```

【练习 2-18】求组合数：根据下列公式可以算出从 n 个不同元素中取出 m 个元素 ($m \leq n$) 的组合数。输入两个正整数 m 和 n ($m \leq n$)，计算并输出组合数。要求定义和调用函数 $\text{fact}(n)$ 计算 $n!$ ，函数类型是 double 。

$$C_n^m = \frac{n!}{m!(n-m)!}$$

解答：

```

#include <stdio.h>
double fact(int n);
int main(void)
{
    int m, n;
    double s;

    printf("Input m and n: ");
    scanf("%d%d", &m, &n);
    s = fact(n)/(fact(m) * fact(n-m));
    printf("result = %.0f\n", s);

    return 0;
}
double fact(int n)
{
    int i;
    double product;

    product = 1;
    for(i = 1; i <= n; i++)
        product = product * i;
    return product;
}

```

习题 2

1. 求整数均值：输入 4 个整数，计算并输出这些整数的和与平均值，其中平均值精确到小数点后 1 位。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    double num1, num2,num3,num4,;

    printf("Input num1, num2, num3 and num4:          ");
    scanf("%lf%lf%lf%lf", &num1, &num2,&num3,&num4);
    printf("sum=%.0lf\n", num1+ num2+ num3+num4);
    printf("average= %.1lf\n", (num1+num2+num3+num4)/4);

    return 0;
}

```

2. 阶梯电价：为了倡导居民节约用电，某省电力公司执行“阶梯电价”，安装一户一表的居民用户电价分两个“阶梯”：月用电量 50 千瓦时（含 50 千瓦时）以内的，电价为 0.53 元/千瓦时；超过 50 千瓦时，超出的用电量电价上调 0.05 元/千瓦时。输入用户的月用电量（千瓦时），计算并输出该用户应支付的电费（元）。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    double cost, e;
    printf("Input e:");
    scanf("%lf", &e);
    if(e <= 50)
        cost = 0.53 * e;
    else
        cost = 0.53 * 50 + (e - 50) * 0.58;
    printf("cost = %.2f\n", cost);

    return 0;
}

```

3. 序列求和：输入两个正整数 m 和 n ($0 < m \leq n$)，求 $\sum_{i=m}^n \left(i^2 + \frac{1}{i}\right)$ ，结果保留 6

位小数。试编写相应程序

解答：

```

#include <stdio.h>
int main(void)
{
    int i, m, n;
    double sum;

```

```

printf("Input m and n:"          ) ;
scanf("%d%d", &m, &n);
sum = 0;
for(i = m; i <= n; i++)
    sum = sum + i * i + 1.0 / i;
printf("sum = %.6f\n", sum);

return 0;
}

```

4 . 求交错序列前 n 项和：输入一个正整数 n，计算交错序列 $1-2/3+3/5-4/7+5/9-6/11+\dots$ 的前 n 项之和。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    int flag, i, n;
    double item, sum;

    printf("    Input n:    ");
    scanf("%d", &n);
    sum = 0;
    flag = 1;
    for(i = 1; i <= n; i++){
        item = flag * 1.0 * i / (2 * i - 1);
        sum = sum + item;
        flag = -flag;
    }
    printf("sum = %.3f\n", sum);

    return 0;
}

```

5. 平方根求和：输入一个正整数 n，计算 $\sqrt{2}+\sqrt{3}+\dots+\sqrt{n}$ ，可包含头文件 `mat`

`h.h`，并调用 `sqrt()` 函数求平方根。试编写相应程序。

解答：

```

#include <stdio.h>
#include <math.h>
int main(void)
{
    int i, n;
    double sum;

```

```

printf("Input n:");
scanf("%d", &n);
sum = 0;
for(i = 1; i <= n; i++){
    sum = sum + sqrt(ii);
}
printf("sum = %lf\n", sum);

return 0;
}

```

6. 求给定序列前 n 项和 ($1! + 2! + \dots$): 输入一个正整数 n , 求 $e = 1! + 2! + 3! + \dots + n!$ 的值。要求定义和调用 `fact (n)` 计算 $n!$, 函数类型是 `double`。试编写相应程序。

解答：

```

#include <stdio.h>
double fact(int n);
int main(void)
{
    int i, n;
    double sum;

    printf("Input n:");
    scanf("%d", &n);
    sum = 0;
    for(i = 0; i <= n; i++)
        sum = sum + fact(i);
    printf("sum = %.0f\n", sum);

    return 0;
}
double fact(int n)
{
    int i;
    double product;

    product=1;
    for(i = 1; i <= n; i++)
        product = product * i;
    return product;
}

```

第 3 章 分支结构

【练习 3-1】例 3-4 中使用 else-if 语句求解多分段函数，为了检查 else-if 语句的三个分支是否正确，已经设计了三组测试用例，请问还需要增加测试用例吗？为什么？如果要增加，请给出具体的测试用例并运行程序。

解答：

最好再增加两组测试用例，因为尚未对分段函数参数的边界值进行测试。可再给出 $x=0$ 和 $x=15$ 时的两种情况。

【练习 3-2】计算符号函数的值：输入一个整数 x ，计算并输出下列分段函数 $\text{sign}(x)$ 的值。

$$y=\text{sign}(x)=\begin{cases} -1 & x<0 \\ 0 & x=0 \\ 1 & x>0 \end{cases}$$

解答：

```
#include <stdio.h>
int main(void)
{
    int x, y;

    printf("Enter x: ");
    scanf("%d",&x);
    if(x>0) y=1;
    else if (x==0) y=0;
    else y=-1;
    printf("sign( %d ) = %d\n",x,y);

    return 0;
}
```

【练习 3-3】统计学生平均成绩与及格人数：输入一个正整数 n ，再输入 n 个学生的成绩，计算平均成绩，并统计所有及格学生的人数。试编写相应程序。

解答：

```
#include <stdio.h>
int main(void)
{
    int count, i, n;
    double grade, total;

    printf("Enter n: ");
    scanf("%d", &n);
    total = 0;
    count = 0;
```

```

        for(i = 1; i <= n; i++){
            printf("Enter grade #%d: ", i);
            scanf ("%lf", &grade);
            total = total + grade;
            if(grade >= 60)    count++;
        }
        printf("Grade average = %.2f\n", total / n);
        printf("Number of pass = %d\n", count);

        return 0;
}

```

【练习 3-4】统计字符：输入 10 个字符，统计其中英文字母、空格或回车、数字字符和其他字符的个数。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    int i;
    int blank, digit, letter, other;
    char c;

    blank = digit = letter = other = 0;
    for(i=1;i<=15;i++){
        c = getchar();
        if(c>='a' && c <= 'z' || c>='A' && c <= 'Z')
            letter++;
        else if(c>='0' && c <= '9')
            digit++;
        else if(c==' '||c=='\n')
            blank++;
        else
            other++;}
    printf("letter=%d,blank=%d,digit=%d,other=%d\n", letter, blank, digit, other);

    return 0;
}

```

【练习 3-5】输出闰年：输出 21 世纪中截至某个年份之前的所有闰年年份。判断闰年的条件是：能被 4 整除但不能被 100 整除，或者能被 400 整除。试编写相应程序。

解答：

```

# include <stdio.h>
int main(void)
{

```

```

int year;

printf("Enter year: ");
scanf("%d", &year);
    for(;year>=2000;year--){
        if( (year % 4 == 0 && year % 100 != 0 ) || year % 400 == 0)
            printf("It is a leap year\n");
    else
        printf("It is not a leap year\n ");}

return 0;
}

```

【练习 3-6】在例 3-8 程序中，如果把 switch 语句中所有的 break 都去掉，运行结果会改变吗？如果有变化，输出什么？为什么？

解答：

如果去掉所有的 break 语句，运行结果会改变，输出 price = 0.0，因为不管 choice 值与其中某个常量表达式相等，当去掉 break 语句时，其后的所有语句段都将运行，故每次都将执行到 price=0.0 语句为止。

【练习 3-7】成绩转换：输入一个百分制成绩，将其转换为五分制成绩。百分制成绩到五分制成绩的转换规则：大于或等于 90 分为 A，小于 90 分且大于或等于 80 分为 B，小于 80 分且大于或等于 70 分为 C，小于 70 分且大于或等于 60 分为 D，小于 60 分为 E。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    double grade;

    printf("Input grade: ");
    scanf(" %lf", &grade);
    if(grade>=90)    printf("A");
    else if(grade>=80) printf("B");
        else if(grade>=70) printf("C");
        else if(grade>=60) printf("D");
        else printf("E");

    return 0;
}

```

或

```

#include <stdio.h>
int main(void)
{
    double score;
    char grade;

```

```

printf("Input your score: ");
scanf("%lf",&score);
switch(int(score/10)){
    case 10:
        case 9:
            grade='A' ; break;
    case 8:
        grade='B'; break;
    case 7:
        grade='C'; break;
    case 6:
        grade='D'; break;
    default:
        grade='E'; break;
}
printf("Your grade is: %c",grade);

return 0;
}

```

【练习 3-8】查询水果的单价：有 4 种水果，苹果（ apples ） 梨（ pears ） 桔子（ oranges ） 和葡萄（ grapes ），单价分别是 3.00 元/ 公斤， 2.50 元/ 公斤， 4.10 元/ 公斤和 10.20 元/ 公斤。在屏幕上显示以下菜单（编号和选项），用户可以连续查询水果的单价，当查询次数超过 5 次时，自动退出查询；不到 5 次时，用户可以选择退出。当用户输入编号 1 ~ 4，显示相应水果的单价（保留两位小数）；输入 0，退出查询；输入其他编号，显示价格为 0。试编写相应程序。

```

[1] apples
[2] pears
[3] oranges
[4] grapes
[0] Exit

```

解答：

```

#include <stdio.h>
int main(void)
{
    int ri;
    int choice;
    float price;

    for(ri=1; ri<=5; ri++){
        printf("[1] apples\n");
        printf("[2] pears\n");
        printf("[3] oranges\n");
        printf("[4] grapes\n");
    }
}

```



```

        printf("[0] Exit\n");
        scanf("%d", &choice);
        if(choice==0)
            break;
        else {
            switch(choice){
                case 1: price=3.00; break;
                case 2: price=2.50; break;
                case 3: price=4.10; break;
                case 4: price=10.20; break;
                default: price=0;break;
            }

            printf("price=%0.2f\n", price);
        }

        return 0;
    }
}

```

【练习 3-9】 请读者重新编写例 3-4 的程序，要求使用嵌套的 if - else 语句，并上机运行。

解答：

```

#include <stdio.h>
int main(void)
{
    double x, y;

    printf("Enter x:");
    scanf("%lf", &x);
    if (x > 15){
        y = 2.5 * x - 10.5;
    }
    else {
        if(x<0)
            y=0;
        else
            y=4*x/3;
    }

    printf("f(%.2f)=%.2f\n", x, y);

    return 0;
}

```

【练习 3-10】 在例 3-12 中，改写 if 语句前， $y = x + 1$ ；和 $y = x + 2$ ； 两条语句的执行条件是什么？改写后呢？

解答：

改写前： $y = x + 1$ 的执行条件是 $x < 1$ ； $y = x + 2$ 的执行条件是 $1 \leq x < 2$ 。

改写后：y=x+1 的执行条件是 x<1; y=x+2 的执行条件是 2<=x。

习题 3

一．选择题

1．能正确表示逻辑关系：“a ≥ 10 或 a ≤ 0”的 C 语言表达式是 D 。

- A. a >= 10 or a <= 0 B. a >= 0 | a <= 10
C. a >= 10 && a <= 0 D. a >= 10 || a <= 0

2．在嵌套使用 if 语句时，C 语言规定 else 总是 C 。

- A. 和之前与其具有相同缩进位置的 if 配对
B. 和之前与其最近的 if 配对
C. 和之前与其最近的且不带 else 的 if 配对
D. 和之前的第一个 if 配对

3．下列叙述中正确的是 D 。

- A. break 语句只能用于 switch 语句
B. 在 switch 语句中必须使用 default
C. break 语句必须与 switch 语句中的 case 配对使用
D. 在 switch 语句中，不一定使用 break 语句

4．有一函数 $y = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}$ ，以下程序段中错误的是 C 。

- A. if(x > 0) y = 1;
 else if(x == 0) y = 0;
 else y = -1;
C. y = 0;
 if(x >= 0);
 if(x > 0) y = 1;
 else y = -1;
B. y=0;
 if(x>0) y=1;
 else if(x<0) y=-1;
D. if(x>=0)
 if(x>0) y=1;
 else y=0;
 else y=-1;

5．下列程序段的输出结果是 C 。

```
int main(void)
{
    int a=2,b=-1,c=2;
    if(a<b)
        if(b<0)
            c=0;
        else c++;
    printf("%d\n",c);
    return 0;
}
```

- A. 0 B. 1 C. 2 D. 3

6．下列程序段的输出结果是 A 。

```
int main(void)
{
```

```

int x = 1,a = 0,b = 0;
switch(x)
{
    case 0:    b++;
    case 1:    a++    ;
    case 2:    a++;b++;
}
printf("a=%d,b=%d\n",a,b);
return 0;
}

```

- A . a=2,b=1 B . a=1,b=1
 C . a=1,b=0 D . a=2,b=2

7 . 在执行以下程序时 , 为使输出结果为 : t=4 , 则给 a 和 b 输入的值应满足的条件是 C 。

```

int main(void)
{
    int  a,b,s,t;
    scanf("%d,%d",&a,&b);
    s = 1; t = 1;
    if(a > 0) s = s + 1;
    if(a > b) t = s + t;
    else if(a == b) t = 5;
    else t = 2 * s;
    printf("t=%d\n",t);
    return 0;
}

```

- A . a>b B . a<b<0
 C . 0<a<b D . 0>a>b

二、填空题

- 1 . 表示条件 : 10<x<100 或者 x<0 的 C 语言表达式是 (x>10&&x<100)|| (x<0) 。
- 2 . 已知 a=7.5 , b=2 , c=3.6 , 则表达式 a>b&& c>a||a<b&&!c>b 的值为 0 。
- 3 . 若从键盘输入 58 , 则以下程序段的输出结果是 585858 。

```

int main(void)
{
    int  a;
    scanf("%d",&a);
    if(a > 50)  printf("%d",a);
    if(a > 40)  printf("%d",a);
    if(a > 30)  printf("%d",a);
    return 0;
}

```

- 4 . 下列程序运行的输出结果是 9 。

```

int main(void)
{
    char c = ' b ' ;
    int k = 4;
    switch(c)
    {
        case 'a': k = k + 1;break;
        case 'b': k = k + 2;
        case 'c': k = k + 3;
    }
    printf("%d\n",k);
    return 0;
}

```

三 . 程序设计题

1. 比较大小：输入 3 个整数，按从小到大的顺序输出。试编写相应程序。

解答：

```

#include<stdio.h>
int main(void)
{
    int a,b,c;
    char op='<';

    printf("Enter a:");
    scanf("%d",&a);
    printf("Enter b:");
    scanf("%d",&b);
    printf("Enter c:");
    scanf("%d",&c);
    if(a>b)
    if(b>c)printf("%d%c%d%c%d",c,op,b,op,a);
    else{
        if(a>c)printf("%d%c%d%c%d",b,op,c,op,a);
        else printf("%d%c%d%c%d",b,op,a,op,c);
    }
    else{
        if(a>c)printf("%d%c%d%c%d",c,op,a,op,b);
        else{
            if(b<c)printf("%d%c%d%c%d",a,op,b,op,c);
            else printf("%d%c%d%c%d",a,op,c,op,b);
        }
    }

    return 0;
}

```

2. 高速公路超速处罚：按照规定，在高速公路上行驶的机动车，超过本车道限速的 10%则处 200 元罚款；若超出 50%，就要吊销驾驶证。请编写程序根据车速和限速判别对该机动车的处理。

解答：

```
#include<stdio.h>
int main(void)
{
    double speed,rate_limiting;

    printf("Enter speed:");
    scanf("%lf",&speed);
    printf("Enter rate_limiting:");
    scanf("%lf",&rate_limiting);
    if(rate_limiting*1.5>speed>rate_limiting*1.1)
        printf("    罚款 200 元");
    else if(speed>rate_limiting*1.5)
        printf("    罚款 200 元，并吊销驾驶证 ");
    else;

    return 0;
}
```

3. 出租车计价：某城市普通出租车收费标准如下：起步里程 3 公里，起步费 10 元；超起步里程后 10 公里内，每公里 2 元，超过 10 公里以上的部分加收 50% 的空驶补贴费，即每公里 3 元；营运过程中，因路阻及乘客要求临时停车的，按每 5 分钟 2 元计收（不足 5 分钟则不收费）。运价计费尾数四舍五入，保留到元。编写程序，输入行驶里程（公里）与等待时间（分钟），计算并输出乘客应支付的车费（元）。

解答：

```
#include <stdio.h>
int main(void)
{
    int mile,time,cost;

    printf("Input mile and time:");
    scanf("%d%d",&mile,&time);
    cost=0;
    mile=mile+time/5;
    if(mile<=3)
        cost=10;
    else if(mile<=13)
        cost=10+(mile-3)*2;
    else
        cost=10+10*2+(mile-13)*3;
    printf("cost=%d\n",cost);
}
```

```
return 0;
}
```

4. 统计学生成绩：输入一个正整数 n ，再输入 n 个学生的成绩，统计五分制成绩的分布。百分制成绩到五分制成绩的转换规则：大于或等于 90 分为 A，小于 90 分且大于或等于 80 分为 B，小于 80 分且大于或等于 70 分为 C，小于 70 分且大于或等于 60 分为 D，小于 60 分为 E。试编写相应程序。

解答：

```
#include <stdio.h>
int main(void)
{
    int mark, n, i, sum;
    double average;
    int counta, countb, countc, countd, counte;

    printf("Enter n: ");
    scanf("%d",&n);
    counta= countb = countc = countd = counte =sum=0;
    for(i=1; i<=n; i++){
        scanf("%d",&mark);
        sum=sum+mark;
        if(mark>=90 ) counta++;
        else if (mark>=80) countb++;
        else if(mark>=70) countc++;
        else if(mark>=60) countd++;
        else counte++;
    }
    average=sum*1.0/n;
    printf("average=%.1f\n",average);
    printf("A:    %d, B: %d,C: %d,D: %d,E: %d\n",counta,  countb, countc, countd,
    counte);

    return 0;
}
```

5. 三角形判断：输入平面上任意三个点的坐标 (x_1, y_1) (x_2, y_2) (x_3, y_3)，检验他们能否构成三角形。如果这 3 个点能构成一个三角形，输出周长和面积（保留 2 位小数）；否则，输出“Impossible”。试编写相应程序。

提示：在一个三角形中，任意两边之和大于第三边。三角形面积计算公式如下：

$$area = \sqrt{s(s-a)(s-b)(s-c)}, \text{ 其中 } s = (a+b+c)/2.$$

解答：

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

```
int main(void)
{
    double x1,y1,x2,y2,x3,y3;
    double area,l,a,b,c,s;

    printf("Enter x1 and y1:");
    scanf("%lf%lf",&x1,&y1);
    printf("Enter x2 and y2:");
    scanf("%lf%lf",&x2,&y2);
    printf("Enter x3 and y3:");
    scanf("%lf%lf",&x3,&y3);
    a=sqrt(pow(fabs(x2-x1),2)+pow(fabs(y2-y1),2));
    b=sqrt(pow(fabs(x3-x2),2)+pow(fabs(y3-y2),2));
    c=sqrt(pow(fabs(x1-x3),2)+pow(fabs(y1-y3),2));
    if(a+b>c&&a+c>b&&b+c>a){
        l=a+b+c;
        s=(a+b+c)/2;
        area=sqrt(s*(s-a)*(s-b)*(s-c));
        printf("l=%.2lf\n",l);
        printf("area=%.2lf\n",area);
    }
    else
        printf("Impossible");

    return 0;
}
```

第 4 章 循环结构

【练习 4-1】在例 4-1 程序中，如果对 item 赋初值 0，运行结果是什么？为什么？如果将精度改为 10^{-3} ，运行结果有变化吗？为什么？

解答：

如果对 item 赋初值 0，则程序运行结果是 $\pi=0$ ，因为 item 为 0 时不满足 while 循环的执行条件，即不执行循环，故 π 值为 0。如果将精度改为 10^{-3} ，运行结果会有变化，因为精度改变意味着 while 循环的条件改变，精度变大使得 while 循环次数减少，必然影响到 π 的值。

【练习 4-2】运行例 4-2 程序时，如果将最后一个输入数据改为 -2，运行结果有变化吗？如果第一个输入数据是 -1，运行结果是什么？为什么？

解答：

如果将最后一个输入数据改为 -2，运行结果没有变化，因为最后一个负数是一结束标志，不进行统计，故任意一个负数都可以。如果第一个输入数据是 -1，运行结果是：Grade average is 0，因为第一个输入就是 -1，则 while 循环条件不成立，不执行循环体。

【练习 4-3】序列求和 ($1-1/4+1/7-1/10+1/13-1/16+\dots$): 输入一个正实数 eps，计算序列 $1-1/4+1/7-1/10+1/13-1/16+\dots$ 的值，精确到最后一项的绝对值小于 eps (保留 6 位小数)。试编写相应程序。

解答：

```
#include <stdio.h>
#include <math.h>
int main(void)
{
    int denominator, flag;
    double item, s, eps;

    printf( " Input eps:  " );
    scanf( " %lf ", &eps);
    flag = 1;
    denominator = 1 ;
    item = 1.0;
    s = 0;
    while(fabs (item) >= eps) {
        item = flag * 1.0 / denominator;
        s = s + item;
        flag = -flag;
        denominator = denominator +3;
    }
    printf ( " s = %f  n" , s);
```



```

    return 0;
}

```

【练习 4-4】如果将例 4-3 程序中的 do-while 语句改为下列 while 语句，会影响程序的功能吗？为什么？再增加一条什么语句，就可以实现同样的功能？

```

while(number != 0){
    number = number / 10;
    count ++;
}

```

解答：

会有影响，因为当输入数据 number 为 0 时，上述 while 循环将不执行，count 值仍为 0，故输出为 0，不符合题目要求。可增加一条 if 语句来解决上面的问题，在 while 循环前加上语句“if (number==0) count=1; ”。

【练习 4-5】例 4-4 程序中的第 9 ~ 15 行可以用下列 for 语句替代吗？为什么？

```

for( i = 2; i <= m/2; i++)
    if( m%i == 0 ) printf("No!\n");
    else printf("%d is prime number!\n",m);

```

解答：

不能代替，因为只用一个数来除不足以判断该数是否是素数。

【练习 4-6】猜数字游戏：先输入 2 个不超过 100 的正整数，分别是被猜数 mynumber 和允许猜测的最大次数 n，再输入你所猜的数 yournumber，与被猜数 mynumber 进行比较，若相等，显示猜中，；若不等，显示与被猜数的大小关系，最多允许猜 n 次。如果 1 次就猜出该数，提示“Bingo!”；如果 3 次以内猜到该数，则提示“Lucky You!”；如果超过 3 次但不超过 n 次猜到该数，则提示“Good Guess!”；如果超过 n 次都没有猜到，则提示“Game Over”；如果在到达 n 次之前，用户输入了一个负数，也输出“Game Over”，并结束程序。试编写相应程序。

解答：

```

#include<stdio.h>
int main(void)
{
    int mynumber,n,yournumber,i;

    printf("Input mynumber:");
    scanf("%d",&mynumber);
    printf("Input n:");
    scanf("%d",&n);
    for(i=1;i<=n;i++){
        printf("Input yournumber:");
        scanf("%d",&yournumber);
        if(i==1&&yournumber==mynumber){
            printf("Bingo!");break; }
        if((i==2||i==3)&&yournumber==mynumber){
            printf("Lucky you!");break;}
        if((i>3&&i<=n)&&yournumber==mynumber){

```

```

        printf("Good Guess!");break;}
    if((i==n&&mynumber!=yournumber)||mynumber<0||yournumber<0){
        printf("Game Over!");break;}
}

```

```

    return 0;
}

```

【练习 4-7】求 e 的值：输入 1 个正整数 n，计算下式求出 e 的值（保留 2 位小数），要求使用嵌套循环。试编写相应程序。

$$e = 1 + 1/1! + 1/2! + 1/3! + \dots + 1/n!$$

解答：

```

#include<stdio.h>
int main(void)
{
    int i, j, n;
    double e, t;

    printf("Input n: ");
    scanf("%d", &n);
    e = 1;
    for (i = 1; i <= n; i++){
        t=1;
        for(j=1;j<=i;j++)
            t = t * j;
        e = e + 1.0/t;
    }
    printf("e = %0.2f\n", e);

    return 0;
}

```

【练习 4-8】运行例 4-8 的源程序 1 时，如果先输入 0，即输入数据个数 n=0，表示不再输入任何成绩，运行结果是什么？如何修改程序以应对这种情况？

解答：

当输入数据个数 n = 0 时，由于在 for 循环外首先得输入一个成绩，故程序仍将等待输入第一个人的成绩，这与输入个数 n = 0 矛盾，可修改如下：

```

#include <stdio.h>
int main(void)
{
    int i, mark, max, n;

    printf("Enter n: ");
    scanf ("%d", &n);
    if (n>0) {
        printf("Enter %d marks: ", n);

```

```

scanf ("%d", &mark);          /*      读入第一个成绩      */
max = mark;                    /*      假设第一个成绩是最高分      */
for (i = 1; i < n; i++) {
    scanf ("%d", &mark);
    if (max < mark)
        max = mark;
}
printf("Max = %d\n", max);
}

return 0;
}

```

【练习 4-9】运行例 4-8 的源程序 2 时，如果输入的第一个数就是负数，表示不再输入任何成绩，运行结果是什么？如何修改程序以应对这种情况？

解答：

当输入的第一个数就是负数时，将不执行 while 循环，程序运行结果就是输出所输入的负数。

【练习 4-10】找出最小的值：输入一个正整数 n，再输入 n 个整数，输出最小值。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    int i, min, n, x;

    printf("input n: ");
    scanf("%d", &n);
    printf("input numbers: ");
    scanf("%d", &x);
    min = x;
    for (i = 1; i < n; i++){
        scanf("%d", &x);
        if (min > x)
            min = x;
    }
    printf("min = %d\n", min);

    return 0;
}

```

【练习 4-11】统计素数并求和：输入 2 个正整数 m 和 n (1 ≤ m ≤ n ≤ 500)，统计并输出 m 和 n 之间素数的个数以及这些素数的和。素数就是只能被 1 和自身整除的正整数，1 不是素数，2 是素数。试编写相应程序。

解答：

```

#include<stdio.h>

```

```

#include<math.h>
int main(void)
{
    int i,m,n,j,count;
    double sum;

    printf("Enter m:");
    scanf("%d",&m);
    printf("Enter n:");
    scanf("%d",&n);
    count=0;
    sum=0;
    for(;m<=n;m++){
        j=sqrt(m);
        for(i=2;i<=j;i++)
            if(m%i==0)
                break;
        if(i>j&&m!=1){
            sum=sum+m;
            count++;}
    }
    printf("count=%d\nsum=%.0lf",count,sum);

    return 0;
}

```

习题 4

一．选择题

1．C 语言中 while 和 do-while 循环的主要区别是 A 。

- A．do-while 的循环体至少无条件执行一次
- B．while 的循环控制条件比 do-while 的循环控制条件严格
- C．do-while 允许从外部转到循环体内
- D．do-while 的循环体不能是复合语句

2．下列叙述中正确的是 C 。

- A．break 语句只能用于 switch 语句体中
- B．continue 语句的作用是使程序的执行流程跳出包含它的所有循环
- C．break 语句只能用在循环体内和 switch 语句体内
- D．在循环体内使用 break 语句和 continue 语句的作用相同

3．下列叙述中正确的是 D 。

- A . do-while 语句构成的循环不能用其它语句构成的循环来代替。
B . do-while 语句构成的循环只能用 break 语句退出。
C . 用 do-while 语句构成的循环，在 while 后的表达式为非零时结束循环。
D . 用 do-while 语句构成的循环，在 while 后的表达式为零时结束循环。

4 . 执行 `x=-1; do { x=x*x; } while (!x);` 循环时，下列说法正确的是
A 。

- A . 循环体将执行一次
C . 循环体将执行无限次
B . 循环体将执行两次
D . 系统将提示有语法错误

5 . 假设变量 s、a、b、c 均已定义为整型变量，且 a、c 均已赋值（c 大于 0），
则与以下
程序段

```
s=a;
for(b=1;b<=c;b++)
    s=s+1;
```

功能等价的赋值语句是 B 。

- A . `s=a+b;` B . `s=a+c;` C . `s=s+c;` D . `s=b+c;`

6 . 以下程序段的输出结果是 B 。

```
int main(void)
{
    int num=0,s=0;
    while(num<=2){
        num++;
        s+=num;
    }
    printf("%d\n",s);
    return 0;
}
```

- A . 10 B . 6 C . 3 D . 1

7 . 运行以下程序后，如果从键盘上输入 65 14< 回车>，则输出结果为 C 。

```
int main(void)
{
    int m, n;
    printf("Enter m,n;");
    scanf("%d%d", &m,&n);
    while (m!=n)
    { while (m>n) m=m-n;
```

```

        while (n>m) n=n-m;
        }
    printf("m=%d\n",m);
    return 0;
}

```

A . m=3 B . m=2 C . m=1 D.m=0

8 . 下列程序段的输出结果是 A 。

```

int main(viod)
{
    int i;
    for(int i=1;i<6;i++){
        if(i%2!=0){
            printf("#");
            continue;
        }
        printf("*");
    }
    printf("\n");
    return 0;
}

```

A . #*#*# B . ##### C . ***** D . *#*#*

二、填空题

1 . 若 $s = 0$, 执行循环语句 "for(i=1; i<=10; i=i+3) s=s+i; " 后 , i 的值为 13 。

2 . 下列程序段的输出结果是 ** 。

```

int x=2;
do {
    printf( " * " );
    x--;
} while (!x==0);

```

3 . 下列程序段的功能是计算 : $s=1+12+123+1234+12345$ 则划线处应填 $t*10$ 。

```

int t=0,s=0,i;
for(i=1;i<=5;i++) {
    t=i+_____ ;
    s=s+t;
}

```

```

printf("s=%d\n",s);

```

4 . 以下程序的输出结果是 12 。

```

int main(void)
{
    int num=0;
    while(num<6) {
        num++;
    }
}

```

```

        if(num==3) break;
        printf("%d,",num); }
    return 0;
}

```

5 . 阅读下列程序并回答问题。

```

int main(void)
{   int i, j, k=0, m=0;
    for(i=0; i<2; i++)
    {
        ; /* 第 5 行*/
        for(j=0; j<3; j++)
            k++;
            m++; /* 第 9 行*/
    }
    printf("k=%d, m=%d\n", k, m);
}

```

- (1) 程序的输出是 k=6,m=2 。
- (2) 将第 8 行改为 " m=m+k; " , 程序的输出是 k=6,m=9 。
- (3) 将第 5 行改为 "k=0;" , 将第 8 行改为 " m=m+k; " , 程序的输出是 k=3,m=6 。

6. 下列程序运行时输入 62pay ! , 则结果是 62ufd! 。

```

int main(void)
{   char c;
    int i;
    for(i=1;i<=6;ii++){
        c=getchar( );
        if (c>='a'&&c<= ' u ' ) c=c+5;
        else if (c>='v'&&c<='z') c='a'+c-'v';
        putchar(c);
    }
    return 0;
}

```

三 . 程序设计题

1 . 求奇数和：输入一批正整数 （以零或负数为结束标志） , 求其中的奇数和。试编写相应程序。

解答：

```

#include <stdio.h>
int main(void)
{
    int x, sum;

    printf("Input integers: ");
    scanf("%d", &x);
    sum = 0;

```

```

while(x>0){
    if(x%2 !=0)
        sum = sum + x;
    scanf("%d", &x);
}
printf("The sum of the odd numbers is %d\n", sum);

return 0;
}

```

2. 展开式求和：输入 1 个实数 x ，计算并输出下式的值，直到最后一项的绝对值小于 0.00001，计算结果保留 2 位小数。要求定义和调用函数 $\text{fact}(n)$ 计算 n 的阶乘，可以调用 $\text{pow}()$ 函数求幂。试编写相应程序。

$$s = x + x^2/2! + x^3/3! + x^4/4! + \dots$$

解答：

```

#include<stdio.h>
#include<math.h>
double fact(int n);
int main(void)
{
    int i;
    double x,s,item;

    printf("Input x:");
    scanf("%lf",&x);
    s=0;
    item=1.0;
    i=1;
    while(fabs(item)>=1e-5){
        item=pow(x,i)/fact(i);
        s=s+item;
        i++;
    }
    printf("s=%.2lf",s);

    return 0;
}
double fact(int n)
{
    int i;
    double result;

    for(result=1,i=1;i<=n;i++)
        result=result*i;
}

```



```
    return result;
}
```

3. 求序列和：输入一个正整数 n ，输出 $2/1+3/2+5/3+8/5 + \dots$ 的前 n 项之和，保留 2 位小数。该序列从第 2 项起，每一项的分子是前一项分子与分母的和，分母是前一项的分子。试编写相应程序。

解答：

```
#include<stdio.h>
int main(void)
{
    int n,i,s1=2,s2=3,s,p1=1,p2=2,p;
    double sum,item;

    printf("Input n:");
    scanf("%d",&n);
    if(n>=3){
        for(i=1;i<=n-2;i++){
            s=s1+s2;
            p=p1+p2;
            item=1.0*s/p;
            sum+=item;
            s1=s2;s2=s;
            p1=p2;p2=p;
        }
    }
    else if(n==1)sum=2;
    else if(n==2)sum=3.5;
    printf("sum=%.2lf\n",sum);

    return 0;
}
```

4. 求序列和：输入 2 个正整数 a 和 n ，求 $a+aa+aaa+aa..a$ (n 个 a) 之和。例如，输入 2 和 3，输出 246(2+22+222)。试编写相应程序。

解答：

```
#include<stdio.h>
int main(void)
{
    int a,n,i;
    double sum,temp;

    printf("Input a,n:");
    scanf("%d%d",&a,&n);
    sum=temp=0;
    for(i=1;i<=n;i++){
        temp=temp*10+a;
```

```

        sum=sum+temp;
    }
    printf("sum=%.0lf",sum);

    return 0;
}

```

5. 换硬币。将一笔零钱（大于 8 分，小于 1 元，精确到分）换成 5 分、2 分和 1 分的硬币，每种硬币至少有一枚。输入金额，问有哪几种换法？针对每一种换法，输出各种面额硬币的数量和硬币的总数量。试编写相应程序。

解答：

```

#include<stdio.h>
int main(void)
{
    int money,money1,money2,sum;

    printf("Input money:");
    scanf("%d",&money);
    sum=0;
    for(money1=1;money1<=8;money1++)
        for(money2=1;money2<=3;money2++)
            if(money1+money2*2+5==money) {
                sum=money1+money2+1;
                if(money2>money1)
                    printf("money5=1,money1=%d,money2=%d\nsum=%d\n",m
oney1,money2,sum);
                else
                    printf("money5=1,money2=%d,money1=%d\nsum=%d\n",m
oney2,money1,sum);
            }
    return 0;
}

```

6. 输出水仙花数。输入一个正整数 n ($3 \leq n \leq 7$) 输出所有 n 位水仙花数。水仙花数是指一个 n 位正整数，它各位数字的 n 次幂之和等于它本身。例如 153 的各位数字的立方和是： $1^3+5^3+3^3=153$ 。试编写相应程序。

解答：

```

#include<stdio.h>
#include<math.h>
int main(void)
{
    int n,i,t,m,sum,j,k;

    printf("Input n:");
    scanf("%d",&n);
    for(j=0,i=1;i<=n;i++)

```

```

        j=j*10+9;
    for(k=1,i=1;i<n;i++)
        k=k*10;
    for(;k<=j;k++){
        t=k;
        sum=0;
        while(k!=0){
            m=k;
            k=k%10;
            sum=sum+pow(k,n);
            k=m/10;
        }
        k=t;
        if(sum==t)
            printf("%d\n",t);
    }

    return 0;
}

```

7 . 求最大公约数和最小公倍数。输入两个正整数 m 和 n ($m \leq 1000, n \leq 1000$), 求其最大公约数和最小公倍数。试编写相应程序。

解答：

```

#include<stdio.h>
int main(void)
{
    int m,n,j,k;
    do{
        printf("Input m:");
        scanf("%d",&m);
        printf("Input n:");
        scanf("%d",&n);
    }while(m<0||n<0);
    j=m;
    while(j%n!=0)
        j=j+m;
    k=(m*n)/j;
    printf("    最小公倍数是 %d,最大公约数是 %d\n",j,k);
    return 0;
}

```

8 . 高空坠球。皮球从 height (米) 高度自由落下，触地后反弹到原高度的一半，再落下，再反弹.....如此反复。问皮球在第 n 次落地时，在空中一共经过多少距离？第 n 次反弹的高度是多少？输出保留 1 位小数。试编写相应程序。

解答：

```

#include <stdio.h>

```

```

int main(void)
{
    double distance, height;
    int i, n;

    printf("Input height:");
    scanf("%lf",&height);
    printf("Input n:");
    scanf("%d",&n);
    if(n == 0)
        distance = 0;
    else
    {
        for(int i = 1; i <= n; i++){
            height = height/2;
            distance += 3*height;
        }
    }
    printf("    第%d次落地时，共经过距离为  %0.1lf.\n", n, distance);
    printf("    第%d次落地后的反弹高度为  %0.1lf.\n", n, height);

    return 0;
}

```

9. 打菱形星号 “ * ” 图案。输入一个正整数 n (n 为奇数) , 打印一个高度为 n 的 “ * ” 菱形图案。例如，当 n 为 7 时，打印出以下图案。试编写相应程序。

```

      *
    * * *
  * * * * *
* * * * * * *
  * * * * *
    * * *
      *

```

解答：

```

#include<stdio.h>
int main(void)
{
    int i,j,n;

    printf("Input n:");
    scanf("%d",&n);
    for(i=1;i<=n;i=i+2){
        for(j=1;j<=n-i;j++)
            printf(" ");

```

```

        for(j=1;j<=i;j++)
            printf("* ");
        putchar('\n');
    }
    for(i=1;i<=n-2;i=i+2){
        for(j=1;j<=i+1;j++)
            printf(" ");
        for(j=1;j<=n-i-1;j++)
            printf("* ");
        putchar('\n');
    }

    return 0;
}
或
#include<stdio.h>
#include<math.h>
int main(void)
{
    int i,j,n;

    printf("Input n:");
    scanf("%d",&n);
    for(i=1;i<=n;i++){
        for(j=1;j<=fabs(n+1-2*i)/2;j++)
            printf(" ");
        for(j=1;j<=n-fabs(n+1-2*i);j++)
            printf("* ");
        putchar('\n');
    }

    return 0;
}

```

10．猴子吃桃问题。猴子第一天摘下若干个桃子，当即吃了一半，还不过瘾，又多吃了一个；第二天早上将剩下的桃子吃掉一半，又多吃了一个。以后每天早上都吃了前一天剩下的一半加一个。到第 n 天早上想再吃时，见只剩下一个桃子了。问：第一天共摘了多少桃子？试编写相应程序。（提示：采取逆向思维的方法，从后往前推断。）

解答：

```

#include <stdio.h>
int main(void)
{
    int i, peach;

```

```

        peach = 1;
    for(i = 1; i < n; i++){
        peach = 2 * peach + 1;
    }
    printf("%d\n", peach);

    return 0;
}

```

11. 兔子繁衍问题。一对兔子，从出生后第 3 个月起每个月都生一对兔子。小兔子长到第 3 个月后每个月又生一对兔子。假如兔子都不死，请问第 1 个月出生的一对兔子，至少需要繁衍到第几个月时兔子总数才可以达到 n 对？输入一个不超过 10000 的正整数 n，输出兔子总数达到 n 最少需要的月数。试编写相应程序。

解答：

```

#include<stdio.h>
int main(void)
{
    int n,mouth,x1=1,x2=1,x;

    printf("Input n:");
    scanf("%d",&n);
    mouth=1;
    x=0;
    while(x<=n){
        x=x1+x2;
        x1=x2;
        x2=x;
        mouth++;
    }
    printf("mouths=%d",mouth);

    return 0;
}

```

第 5 章 函数

【练习 5-1】使用函数求 1 到 n 之和：输入一个正整数 n ，输出 $1 \sim n$ 之和。要求自定义和调用函数 $\text{sum}(n)$ 求 $1 \sim n$ 之和。若要计算 $m \sim n$ ($m < n$) 之和，又如何定义函数？试编写相应程序。

解答：

```
#include <stdio.h>
int sum(int n)    ;
int main(void)
{
    int n;
    int sum;

    printf("Enter n      : ")
    scanf("%d", &n);
    printf("sum = %d\n", sum(n));

    return 0;
}
int sum(int n)
{
    int result, i;
    result = 0;
    for(i = 1; i <= n; i++)
        result = result + i;

    return result;
}
```

若要计算 $m \sim n$ ($m < n$) 之和，则需要先在 $\text{main}()$ 中定义 2 个变量 m 和 n ：

```
scanf("%d%d", &m, &n);
printf("sum = %d\n", sum(m, n));
同时在函数定义时需设置 2 个形参：
int sum(int m, int n)
{
    int result, i;

    result = 0;
    for(i = m; i <= n; i++)
        result = result + i;

    return result;
}
```

【练习 5-2】使用函数找最大值： 输入 2 个数，输出其中较大的数。要求定义和调用函数 max(a, b) 找出并返回 a、b 中较大的数。试编写相应程序。

解答：

```
#include <stdio.h>
double max(double a, double b);
int main(void)
{
    int a, b;

    printf("    Input a and b:    ");
    scanf("%lf%lf",&a, &b);
    printf("max= %lf\n", max(a, b));

    return 0;
}
double max(double a, double b)
{
    if(a > b)
        return a;
    else
        return b;
}
```

【练习 5-3】数字金字塔：输入一个正整数 n，输出 n 行数字金字塔。试编写相应程序。

解答：

```
#include <stdio.h>
int main(void)
{
    int n,i,j;

    printf("    Enter n:    ");
    scanf("%d", &n);
    for(i = 1; i <= n; i++){
        for(j = 1; j <= n - i; j++)
            printf(" ");
        for(j = 1; j <= i; j++)
            printf("%d ", i);
        putchar('\n');
    }

    return 0;
}
```

【练习 5-4】 思考：若把例 5-9 中静态变量 f 定义成普通局部变量，还能实现计算 n！吗？请上机检验。若把 f 换成全局变量又会如何？

解答：

若把例 5-9 中静态变量 `f` 定义成普通局部变量，无法实现计算 $n!$ ，因为函数返回后 `f` 单元被系统回收，不再保存原有数据。若把 `f` 换成全局变量，则能实现计算 $n!$ ，因为 `f` 的作用范围和生命周期一直存在。

习题 5

一、选择题

1. 在 C 语言程序中，若对函数类型未加显式说明，则函数的隐含类型为 **D**。

A. `void` B. `double` C. `char` D. `int`

2. 下列程序的输出结果是 **D**。

```
fun(int a, int b, int c)
{ c = a * b; }
int main(void)
{ int c;
  fun(2, 3, c);
  printf(" %dn ", c);
  return 0;
}
```

A. 0 B. 1 C. 6 D. 无法确定

3. 建立自定义函数的目的之一是 **B**（优先）、**C**。

A. 提高程序的执行效率 B. 提高程序的可读性
C. 减少程序的篇幅 D. 减少程序文件所占内存

4. 以下正确的函数定义形式是 **A**。

A. `double fun(int x, int y)` B. `double fun(int x; int y)`
C. `double fun(int x, int y);` D. `double fun(int x, y)`

5. 以下不正确的说法是 **C**。

A. 实参可以是常量、变量或表达式
C. 形参可以是常量、变量或表达式
B. 实参可以是任何类型
D. 形参应与对应的实参类型一致

6. 以下正确的说法是 **D**。

A. 实参与其对应的形参共同占用一个存储单元
B. 实参与其对应的形参各占用独立的存储单元
C. 只有当实参与其对应的形参同名时才占用一个共同的存储单元
D. 形参是虚拟的，不占用内存单元

二、填空题

1．下列程序的输出结果为 2357 。

```
#include <stdio.h>
int fun3(int m);
int main(void)
{ int n;
  for (n = 1; n < 10; n++)
    if (fun3(n) == 1) printf("%d ", n);
  return 0;
}
int fun3(int m)
{ int i;
  if (m == 2 || m == 3) return 1;
  if (m < 2 || m % 2 == 0) return 0;
  for (i = 3; i < m; i = i + 2)
    if (m % i == 0) return 0;
  return 1;
}
```

2．输入若干个正整数，判断每个数从高位到低位各位数字是否按值从小到大排列，请根据题意，将程序补充完整。

```
#include <stdio.h>
int fun1( int m );
int main(void)
{ int n;
  scanf("%d", &n);
  while (n > 0) {
    if(fun1(n))
      printf("%d 中各位数字按从小到大排列 \n", n);
    scanf("%d", &n);
  }
}
int fun1( int m )
{ int k;
  k = m % 10;
  while(m)
    if( m / 10 % 10 > k)
      return 0;
    else { m = m/10 ; k = m % 10; }
  return 1;
}
```

3．下列函数输出一行字符：先输出 kb 个空格，再输出 n 个指定字符（由实参指定）。请根据题意，将程序补充完整。

```
#include <stdio.h>
void print( int kb, int n, char zf )
```

```

{   int i;
    for(i = 1; i <= kb; i++)
        printf(    "    ");
    for (i = 1; i <= n; i++ )
        printf(    " %c" , z f);
}

```

三、程序设计题

1．使用函数计算分段函数的值： 输入 x ，计算并输出下列分段函数 $f(x)$ 的值。
要求定义和调用函数 $\text{sign}(x)$ 实现该分段函数。试编写相应程序。

$$f(x) = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}$$

解答：

```

#include <stdio.h>
int main(void)
{
    int sign(int x);
    int x, y;
    scanf("%d",&x);
    y = sign(x);
    printf("sign(%d) = %d\n", x, y);
    return 0;
}

int sign(int x)
{
    int y;
    if(x>0) y=1;
    else if (x==0) y=0;
    else y=-1;
    return y;
}

```

2．使用函数求奇数和：输入一批正整数（以零或负数为结束标志），求其中的奇数和。要求定义和调用函数 $\text{even}(n)$ 判断数的奇偶性，当 n 为偶数时返回 1，否则返回 0。试编写相应程序。

解答：

```

#include <stdio.h>
int even(int n);
int main(void)
{
    int n, sum;
    scanf("%d",&n);
    sum = 0;
    while(n > 0){
        if(even(n) == 0) sum = sum + n;
    }
}

```

```

        scanf("%d", &n);
    }
    printf("The sum of the odd numbers is %d.\n", sum);

    return 0;
}
int even(int n)
{
    int y;
    if (n % 2 == 0) y = 1;
    else y = 0;
    return y;
}

```

3. 使用函数计算两点间的距离：给定平面任意两点坐标（ x_1, y_1 ）和（ x_2, y_2 ），求这两点之间的距离（保留 2 位小数）。要求定义和调用函数 `dist(x1, y1, x2, y2)` 计算两点之间的距离。试编写相应程序。

解答：

```

#include<stdio.h>
#include<math.h>
double dist(double x1,double y1,double x2,double y2);
int main(void)
{
    double x1,y1,x2,y2;

    printf("Enter x1 and y1:");
    scanf("%lf%lf",&x1,&y1);
    printf("Enter x2 and y2:");
    scanf("%lf%lf",&x2,&y2);
    printf("Distance=%.2lf",dist(x1,y1,x2,y2));

    return 0;
}
double dist(double x1,double y1,double x2,double y2)
{
    double distance;

    distance=sqrt(pow(fabs(x2-x1),2)+pow(fabs(y2-y1),2));

    return distance;
}

```

4. 利用函数计算素数个数并求和：输入两个正整数 m 和 n ($1 \leq m, n \leq 500$)，统计并输出 m 和 n 之间的素数的个数以及这些素数的和。要求定义并调用函数 `prime(m)` 判断 m 是否为素数。试编写相应程序。

解答：

```

#include<stdio.h>
#include<math.h>
int prime(int m);
int main(void)
{
    int m,n,sum,count;

    printf("Input m:");
    scanf("%d",&m);
    printf("Input n:");
    scanf("%d",&n);
    sum=0;
    count=0;
    for(;m<=n;m++){
        if(prime(m)==1){
            sum=sum+m;
            count++;
        }
    }
    printf("%d    和%d之间素数的个数： %d\n",m,n,count);
    printf("%d    和%d之间素数的和： %d\n",m,n,sum);

    return 0;
}
int prime(int m)
{
    int i;

    for(i=2;i<=sqrt(m);i++)
        if(m%i==0)return 0;
    if(i>h&& m!=1)return 1;
}

```

5. 使用函数统计指定数字的个数：读入一个整数，统计并输出该数中“2”的个数。要求定义并调用函数 `countdigit(number,digit)`，它的功能是统计整数 `number` 中数字 `digit` 的个数。例如，`countdigit(12292,2)` 的返回值是 3。试编写相应程序。

解答：

```

#include<stdio.h>
int countdigit(int number,int digit);
int main(void)
{
    int number,digit;

    printf("Enter an integer:");

```

```

scanf("%d",&number);
printf("Number of digit ");
scanf("%d",&digit);
printf(":%d",countdigit(number,digit));

return 0;
}
int countdigit(int number,int digit)
{
    int a,count;

    if(number<0)number=-number;
    while(number>0){
        a=number;
        if(number%10==digit)
            count++;
        number=number/10;
    }
    return count;
}

```

6 . 使用函数输出水仙花树： 输入两个正整数 m 和 $n(1 \leq m, n \leq 1000)$, 输出 $m \sim n$ 之间所有满足 各位 数字 的立方和等于它本身的 数。要求定 义并调 用函数 `is(number)` , 判断 `number`的各位数字之立方和是否等于它本身。试编写相应程序。

解答：

```

#include<stdio.h>
int is(int number);
int main(void)
{
    int i, m, n;

    printf("Input m and n:");
    scanf("%d%d", &m, &n);
    printf("result:\n");
    for(i = m; i <= n; i++)
        if(is(i) != 0)
            printf("%d\n", i);

    return 0;
}
int is(int number)
{
    int digit, sum, temp;

```

```

temp = number;
sum = 0;
while(temp != 0){
    digit = temp % 10;
    temp = temp / 10;
    sum = sum + digit * digit * digit;
}
return sum == number;
}

```

7. 使用函数求余弦函数的近似值：输入精度 e ，用下列公式求 $\cos(x)$ 的近似值，精确到最后一项的绝对值小于 e 。要求定义和调用函数 `funcos(e, x)` 求余弦函数的近似值。试编写相应程序。

$$\cos x = x^0/0! - x^2/2! + x^4/4! - x^6/6! + \dots$$

解答：

```

#include<stdio.h>
#include<math.h>
double funcos(double e, double x);
int main(void)
{
    double e, sum, x;

    printf("Input e and x:");
    scanf("%lf%lf",&e,&x);
    sum=funcos(e,x);
    printf("cos%lf=%lf\n",x,sum);

    return 0;
}
double funcos(double e,double x)
{
    int flag, i;
    double item, sum, tmp, power;

    flag=1;
    power=1;
    tmp=1;
    i=0;
    item=1;
    sum=0;
    while(fabs(item)>=e){
        item=flag*power/tmp;

```

```
    sum=sum+item;
    power=power*x*x;
    tmp=tmp*(i+1)*(i+2);
    flag=-flag;
    i=i+2;
}

return sum;
}
```


第 6 章 回顾数据类型和表达式

【练习 6-1】 输入一个十进制数，输出相应的八进制数和十六进制数。例如：输入 31，输出 37 和 1F。

解答：

```
#include <stdio.h>

int main(void)
{
    int x;

    scanf("%d", &x);

    printf("%o %x\n", x, x);

    return 0;
}
```

【练习 6-2】 在程序段：

```
printf("input a, b: ");

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

printf("%d %5d\n", a, b);    /* %5d 指定变量 b 的输出宽度为 5 */
```

中，如果将 `scanf("%o%d", &a, &b)` 改为 `scanf("%x%d", &a, &b)`，仍然输入 `__17 __17__`，输出是什么？

解答： 23 17(中间有三个空格)

【练习 6-3】如果字符型变量 ch 的值是大写字母， 怎样的运算可以把它转换为小写字母？

解答：

```
ch=ch- A + a ;
```

【练习 6-4】 证明下列等价关系。

(1) $a \&\&(b||c)$ 等价于 $a\&\&b||a\&\&c$ 。

(2) $a||(b\&\&c)$ 等价于 $(a||b) \&\&(a||c)$ 。

(3) $!(a\&\&b)$ 等价于 $! a||!b$ 。

(4) $!(a||c)$ 等价于 $! a\&\& ! b$ 。

解答：

(1)

a	b	c	$a\&\&(b c)$	$a\&\&b$	$a\&\&c$	$a\&\&b a\&\&c$
0(假)	0(假)	0(假)	0	0	0	0
0(假)	0(假)	非 0(真)	0	0	0	0
0(假)	非 0(真)	0(假)	0	0	0	0
0(假)	非 0(真)	非 0(真)	0	0	0	0
非 0(真)	0(假)	0(假)	0	0	0	0
非 0(真)	0(假)	非 0(真)	1	0	1	1
非 0(真)	非 0(真)	0(假)	1	1	0	1
非 0(真)	非 0(真)	非 0(真)	1	1	1	1

(2)

a	b	c	$b\&\&c$	$a (b\&\&c)$	$a\&\&b$	$a c$	$(a b)\&\&(a c)$
---	---	---	----------	---------------	----------	--------	--------------------

0(假)	0(假)	0(假)	0	0	0	0	0
0(假)	0(假)	非 0(真)	0	0	0	1	0
0(假)	非 0(真)	0(假)	0	0	1	0	0
0(假)	非 0(真)	非 0(真)	1	1	1	1	1
非 0(真)	0(假)	0(假)	0	1	1	1	1
非 0(真)	0(假)	非 0(真)	0	1	1	1	1
非 0(真)	非 0(真)	0(假)	0	1	1	1	1
非 0(真)	非 0(真)	非 0(真)	1	1	1	1	1

(3)

a	b	a&&b	!a&&b	!a	!b	!a b
0(假)	0(假)	0	1	1	1	1
0(假)	非 0(真)	0	1	1	0	1
非 0(真)	0(假)	0	1	0	1	1
非 0(真)	非 0(真)	1	0	0	0	0

(4)

a	b	a b	!(a b)	!a	!b	!a&&!b
0(假)	0(假)	0	1	1	1	1
0(假)	非 0(真)	1	0	1	0	0
非 0(真)	0(假)	1	0	0	1	0
非 0(真)	非 0(真)	1	0	0	0	0

习题 6

1 . 设 float x=2.5, y=4.7; int a=7; , printf(“ %.1f ”, x+a%3*(int)(x+y)%2/4) 的结果为 A

A . 2.5 B . 2.8 C . 3.5 D . 3.8

2 . 执行下列程序段的输出结果是 D 。

```
int num =1234,s=0;
```

```
while(num!=0){
```

```
    s+=num%10;
```

```
    num/=10;
```

```
}
```

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

A . 4 321 B . 1 234 C . 0 D . 10

3 . 设字符型变量 x 的值是 064 , 表达式 “ ~ x ^ x << 2 & x ” 的值是A 。

A . 0333 B . 333 C . 0x333 D . 020

4 . 设 a 为整型变量 , 不能正确表达数学关系 : 10<a<15 的 C 语言表达式是 A 。

A . 10<a<15

B . `a==11|| a==12 || a==13 || a==14`

C . `a>10 && a<15`

D . `!(a<=10) && !(a>=15)`

5 . 设以下变量均为 `int` 类型 , 表达式的值不为 9 的是 C 。

A . `(x=y=8,x+y,x+1)`

B . `(x=y=8,x+y,y+1)`

C . `(x=8,x+1,y=8,x+y)`

D . `(y=8,y+1,x=y,x+1)`

二 . 填空题

1 . -127 的原码为 `1000000001111111` 、反码为 `1111111110000000` 、补码为

`1111111110000001` 。

2. 逻辑表达式 `x && 1` 等价于关系表达式 `x!=0` 。

3 . 设 `int a=5, b=6;` 则表达式 `(++a==b--)? ++a : --b` 的值是 7 。

4 . 设 `c = 'w', a = 1, b = 2, d = -5,` 则表达式 `'x'+1>c, 'y'!=c+2, -a-5*b<=d+1, b==(a=2)`

的值分别为 1 、 0 、 1 、 1 。

5 . 运行以下程序后 , 如果从键盘上输入 `china#< 回车 >` , 则输出结果为 `c1=2,c2=5` 。

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

```
int main(void)
```

```
{
```

```

int c1 = 0, c2 = 0; char ch;

while((ch = getchar()) != ' # ' )

    switch(ch)

    {
        case ' a ' :

            case ' h ' : c1++;

            default: c2++;

    }

printf(    “ c1=%d,c2=%d” , c1, c2);

return 0;

}

```

三 . 程序设计题

1.分类统计字符个数：输入一行字符，统计出其中的英文字母、空格、数字和其他字符的个数。试编写相应程序。

解答：

```

#include<stdio.h>

int main(void)

{

    char op;

    int letter,blank,digit,other;

    printf("Input characters:");

    op=getchar();

```

```

letter=blank=0;

digit=other=0;

while(op!='\n'){

    if(op==' ')blank++;

    else if(op>='a'&&op<='z' || op>='A'&&op<='Z')letter++;

    else if(op>='0'&&op<='9')digit++;

    else other++;

    op=getchar();

}

printf("blank=%d,other=%d,letter=%d,digit=%d",blank,other,letter,digit);


return 0;

}

```

2.使用函数累加由 n 个 a 构成的整数之和：输入两个正整数 a 和 n ,求 $a+aa+aaa+aa\dots a(n$ 个 $a)$ 之和。要求定义和调用函数 $fn(a, n)$, 它的功能是返回 $aa\dots a(n$ 个 $a)$ 。例如, $fn(3,2)$ 的返回值是 33。试编写相应程序。

解答：

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

```
int main(void)
```

```
{
```

```
    int a, i, n;
```

```

long sn;

long fn(int a, int n);

scanf("%ld%d", &a, &n);

sn=0;

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

    sn = sn + fn(a, i);

printf("%ld\n", sn);


return 0;

}

long fn(int a, int n)

{

    int i;

    long tn;

    tn = 0;

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

        tn = 10 * tn + a;

    return tn;

}

```

3.使用函数输出指定范围内的函数：输入两个正整数 m 和 $n(1 \leq m, n \leq 1000)$ ，输出 $m \sim n$

之间的所有完数，完数就是因子和与它本身相等的数。要求定义并调用函数

factorsum(number) ，它的功能是返回 number 的因子和。 例如， factorsum(12) 的返回值是 16 (1+2+3+4+6)。 试编写相应程序。

解答：

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

```
int factorsum(int number);
```

```
int main(void)
```

```
{
```

```
    int m,n;
```

```
    printf("Input m:");
```

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

```
    printf("Input n:");
```

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

```
    printf(" 完全数是：  ")
```

```
    for(;m<=n;m++)
```

```
        if(factorsum(m)==1)
```

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

```
    return 0;
```

```
}
```

```
int factorsum(int number)
```

```
{
```

```

int i,sum;

if(number==1) return 1;

sum=0;

for(i=1;i<=number/2;i++)

    if(number%i==0)

        sum=sum+i;

if(sum==number)return 1;


return 0;

}

```

4.使用函数输出指定范围内的 Fibonacci 数：输入两个正整数 m 和 $n(1 \leq m, n \leq 10000)$ ，输出 $m \sim n$ 之间所有的 Fibonacci 数。Fibonacci 序列（第 1 项起）：1 1 2 3 5 8 13 21.....要求定义并调用函数 $\text{fib}(n)$ ，它的功能是返回第 n 项 Fibonacci 数。例如， $\text{fib}(7)$ 的返回值是 13。试编写相应程序。

解答：

```

#include<stdio.h>

int fib(int n);

#include<math.h>

int main(void)

{

    int i,m,n,f;

```

```
printf("Input m:");

scanf("%d",&m);

printf("Input n:");

scanf("%d",&n);

i=f=1;

while(f<=n){

    if(f>=m) printf("%d ",f);

    i++;

    f=fib(i);

}

return 0;

}

int fib(int n)

{

    int i,x1,x2,x;

    if(n==1||n==2) return 1;

    else{

        x1=x2=1;

        for(i=3;i<=n;i++){
```

```

        x=x1+x2;

        x1=x2;

        x2=x;

    }

}

return x;

}

```

5. 使用函数验证哥德巴赫猜想： 任何一个不小于 6 的偶数均可表示为两个奇素数之和。 例

如 $6=3+3$, $8=3+5$, ... , $18=5+13$ 。将 6 ~ 100 之间的偶数都表示成两个素数之和，打印

时一行打印 5 组。试编写相应程序。

解答：

```

#include<stdio.h>

int prime(int m);

int main(void)

{

    int count, i,number;

    count=0;

    for(number=6;number<=100;number=number+2){

        for(i=3;i<=number/2;i=i+2)

            if(prime(i)&&prime(number-i)){

                printf("%d=%d+%d    ",number,i,number-i);

```

```

        count++;

        if(count%5==0) printf("\n");

    }

}

return 0;

}

int prime(int m)

{

    int k,i;

    for(i=2;i<=m/2;i++)

        if(m%i==0)return 0;

        if(i>m/2&& m!=1)return 1;

}

```

6.使用函数输出一个整数的逆序数：输入一个整数，将它逆序输出。要求定义并调用函数

reverse(number) ，它的功能是返回 number 的逆序数。例如， reverse(12345) 的返回值是 54321 。试编写相应程序。

解答：

```

#include<stdio.h>

int reverse(int number);

int main(void)

{

```

```
int number;

printf("Input number:");

scanf("%d",&number);

printf("%d",reverse(number));

return 0;
}

int reverse(int number)
{

int res,a;

res=0;

while(number!=0){

a=number;

number=number%10;

res=res*10+number;

number=a/10;}

return res;

}
```

7.简单计算器：模拟简单运算器的工作：输入一个算式（没有空格），遇等号 "=" 说明输入结

束，输出结果。假设计算器只能进行加、减、乘、除运算，运算数和结果都是整数，4 种运算符的优先级相同，按从左到右的顺序计算。例如，输入 “1+2*10/2= ” 后，输出 10。试编写相应程序。

解答：

```
#include <stdio.h>

int main(void)
{

    char op;

    int operand1,operand2,res;

    scanf("%d",&operand1);

    op=getchar();

    while(op!='='){

        scanf("%d",&operand2);

        switch(op){

            case '+': res=operand1+operand2; break;

            case '-': res=operand1-operand2; break;

            case '*': res=operand1*operand2; break;

            case '/': res=operand1/operand2; break;

            default: res=0;

        }

    }
```

```
        operand1=res;

        op=getchar();

    }

    printf("%d",res);


    return 0;

}
```

8. 统计一行文本的单词个数：输入一行字符，统计其中单词的个数。各单词之间用空格分隔，空格数可以是多个，试编写相应程序。

解答：

```
#include <stdio.h>

int main(void)

{

    char op;

    int count,word;


    printf("Input words:");

    op=getchar();

    count=word=0;

    while(op!='\n'){

        if(op==' ')

            word=0;
```



```
else if(word==0){  
  
    count++;  
  
    word=1;  
  
}  
  
op=getchar();  
  
}  
  
printf("count=%d\n",count);  
  
  
return 0;  
  
}
```

第 7 章 数组

【练习 7-1】将例 7-3 程序中的 break 语句去掉，输出结果有变化吗？假设输入数据不变，输出什么？

解答：

当去掉 break 语句后，一旦找到数据将不跳出循环，而是继续往下找值为 x 的元素，因此程序输出会有变化。当输入数据仍为 2 9 8 1 9 时，输出将是 index is 1 index is 4 。

【练习 7-2】将数组中的数逆序存放。输入一个正整数 n ($1 < n \leq 10$) ,再输入 n 个整数，存入数组 a 中，输出最大值及其对应的最小下标，下标从 0 开始。试编写相应程序。

解答：

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

```
int main(void)
```

```
{
```

```
    int i,n,temp;
```

```
    int a[10];
```

```
    printf("Input n:");
```

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

```
    printf("Input %d integer:",n);
```

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

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

```
    printf("After reversed:");
```

```

        for(i=0;i<n/2;i++){

            temp=a[i];

            a[i]=a[n-1-i];

            a[n-1-i]=temp;

        }

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

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


        return 0;

    }

```

【练习 7-3】求最大值及其下标。输入一个正整数 n ($1 < n \leq 10$), 再输入 n 个整数, 存入数组 a 中, 将先数组 a 中的这 n 个数逆序存放, 在按顺序输出数组 a 中的 n 各元素。试编写相应程序。

解答：

```

#include<stdio.h>

int main(void)

{

    int i,index,n;

    int a[10];


    printf("Enter n:");

    scanf("%d",&n);

```

```

printf("Enter %d integrs:",n);

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

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

for(i=n-1;i>=0;i++)

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


return 0;

}

```

【练习 7-4】找出不是两个数组共有的元素。输入一个正整数 n ($1 < n \leq 10$)，再输入 n 个整数，存入第 1 个数组中；然后输入一个正整数 m ($1 \leq m \leq 10$)，再输入 m 个整数，存入第 2 个数组，找出所有不是这两个数组共有的元素。试编写相应程序。

解答：

```

#include<stdio.h>

int main(void)

{

    int i,j,k,m,n,flag,equal;

    int a[25],b[25],c[25];


    printf("Enter m:");

    scanf("%d",&m);

    printf("Enter %d integers:",m);

```

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

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

printf("Enter n:");

scanf("%d",&n);

printf("Enter %d integers:",n);

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

    scanf("%d",&b[j]);

    k=0;

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

    flag=0;

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

        if(a[i]==a[j])

            flag=1;// 判断一个数组中是否有相重的元素，有的话不执行

    if(!flag){

        equal=0;

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

            if(a[i]==b[j]){

                equal=1;

                break;

            }

        if(!equal)

            c[k++]=a[i];
```

```
        }

    }

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

        flag=0;

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

            if(b[i]==b[j])

                flag=1;

        if(!flag){

            equal=0;

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

                if(b[i]==a[j]){

                    equal=1;

                    break;

                }

            if(!equal)

                c[k++]=b[i];

        }

    }

    printf("%d",c[0]);

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

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

    printf("\n");
```

```
    return 0;

}
```

【练习 7-5】给二维数组赋值时，如果把列下标作为外循环的循环变量，行下标作为内循环的循环变量，输入的数据在二维数组中如何存放？用下列 for 语句替换例 7-7 中的对应语句，将输入的 6 个数存入二维数组中，假设输入数据不变，输出什么？与例 7-7 中的输出结果一样吗？为什么？

```
for( j=0;j<2;j++)

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

scanf( " %d" ,&a[i][j]);
```

解答：

当把列下标作为外循环的循环变量，行下标作为内循环的循环变量时，输入的数据将以列优先的方式存放。当用上述 for 循环方式时，输出结果为：max=a[2][0]=10，与原例 7-7 不一样，因为当用上述方式输入是，二维数组中存放值如下：

3	- 9
2	6
10	- 1

【练习 7-6】在例 7-9 的程序中，如果将遍历上三角矩阵改为遍历下三角矩阵，需要怎样修改程序？运行结果有变化吗？如果改为遍历整个矩阵，需要怎样修改程序？输出是什么？为什么？

解答：

只需按要求修改矩阵的输出部分，方法如下，其运行结果不变。

```

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

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

        temp = a[i][j];

        a[i][j] = a[j][i];

        a[j][i] = temp;

    }

```

若修改为遍历整个程序，方法如下，则运行结果仍将输出原矩阵，无法达到转置要求，

原因是矩阵中每个元素相应被交换了 2 次。

```

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

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

        temp = a[i][j];

        a[i][j] = a[j][i];

        a[j][i] = temp;//// 先是下三角进行交换，后是上三角进行交换

    }

```

【练习 7-7】矩阵运算：读入 1 个正整数 $n(1 \leq n \leq 6)$ ，再读入 n 阶方阵 a ，计算该矩阵除副对角

线、最后一列和最后一行以外的所有元素之和。副对角线为从矩阵的右上角至左下角的连线。

试编写相应程序。

解答：

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

```
int main(void)
```

```
{
```

```
    int i,j,n,sum;
```



```

int a[6][6];

printf("Input n:");

scanf("%d",&n);

printf("Input array:\n ");

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

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

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

sum=0;

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

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

        if(i!=n-1&&j!=n-1&&i+j!=n-1) sum+=a[i][j];

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

return 0;

}

```

【练习 7-8】方阵循环右移。读入 2 个正整数 m 和 n ($1 \leq n < 6$)，在读入 n 阶方阵 a ，将该方阵中的每个元素循环向右移 m 个位置，即将第 $0、1、\dots、n-1$ 列变换为第 $n-m、n-m+1、\dots、n-1、0、1、\dots、n-m-1$ 列，移动后的方阵可以存到另一个二维数组中。试编写相应程序。

解答：

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

```
int main(void)
```

```
{

    int m,n,i,j,count;

    int a[6][6],b[6][6];


    printf("Enter m:");

    scanf("%d",&m);

    printf("Enter n:");

    scanf("%d",&n);

    printf("Input array:\n");

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

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

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

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

        count=0;

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

            if(j<n-m)

                b[i][j+m]=a[i][j];

            else

                b[i][count++]=a[i][j];

    }

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

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

```

        printf("%d ",b[i][j]);

    printf("\n");

}

return 0;

}

```

【练习 7-9】计算天数：输入日期（年、月、日），输出它是该年的第几天。要求调用例 7-10

中定义的函数 `day_of_year(year, month, day)`。试编写相应程序。

解答：

```

#include<stdio.h>

int day_of_year(int year, int month, int day);

int main(void)

{

    int year,month,day,day_year;


    printf("Input year,month,day: ");

    scanf("%d%d%d",&year,&month,&day);

    day_year=day_of_year(year,month,day);

    printf("Days of year: %d\n",day_year);


    return 0;

}

```

```

int day_of_year(int year,int month,int day)

{

    int k,leap;

    int tab[2][13]={

        {0,31,28,31,30,31,30,31,31,30,31,30,31},

        {0,31,29,31,30,31,30,31,31,30,31,30,31}

    };


    leap=(year%4==0&&year%100!=0||year%400==0);

    for(k=1;k<month;k++)

        day=day+tab[leap][k];


    return day;

}

```

【练习 7-10】查找指定字符。输入一个字符，再输入一个以回车符结束的字符串（少于 80 个字符）。如果找到，则输出该字符在字符串中所对应的最大下标，下标从 0 开始；否则输出 "Not Found"。试编写相应程序。

解答：

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

```
int main(void)
```

```
{
```

```
int i,max,flag;
```

```
char a[80],op;
```

```
printf("Enter a x:");
```

```
op=getchar();
```

```
fflush(stdin);// 或 setbuf(stdin,NULL);
```

```
printf("Enter a string:");
```

```
i=0;
```

```
while((a[i]=getchar())!='\n')
```

```
    i++;
```

```
a[i]='\0';
```

```
max=0;
```

```
flag=0;
```

```
for(i=0;a[i]!='\0';i++)
```

```
    if(a[i]==op) {
```

```
        max=i;
```

```
        flag=1;
```

```
    }
```

```
if(flag==1)
```

```
    printf("Max=%d",max);
```

```
else
```

```
    printf("Not Found!");
```

```
    return 0;

}
```

【练习 7-11】字符串逆序：输入一个以回车符结束的字符串（少于 80 个字符），将该字符串逆序存放，输出逆序后的字符串。试编写相应程序。

解答：

```
#include<stdio.h>

int main(void)
{
    int i;

    char str[80],a[80];

    printf("Enter a string:");

    i=0;

    while((str[i]=getchar())!='\n')

        i++;

    str[i]='\0';

    for(;i>=0;i--)

        putchar(str[i]);

    return 0;

}
```

习题 7

一．选择题

1．假定 int 类型变量占用两个字节，其有定义：`int x[10]={0,2,4};` 则数组 x 在内存中所占字节数是 D 。

A．3 B．6 C．10 D．20

2．以下能正确定义数组并正确赋初值的语句是 D 。

A．`int N=5,b[N][N];`

B．`int a[1][2]={{1},{3}};`

C．`int c[2][]={{1,2},{3,4}};`

D．`int d[3][2]={{1,2},{34}};`

3．若有定义：`int a[2][3];` 以下选项中对数组元素正确引用的是 D 。

A．`a[2][0]` B．`a[2][3]` C．`a [0][3]` D．`a[1>2][1]`

4．设有数组定义：`char array []="China";` 则数组 array 所占的空间为 C 。

A．4 个字节 B．5 个字节 C．6 个字节 D．7 个字节

5. 下述对 C 语言字符数组的描述中错误的是 D 。

A. 字符数组可以存放字符串

B. 字符数组中的字符串可以整体输入、输出

C. 可以在赋值语句中通过赋值运算符 "=" 对字符数组整体赋值

D. 不可以用关系运算符对字符数组中的字符串进行比较

6. 有以下定义： char x[]="abcdefg"; char y[]={'a','b','c','d','e','f','g'}; 则正确的叙述为

C 。

A. 数组 x 和数组 y 等价

B. 数组 x 和数组 y 的长度相同

C. 数组 x 的长度大于数组 y 的长度

D. 数组 x 的长度小于数组 y 的长度

7. 以下程序的输出结果是 C 。

```
int main(void)

{   int   m[ ][3]={1,4,7,2,5,8,3,6,9};

    int   i, j, k=2;

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

        printf ("%d ",m[k][i]);

}
```

A. 4 5 6

B. 2 5 8

C. 3 6 9

D. 7 8 9

8 . 以下程序的输出结果是 B 。

```
int main(void)

{      int aa[4][4]={1,2,3,4},{5,6,7,8},{3,9,10,2},{4,2,9,6}};

        int i, s=0;

        for(i=0;i<4;i++)

            s+=aa[i][1];

        printf("%d\n",s);

}
```

A . 11 B . 19 C . 13 D . 20

二 . 填空题

1 . 设有定义语句： int a[][3]={0},{1},{2}}; 则数组元素 a[1][2] 的值是 0 。

2 . 下列程序的功能是：求出数组 x 中各相邻两个元素的和依次存放到 a 数组中，然后输出。

请填空。

```
int main(void )

{      int x[10],a[9],i;

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

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

        for( _i=1_, i<10; i++ )
```

```

        a[i-1]=x[i]+_x[i-1] _ ;

for(i=0; i<9; i+ +)

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

printf("\n");

}

```

3 . 写出下列程序的运行结果是 -12 。

```

int main(void)

{
    int a[10]={10,1,-20,-203,-21,2,-2,-2,11,-21};

    int j,s=0;

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

        if(a [j]%2==0)

            s+=a[j];

    printf("s=%d\n",s);

}

```

4 . 写出下面程序的运行结果 。

```

#include <stdio.h>

int main(void)

{
    float s[6]={1, 3, 5, 7, 9};

    float x;

    int i;

    scanf( “ %f ” , &x);

    for (i=4 ; i>=0; i--)

```

```

        if(s[i]>x)

            s[i+1]=s[i];

        else

            break;

        printf("%d \n",i+1);


        return 0;

}

```

(1) 如果输入 4 , 则输出 2 。

(2) 如果输入 5 , 则输出 3 。

5 . 下列程序的功能是输出如下形式的方阵 , 请填空。

13 14 15 16

9 10 11 12

5 6 7 8

1 2 3 4

```
int main(void)
```

```
{     int i,j,x;
```

```
    for(j=4;_j>=1_;j--)
```

```
    {   for(i=1;i<=4;i++)
```

```
        {   x=(j-1)*4+_i_;
```

```
            printf("%4d",x);
```

```

        }

        printf("\n");

    }

}

```

6 . 以下程序的功能是用来检查二维数组是否对称（即对所有的 i, j 都有 $a[i][j]=a[j][i]$ ）。请填空。

```

int main(void)

{
    int a[4][4]={1,2,3,4,2,2,5,6,3,7,8,6,7,4};

    int i,j,found=0;

    for(j=0;j<4;j++)

    {
        for(i=0;i<4;i++)

            if(_a[j][i]!=a[i][j]_) {

                _found=1;_

                break;

            }

        if(_found==1_)break;

    }

    if(found!=0) printf(" 该二维数组不对称 \n");

    else printf(" 该二维数组对称 \n");

    return 0;

}

```

三 . 程序设计题

1. 选择法排序。输入一个正整数 n ($1 < n \leq 10$)，再输入 n 个整数，将它们从大到小排序后输出。试编写相应程序。

解答：

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

```
int main(void)
```

```
{
```

```
    int i,n,index,k,temp;
```

```
    int a[10];
```

```
    printf("Enter n:");
```

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

```
    printf("Input %d integers:",n);
```

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

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

```
    for(k=0;k<n-1;k++){
```

```
        index=k;
```

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

```
            if(a[index]<a[i]) index=i;
```

```
        temp=a[index];
```

```
        a[index]=a[k];
```

```
        a[k]=temp;
```

```
    }
```

```

printf("After sorted:");

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

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

printf("\n");


return 0;

}

```

2. 求一批整数中出现最多的数字。输入一个正整数 n ($1 \leq n \leq 1000$) , 再输入 n 个整数 , 分析每个整数的每一位数字 , 求出现次数最多的数字。例如输入 3 个整数 1234 、 2345 、 3456 , 其中出现次数最多的数字是 3 和 4 , 均出现了 3 次。试编写相应程序。

解答 :

```

#include<stdio.h>

int main(void)

{

    int i,j,k,g,n,max;

    int a[1000],b[1000],count[1000];


    printf("Enter n:");

    scanf("%d",&n);

    printf("Enter %d integers:",n);

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

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

```

```
for(i=0;i<=9;i++)

    count[i]=0;

for(i=0;i<=9;i++)

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

        for(k=0;k<n;k++)

            b[k]=a[k];

        while(a[j]!=0){

            if(a[j]%10==i)

                count[i]++;

            a[j]=a[j]/10;

        }

        for(k=0;k<n;k++)

            a[k]=b[k];

    }

max=count[0];

for(i=0;i<=9;i++)

    if(max<count[i])

        max=count[i];

printf(" 出现最多次数的数字和次数是 :    ");

for(i=0;i<=9;i++)

    if(max==count[i])

        printf("%2d-%d",i,count[i]);
```

```
    return 0;

}
```

3. 判断上三角矩阵。输入一个正整数 n ($1 \leq n \leq 6$) 和 n 阶方阵 a 中的元素，如果 a 是上三角矩阵，输出“YES”，否则，输出“NO”。上三角矩阵，即主对角线以下的元素都为 0 的矩阵，主对角线为从矩阵的左上角至右下角的连线。试编写相应程序。

解答：

```
#include <stdio.h>

int main(void)
{
    int i,j,n,flag;

    int a[6][6];

    printf("Input n:");

    scanf("%d",&n);

    printf("Input array:\n");

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

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

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

    for(j=1;j<n;j++){

        flag=0;

        for(i=1;i<j;i++)
```



```

        if(a[i][j]==0)

            flag=1;

    }

    if(flag)

        printf("YES!\n");

    else

        printf("NO!\n");

    return 0;

}

```

4 . 求矩阵各行元素之和。输入 2 个正整数 m 和 n (1 ≤ m ≤ 6, 1 ≤ n ≤ 6) , 然后输入该 m 行 n 列矩阵 a 中的元素 , 分别求出各行元素之和 , 并输出。试编写相应程序。

解答 :

```

#include<stdio.h>

int main(void)

{

    int i,j,m,n,sum;

    int a[6][6];

    printf("Input m:");

    scanf("%d",&m);

    printf("Input n:");

```

```

scanf("%d",&n);

printf("Input array:\n ");

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

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

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

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

    sum=0;

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

        sum=sum+a[i][j];

    printf("sum of row %d is %d\n",i,sum);

}

return 0;

}

```

5 . 找鞍点。输入一个正整数 $n(1 \leq n \leq 6)$ 和 n 阶方阵 a 中的元素，假设方阵 a 最多有 1 个鞍点，如果找到 a 的鞍点，就输出它的下标；否则，输出 "NO"。鞍点的元素值在该行上最大，在该列上最小。试编写相应程序。

解答：

```

#include<stdio.h>

int main(void)

{

    int flag,i,j,k,row,col,n;

```

```
int a[6][6];
```

```
printf("Input n: ");
```

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

```
printf("Input array:\n ");
```

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

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

```
        scanf("%d",&a[i][j]);
```

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

```
    flag=1;
```

```
    col=0;
```

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

```
        if(a[i][col]<a[i][j]) col=j;
```

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

```
        if(a[i][col]>a[k][col]){
```

```
            flag=0;
```

```
            break;
```

```
        }
```

```
    if(flag){
```

```
        row=i;
```

```
        break;
```

```
    }
```

```

    }

    if(flag)

        printf("a[%d][%d]=%d\n",row,col,a[row][col]);

    else

        printf("NO\n");

    return 0;

}

```

7．字符串替换。输入一个以回车结束的字符串（少于 80 个字符），将其中的大写字母用

下面列出的对应大写字母替换，其余字符不变，输出替换后的字符串。试编写相应程序。

原字母	对应字母
A	Z
B	Y
C	X
D	W
.....	
X	C
Y	B
Z	A

解答：

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

```
int main(void)
```

```

{

    int i;

    char ch,str[80];


    printf("Input a string: ");

    i=0;

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

        str[i++]=ch;

    }

    str[i]='\0';

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

        if(str[i]<='Z'&&str[i]>='A')

            str[i]='A'+'Z'-str[i];

    printf("After replaced:");

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

        putchar(str[i]);

    putchar('\n');


    return 0;

}

```

8 . 字符串转换成十进制整数。 输入一个以字符 ” # ” 结束的字符串滤去所有的非十六进制字符

(不分大小写) ， 组成一个新的表示十六进制数字的字符串 ， 然后将其转换为十进制数后输

出。如果过滤后字符串的首字符为 ” - ” , 代表该数是负数。试编写相应程序。

解答：

```
#include<stdio.h>

int main(void)

{

    int i,j,k;

    char hexad[80],str[80];

    long number;

    printf("Enter a string:");

    i=0;

    while((str[i]=getchar())!='#')

        i++;

    str[i]='\0';

    k=0;

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

        if(str[i]=='-'||str[i]>='0'&&str[i]<='9'||str[i]>='a'&&str[i]<='f'||str[i]>='A'&&str[i]<='F')

            hexad[k++]=str[i];

    hexad[k]='\0';

    j=1;

    if(hexad[0]=='-'){

        for(k=1;hexad[k]!='\0';k++)
```

```
        if(hexad[k]!='-')

            hexad[j++]=hexad[k];

    }

    else{

        j=0;

        for(k=0;hexad[k]!='\0';k++)

            if(hexad[k]!='-')

                hexad[j++]=hexad[k];}

    printf("New string:");

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

        putchar(hexad[i]);

    printf("\n");

    number=0;

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

        if(hexad[i]>='0'&&hexad[i]<='9')

            number=number*16+hexad[i]-'0';

        else if(hexad[i]>='A'&&hexad[i]<='F')

            number=number*16+hexad[i]-'A'+10;

        else if(hexad[i]>='a'&&hexad[i]<='f')

            number=number*16+hexad[i]-'a'+10;

    }

    if(hexad[0]!='-')
```

```
printf("Number=%ld\n",-number);
```

```
else
```

```
printf("Number=%ld\n",number);
```

```
return 0;
```

```
}
```


第 8 章 指针

【练习 8-1】如果有定义 “ `int m, n = 5, *p = &m;` ” 与 `m = n` 等价的语句是 B 。

A. `m = *p;` B. `*p = *&n;` C. `m = &n;` D. `m = **p;`

解答：

A:`p` 是指向 `m` 的指针变量，所以 `*p` 等价于 `m`，即 `m=m`

B:`&n` 是 `n` 的地址，`*&n` 是 `n` 的值，即把 `n` 的值赋给 `p` 指向的值 `m`，即 `m=n`

C:`&n` 是 `n` 的地址。即把 `n` 的地址赋给 `m`。

D:`**p` 是指 `p` 指向的指针所指向的值，在此无意义。

故选 B。

【练习 8-2】调用函数求两个数的和与差：计算输入的两个数的和与差，要求自定义一个函数 `sum_diff(float op1, float op2, float *psum, float *pdiff)`，

其中 `op1` 和 `op2` 是输入的两个数，`*psum` 和 `*pdiff` 是计算得出的和与差。

解答：

```
#include<stdio.h>
void sum_diff(float op1, float op2, float *psum, float *pdiff);
int main(void)
{
    float op1, op2, sum, diff;

    printf("Input op1 and op2: ");
    scanf("%f%f", &op1, &op2);
    sum_diff(op1, op2, &sum, &diff);
    printf("%f+%f=%f; %f-%f=%f\n", op1, op2, sum, op1, op2, diff);

    return 0;
}
void sum_diff(float op1, float op2, float *psum, float *pdiff)
{
    *psum=op1+op2;
    *pdiff=op1-op2;
}
```

【练习 8-3】两个相同类型的指针变量能不能相加？为什么？

解答：

不能。因为指针变量是一种特殊的变量，指针变量的值存放的是所指向变量的地址，两个地址相加并不能保证结果为一个有效的地址值，因而在 C 语言中指针变量相加是非法的。

【练习 8-4】根据表 8.2 所示，这组数据的冒泡排序其实循环到第 6 遍（即 `n-2`）时就已经排好序了，说明有时候并不一定需要 `n-1` 次循环。请思考如何改进冒泡排序算法并编程实现（提示：当发现一遍循环后没有数据发生交换，说明已经排好序了）。

解答：

设置一个标志变量 flag，进入一轮循环前设置为 0，在循环中有发生数据交换就改写 flag 值为 1。当该轮循环结束后检查 flag 值，如果变为 1 说明发生了数据交换，还没有排好序，如果为 0 说明没有发生交换，已经排好序。

```
#include<stdio.h>
void bubble (int a[],int n);
int main(void)
{
    int n,i,a[8];

    printf("Enter n (n<=8):");
    scanf("%d",&n);
    printf("Enter a[%d] : ",n);
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
    bubble(a,n);
    printf("After sorted, a[%d]=", n);
    for(i=0;i<n;i++)
        printf("%3d",a[i]);

    return 0;
}
void bubble(int a[], int n)
{
    int i,j,temp,flag;

    for(i=1;i<n;i++){
        flag=0;
        for(j=0;j<n-i;j++){
            if(a[j]>a[j+1]){
                temp=a[j];
                a[j]=a[j+1];
                a[j+1]=temp;
                flag=1;
            }
        }
        if(flag==0)
            break;
    }
}
```

【练习 8-5】重做例 8-9，要求使用选择排序算法。

解答：

```
#include<stdio.h>
void bubble(int a[], int n);
```

```

int main(void)
{
    int i,n,a[8];

    printf("Enter n (n<=8): ");
    scanf("%d",&n);
    printf("Enter a[%d] :",n);
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
    bubble(a,n);
    printf("After sorted, a[%d]=",n);
    for(i=0;i<n;i++)
        printf("%3d",a[i]);

    return 0;
}
void bubble(int a[],int n)
{
    int i,j,temp,index;

    for(i=0;i<n-1;i++){
        index=i;
        for(j=i+1;j<n;j++)
            if(a[j]<a[index]) index=j;
        temp=a[i];
        a[i]=a[index];
        a[index]=temp;
    }
}

```

8.4 电码加密

【练习 8-6】在使用 scanf() 函数时，输入参数列表需要使用取地址操作符 &，但当参数为字符数组名时并没有使用，为什么？如果在字符数组名前加上取地址操作符 &，会发生什么？

解答：

因为字符数组名的值是一个特殊的固定地址，可以看作是常量指针，因此不需要再使用取地址符来获取该数组的地址。如果在字符数组名 str 前加上取地址操作符 &，那么对其取地址 &str 可以看做是这个数组的第一个元素的地址，由于数组地址和数组第一个元素的地址相同，所以 &str 表示地址值和 str 表示的地址值是相等的。对 scanf() 的变长参数列表的话，编译器只负责参数传递，怎么解释后边的几个地址的含义，是由前边的字符串确定的。所以使用 scanf(“%s”,str) 和 scanf(“%s",&str) 都能通过编译且正常执行。

【练习 8-7】C 语言不允许用赋值表达式直接对数组赋值，为什么？

解答：

数组名可以看作是常量指针，因为不可以对一个常量进行赋值，所以不允许用赋

值表达式直接对数组进行赋值。

【练习 8-8】输入一个字符串，把该字符串的前 3 个字母移到最后，输出变换后的字符串。比如输入 “ abcdef ”，输出为 “ defabc ”。

解答：

```
#include<stdio.h>
#include<string.h>
#define MAXLINE 100
int main(void)
{
    char line[MAXLINE],str[4];
    int i;

    printf("Input the string:");
    gets(line);
    if(strlen(line)<3){
        printf("      字符串长度小于 3，不符合要求！ \n");
    }
    for(i=0;i<3;i++)
        str[i]=line[i];
    str[i]='\0';
    for(i=3;line[i]!='\0';i++)
        line[i-3]=line[i];
    line[i-3]='\0';
    strcat(line,str);
    printf("%s%s\n","After changing: ",line);

    return 0;
}
```

【练习 8-9】使用动态内存分配的方法实现例 8-9 的冒泡排序。

解答：

```
#include<stdio.h>
#include<stdlib.h>
void bubble(int a[],int n);
int main(void)
{
    int n,j,*a,i,temp;

    printf("Enter n(n<=8):");
    scanf("%d",&n);
    if((a=(int*)calloc(n,sizeof(int)))==NULL){
        printf("Not able to allocate memory.\n");
        exit(1);
    }
    printf("Ente a[%d]:",n);
```

```

        for(i=0;i<n;i++)
            scanf("%d",a+i);
        bubble(a,n);
        printf("After sorted,a[%d]=",n);
        for(i=0;i<n;i++)
            printf("%3d",*(a+i));
        free(a);

        return 0;
    }
void bubble(int a[],int n)
{
    int i,j,temp;

    for(i=1;i<n;i++)
        for(j=0;j<n-i;j++)
            if(*(a+j)>*(a+j+1)){
                temp=*(a+j);
                *(a+j)=*(a+j+1);
                *(a+j+1)=temp;
            }
}

```

习题 8

一、选择题

1. 下列语句定义 `x` 为指向 `int` 类型变量 `a` 的指针，其中哪一个是正确的
B 。

- A . `int a, *x = a;` B. `int a, *x = &a;`
C . `int *x = &a,a;` D . `int a, x = a;`

2. 以下选项中，对基本类型相同的指针变量不能进行运算的运算符是 A 。

- A . `+` B . `-` C . `=` D . `==`

3 . 若有以下说明，且 $0 \leq i < 10$ ，则对数组元素的错误引用是 C 。

`int a[] = {0,1,2,3,4,5,6,7,8,9}, *p = a,i;`

- A . `*(a+i)` B . `a[p-a+i]` C . `p+i` D . `*(&a[i])`

4 . 下列程序的输出结果是 B 。

```

int main(void)
{
    int a[10] = {0,1,2,3,4,5,6,7,8,9}, *p = a+3;
    printf(    " %d" , *++p);
}

```

```
    return 0;
}
```

A . 3 B . 4 C . a[4] 的地址 D . 非法

5 . 对于下列程序 , 正确的是 A 。

```
void f(int *p)
{
    *p = 5;
}
int main(void)
{
    int a, *p;
    a = 10;
    p = &a;
    f(p);
    printf(    “ %d” , (*p)++);
    return 0;
}
```

A . 5 B . 6 C . 10 D . 11

二、填空题

1. 下列函数在一维数组 a 中将 x 插入到下标为 i (i >= 0) 的元素前。如果 i >= 元素的个数 , 则 x 插入到末尾。原有的元素个数存放在指针 n 所指向的变量中 , 插入后元素个数加 1 。请填空。

```
void insert(double a[ ], int *n, double x, int i)
{
    int j;
    if _ (i < *n)_
        for (j = *n - 1; j >= i; j--)
            _a[j + 1]_ = a[j];
    else
        i = *n;
    a[i] = _x_;
    (*n)++;
}
```

2 . 下列程序先消除输入字符串的前后空格 , 再判断是否是 “ 回文 ” (即字符串正读和倒读都是一样) , 若是则输出 YES , 否则输出 NO。请填空。

```
#include <stdio.h>
#include <string.h>
int main(void)
{
    char s[80], ch, *p, *q;
    int i, j, n;
```

```

gets(s);
p = _s_;
while ( *p == ' ' ) _p++;
n = strlen(s);
q = _s+n-1_;
while ( *q == ' ' ) _q--;
while ( _p<q_ && *p==*q)
{
    p++;
    _q--;
}
if ( p<q )
    printf( " NO n " );
else
    printf( " YES n " );
return 0;
}

```

3. 下列程序在数组中同时查找最大元素和最小元素的下标，分别存放在 main() 函数的 fmax 和 min 变量中。请填空。

```

void find(int *, int, int *, int *);
int main(void)
{
    int max, min, a[ ]={5,3,7,9,2,0,4,1,6,8};
    find(_a, 10, &max, &min);
    printf( " %d,%dn " , max, min);
    return 0;
}
void find(int *a, int n, int *max, int *min)
{
    int i;
    *max=*min=0;
    for (i = 1; i < n; i++)
    {
        if (a[i] > a [*max]) _*max=i_;
        if (a[i] < a [*min]) _*min=i_;
    }
}

```

4 . 写出下列程序的执行结果

0	0
1	2
2	4
0	0
2	3
4	6

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

```

int main(void)
{
    int a[10], b[10], *pa, *pb, i;
    pa = a;
    pb = b;
    for( i=0; i<3; i++, pa++, pb++)
    {
        *pa = i;
        *pb = 2*i;
        printf(    " %d      n " , *pa, *pb);
    }
    pa = &a[0];
    pb = &b[0];
    for ( i=0; i<3; i++)
    {
        *pa = *pa + i;
        *pb = *pb + i;
        printf(    " %d      n " , *pa++, *pb++);
    }
    return 0;
}

```

三、程序设计题

1. 拆分实数的整数与小数部分：要求定义一个函数 `void splitfloat(float x,int *intpart,float *fracpart)`, 其中 `x` 是被拆分的实数, `*intpart` 和 `*fracpart` 分别是将实数 `x` 拆分出来的整数部分与小数部分。编写主函数, 并在其中调用函数 `splitfloat` ()。试编写相应程序。

解答：

```

#include<stdio.h>
void splitfloat(float x,int *intpart,float *fracpart);
int main(void)
{
    float x,fracpart;
    int intpart;

    printf("Input a number:");
    scanf("%f",&x);
    splitfloat(x,&intpart,&fracpart);
    printf("The intpart is: %d",intpart);
    printf("The fracpart is: %f",fracpart);

    return 0;
}
void splitfloat(float x,int *intpart,float *fracpart)
{

```



```

    *intpart=(int)x;
    *fracpart=x-*intpart;
}

```

2. 在数组中查找指定元素：输入 1 个正整数 n ($1 < n \leq 10$)，然后输入 n 个整数存入数组 a 中，再输入一个整数 x ，在数组 a 中查找 x ，若找到则输出相应的下标，否则显示“Not found”。要求定义和调用函数 `search(int list[], int n, int x)`，在数组 `list` 中查找元素 x ，若找到则返回相应下标，否则返回 `-1`，参数 n 代表数组 `list` 中元素的数量。试编写相应程序。

解答：

```

#include<stdio.h>
int search(int list[],int n,int x);
int main(void)
{
    int i,n,res,x;
    int a[10];

    printf("Input n:");
    scanf("%d",&n);
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
    printf("Input x:");
    scanf("%d",&x);
    res=search(a,n,x);
    if(res>=0)
        printf("index=%d\n",res);
    else
        printf("Not found\n");

    return 0;
}
int search(int list[],int n,int x)
{
    int i;

    for(i=0;i<n;i++)
        if(list[i]==x) return i;
    return -1;
}

```

3. 循环后移：有 n 个整数，使前面各数顺序向后移 m 个位置，移出的数再从开头移入。编写一个函数实现以上功能，在主函数中输入 n 个整数并输出调整后的 n 个数。试编写相应程序。

解答：

```

#include<stdio.h>
void move(int *x,int n,int m);

```

```

int main(void)
{
    int i,m,n;
    int a[80];

    printf("Enter n:");
    scanf("%d",&n);
    printf("Enter m:");
    scanf("%d",&m);
    for(i=0;i<n;i++)
        scanf("%d", &a[i]);
    move(a,n,m);
    printf("After move: ");
    for(i=0;i<n;i++)
        printf("%d ",a[i]);

    return 0;
}
void move(int *p,int n,int m)
{
    int i,j,k=0,a[80];

    for(i=0;i<n;i++)
        if(i<n-m)
            a[i+m]=p[i];
        else
            a[k++]=p[i];
    for(i=0;i<n;i++)
        p[i]=a[i];
}

```

4. 报数：有 n 个人围成一圈，按顺序从 1 到 n 编好号。从第一个人开始报数，报到 m ($m < n$) 的人退出圈子，下一个人从 1 开始报数，报到 m 的人退出圈子。如此下去，直到留下最后一个人。编写程序，输入整数 n 和 m ，并按退出顺序输出退出圈子的人的编号。试编写相应程序。

解答：

```

#include<stdio.h>
int main(void)
{
    int count,i,j,m,n;
    int num[80]={0};

    printf("Input n:");
    scanf("%d",&n);
    printf("Input m:");

```

```

scanf("%d",&m);
i=j=count=0;
while(count<n-1){
    if(num[i]==0)
        j++;
    if(j%m==0&&j){
        count++;
        num[i]=-1;
        j=0;// 报数报到 m后 , j 重归 0 , 开始下次报数
    }
    i++;// 读取下一个元素
    i%=n;// 一次 n 个跟报完数 , 开始下一次 ( 除余 n 是因为 i 是下标 , 从 0
    开始 , n 是具体 数 // 从 1 开始
} //i 为 n 时下次 a[0] 报数)
for(i=0;i<n;i++)
    if(num[i]==0)
        printf("Last No is: %d\n",i+1);

return 0;
}

```

5. 使用函数实现字符串复制：输入一个字符串 `t` 和一个正整数 `m`, 将字符串中从第 `m` 个字符开始的全部字符复制到字符串 `s` 中, 再输出字符串 `s`。要求自定义并调用函数 `void strmcpy(char *s, char *t, int m)`。试编写相应程序。

解答：

```

#include<stdio.h>
#include<string.h>
void strmcpy(char *s,char *t,int m);
int main(void)
{
    char s[80],t[80];
    int m;

    printf("Input the string:");
    gets(t);
    printf("Enter m:");
    scanf("%d",&m);
    if(strlen(t)<m){
        printf("Error Input");
        return 0;
    }
    else{
        strmcpy(s,t,m);
        puts(s);
    }
}

```

```

    }

    return 0;
}
void strmcpy(char *s,char *t,int m)
{
    t=t+m-1;
    while(*t!='\0'){
        *s=*t;
        s++;
        t++;
    }
    *s='\0';
}

```

6. 删除字符：输入一个字符串，再输入一个字符 ch，将字符串中所有的 ch 字符删除后输出该字符串。要求定义和调用函数 delchar(s,c)，该函数将字符串 s 中出现的所有 c 字符删除。试编写相应程序。

解答：

```

#include<stdio.h>
void delchar(char *s,char c);
int main(void)
{
    char c;
    char s[80];

    printf("Input the string:");
    gets(s);
    printf("Input a ch:");
    scanf("%c",&c);
    delchar(s,c);
    printf("result: ");
    puts(s);

    return 0;
}
void delchar(char *s,char c)
{
    int i,j;

    i=j=0;
    while(s[i]!='\0'){
        if(s[i]!=c){
            s[j]=s[i];
            j++;
        }
        i++;
    }
    s[j]='\0';
}

```

```

    }
    i++;
}
s[j]='\0';
}

```

7. 字符串排序：输入 5 个字符串，按由小到大的顺序输出。试编写相应程序。

解答：

```

#include<stdio.h>
#include<string.h>
int main(void)
{
    int n,i,j,index;
    char sx[80][80],stemp[80];

    printf("Enter n:");
    scanf("%d",&n);
    printf("Input %d strings:",n);
    for(i=0;i<n;i++)
        scanf("%s",sx[i]);//          每行的基地址
    for(i=0;i<n-1;i++){
        index=i;
        for(j=i+1;j<n;j++)
            if(strcmp(sx[j],sx[index])<0)
                index=j;
        strcpy(stemp,sx[i]);
        strcpy(sx[i],sx[index]);
        strcpy(sx[index],stemp);
    }
    printf("after sorted: \n");
    for(i=0;i<n;i++)
        printf("%s\t",sx[i]);

    return 0;
}

```

8. 判断回文：判断输入的一串字符是否为“回文”。所谓“回文”是指顺读和倒读都一样的字符串。如“XYZYX”和“xyzzyx”都是回文。试编写相应程序。

解答：

```

#include <stdio.h>
int mirror(char *p);
int main(void)
{
    char s[80];

    printf("Input the string:");

```

```

    gets(s);
    if(mirror(s)==1)
        printf("Yes!\n");
    else
        printf("No!\n");

    return 0;
}
int mirror(char *p)
{
    char *q;

    q=p;
    while(*q!='\0')
        q++;
    q--;
    while(p<q){
        if(*p!=*q)
            return 0;
        p++;
        q--;
    }
    return 1;
}
或
#include <stdio.h>
int mirror(char *p);
int main(void)
{
    char s[80];

    printf("Input the string:");
    gets(s);
    if(mirror(s)==1)
        printf("Yes!\n");
    else
        printf("No!\n");

    return 0;
}
int mirror(char *p)
{
    char *q;

```

```

    q=p;
    while(*q!='\0')
        q++;
    while(p<=q){
        if(*p==*(q-1)){
            return 1;
            p++;
            q--;
        }
        else
            return 0;
    }
}

```

9. 分类统计字符个数： 输入一行文字， 统计其中的大写字母、 小写字母、 空格、 数字以及其他字符各有多少。 试编写相应程序。

解答：

```

#include<stdio.h>
int main(void)
{
    char s[80],*p;
    int blank,digit,lower,other,upper;

    printf("Input the string:");
    gets(s);
    upper=lower=blank=digit=other=0;
    for(p=s;*p!='\0';p++)
        if(*p>='A'&&*p<='Z')
            upper++;
        else if(*p>='a'&&*p<='z')
            lower++;
        else if(*p>='0'&&*p<='9')
            digit++;
        else if(*p==' ')
            blank++;
        else
            other++;

    printf("upper: %d\nlower: %d\nblank: %d\ndigit: %d\nother: %d\n",upper,lower,blank,digit,other);

    return 0;
}

```

10. (选做) 输出学生成绩 (动态分布)： 输入学生人数后输入每个学生的成绩， 最后输出学生的平均成绩、 最高成绩和最低成绩。 要求使用动态内存分配来实现。

试编写相应程序。

解答：

```
#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    int n,i;
    float *p,sum,max,min,avg;

    printf("Input students      ' number n: ");
    scanf("%d",&n);
    if((p=(float *)calloc(n,sizeof(float)))==NULL){
        printf("Not able to allocate memory. \n");
        exit(1);
    }
    sum=0.0;
    max=-1;
    min=1000;
    printf("Input %d students      ' scores: ", n);
    for(i=0;i<n;i++){
        scanf("%f",p+i);
        sum=sum+*(p+i);
        if(min>*(p+i))
            min=*(p+i);
        if(max<*(p+i))
            max=*(p+i);
    }
    avg=sum/n;
    printf("The avg is %f,max is %f,min is %f\n",avg,max,min);
    free(p);

    return 0;
}
```


第 9 章 结构

【练习 9-1】定义一个能够表示复数的结构类型，一个复数包括实数与虚数两个部分。

解答：

```
struct complex{
    float real;
    float imaginary;
};
```

【练习 9-2】人的出生日期由年、月、日组成，请在例 9-1 中的通讯录结构中增加一个成员：出生日期，用嵌套定义的方式重新定义该结构类型。

解答：

```
struct date{
    int year;
    int month;
    int day;
};
struct student{
    int num;
    char name[10];
    struct date birthday;
    int computer,english,math;
    double average;
};
```

【练习 9-3】例 9-1 中，如果要计算的是三门课程的课程平均成绩，应该如何改写程序？

解答：

```
#include<stdio.h>
struct student{
    int num;
    char name[10];
    int computer,english,math;
    double average;
};
int main(void)
{
    int i, n;
    double math_sum,english_sum,computer_sum;
    struct student s1;

    printf("Input n:");
    scanf("%d", &n);
```

```

printf("Input the student's number, name and course scores:\n");
math_sum=english_sum=computer_sum=0;
for(i=1;i<=n;i++){
    printf("No.:%d:",i);
    scanf("%d%s%d%d%d",&s1.num,s1.name,&s1.math,&s1.english,&s1.compu
ter);
    math_sum+=s1.math;
    english_sum+=s1.english;
    computer_sum+=s1.computer;
}

printf("math_ave:%.2lf\nenglish_ave:%.2lf\ncomputer_ave:%.2lf\n",
math_sum/n,english_sum/n,computer_sum/n);

return 0;
}

```

【练习 9-4】定义一个包含 5 名学生信息的结构数组，并对该结构数组的所有元素进行初始化。

解答：

```

struct student{
    int num;
    char name[10];
    int computer, english, math;
};
struct student s[5]={30101, "    张一",69,75,84 },{30132, "    李二",80,8
5,90},{40231, "    王三",71,91,74},{40754, "    赵四",65,76,87},{50426, "    刘五
",81,92,73}};

```

【练习 9-5】参考例 9-2，输入并保存 10 个学生的成绩信息，分别输出平均成绩最高和最低的学生信息。

解答：

```

#include<stdio.h>
struct student{
    int num;
    char name[10];
    int computer,english,math;
    double average;
};
int main(void)
{
    int i,n,max,min;
    struct student students[50];

    printf("Input n:");
    scanf("%d",&n);

```

```

for(i=0;i<n;i++){
    printf("Input the info of No.:%d:\n",i+1);
    printf("number:");
    scanf("%d",&students[i].num);
    printf("name:");
    scanf("%s",&students[i].name);
    printf("math score:");
    scanf("%d",&students[i].math);
    printf("english score:");
    scanf("%d",&students[i].english);
    printf("computer score:");
    scanf("%d",&students[i].computer);
    students[i].average=(students[i].math+students[i].english+stu
dents[i].computer)/3.0;
}
max=min=0;
for(i=1;i<n;i++){
    if(students[max].average<students[i].average) max=i;
    if(students[min].average>students[i].average) min=i;
}
printf("max_average_infortmation:\n");
printf("number:%d,name:%s,math:%d,english:%d,computer:%d,average:
%.2lf\n",students[max].num,students[max].name,students[max].math,stud
ents[max].english,students[max].computer,students[max].average);
printf("min_average_information:\n");
printf("number:%d,name:%s,math:%d,english:%d,computer:%d,average:
%.2lf\n",students[min].num,students[min].name,students[min].math,stud
ents[min].english, students[min].computer, students[min].average);

return 0;
}

```

【练习 9-6】定义一个 struct student 类型的结构指针，用其实现一个学生信息的输入和输出。

解答：

```

struct student{          /*          学生信息结构定义  */
    int num;              /*          学号  */
    char name[10];        /*          姓名  */
    int computer, english, math ; /*          三门课程成绩  */
    double average;       /*          个人平均成绩  */
}s, *p;
p = &s;
scanf("%d%s%d%d%d", &p->num, &p->name, &p->math, &p->english,
&p->computer);

```

【练习 9-7】改写例 9-3 中的函数 update_score()，将第一个形参改为结构

数组形式。

解答：

```
int update_score(struct student s[],int n,int num,int course,int score)
{
    int i,pos;

    for(i=0;i<n;i++)
        if(s[i].num==num)
            break;
    if(i<n){
        switch(course){
            case 1:s[i].math=score; break;
            case 2:s[i].english=score; break;
            case 3:s[i].computer=score; break;
        }
        pos = i;
    }
    else
        pos = -1;

    return pos;
}
```

习题 9

一、选择题

1．下面定义结构变量的语句中错误的是 D 。

- A . struct student{ int num; char name[20]; } s;
- B . struct { int num; char name[20]; } s;
- C . struct student{ int num; char name[20]; }; struct student s;
- D . struct student{ int num; char name[20]; }; student s;

2．如果有定义语句：

```
struct {
    int x, y; } s[2] = { { 1, 3 }, { 2, 7 } };
```

则语句： printf(“ %d n” , s[0].y/s[1].x); 输出结果为 B 。

- A . 0 B . 1 C . 2 D . 3

3．根据下面的定义，能打印出字母 M 的语句是 C 。

```
struct person{
    char name[10];
    int age;
} c[10] = { “ John ” , 17, “ Paul ” , 19, “ Mary ” , 18, “ Adam” , 16 };
```

A . printf(" %c" , c[3].name); B . printf(" %c" , c[3].name[1]);
C . printf(" %c" , c[2].name[0]); D . printf(" %c" , c[2].name[1]);

4 . 设有如下定义 , 则对 data 中的 a 成员的正确引用是 B 。

struct sk{ int a; float b; } data, *p=&data;

A . (*p).data.a B . (*p).a
C . p->data.a D . p.data.a

5 . 对于以下结构定义 , (*p)->str++ 中的 ++加在 D 。

struct { int len; char *str; } *p;

A . 指针 str 上 B . 指针 p 上
C . str 指向的内容上 D . 语法错误

二、填空题

1 . “ . ” 称为 成员_(分量)_运算符 , “ -> ” 称为 _指向_运算符。

2 . 完成下列程序 , 该程序计算 10 名学生的平均成绩。

```
#include <stdio.h>
#include <string.h>
struct student {
    int num;
    char name[20];
    int score;
};
struct student stud[10];
int main(void)
{
    int i , sum = 0 ;
    for(i = 0; i < 10; i++){
        scanf("%d%s%d", & stud[i].num, _stud[i].name_, &
stud[i].score);
        sum += stud[i].score;
    }
    printf("aver = %d \n", sum/10);

    return 0;
}
```

3 . 下列程序读入时间数值 , 将其加 1 秒后输出 , 时间格式为 : hh: mm:ss , 即小时 : 分钟 : 秒 , 当小时等于 24 小时 , 置为 0。

```
#include<stdio.h>
struct {
    int hour, minute, second;
```

```

} time;
int main(void)
{
    scanf("%d: %d: %d", &_time.hour, &time.minute, &time.second_);
time.second++;
    if( _time.second_ == 60){
        _time.minute++ _;
        time.second = 0;
        if(time.minute == 60){
            time.hour++;
            time.minute = 0;
            if( _time.hour == 24_ )
                time.hour = 0;
        }
    }
    printf ("%d: %d: %d \n", time.hour, time.minute, time.second );

    return 0;
}

```

4 . 写出下面程序的运行结果 1 2 A B 。

```

struct s1{
    char c1, c2;
    int n;
};
struct s2{
    int n;
    struct s1 m;
} m = {1, { ' A ' , ' B ' , 2 } };
int main(void)
{
    printf( " %d                      n " , m.n, m.m.n, m.m.c1, m.m.c2);
    return 0;
}

```

5 . 写出下面程序的运行结果 23, wang, 98.5, wang 。

```

struct abc{
    int a;
    float b;
    char *c; };
int main(void)
{
    struct abc x = {23,98.5,"wang"};
    struct abc *px = &x;
    printf("%d, %s, %.1f, %s \n", x.a, x.c, (*px).b, px->c );
}

```

```
    return 0;
}
```

三、程序设计题

1. 时间换算：用结构类型表示时间内容（时间以时、分、秒表示），输入一个时间数值，再输入一个秒数 $n(n < 60)$ ，以 $h:m:s$ 的格式输出该时间再过 n 秒后的时间值（超过 24 点就从 0 开始计时）。试编写相应程序。

解答：

```
#include<stdio.h>
struct time{
    int hour;
    int minute;
    int second;
};
int main(void)
{
    int n;
    struct time tim;

    printf("Enter time:");
    scanf("%d%d%d",&tim.hour,&tim.minute,&tim.second);
    printf("Enter seconds:");
    scanf("%d",&n);
    tim.hour=(tim.hour*60*60+tim.minute*60+tim.second+n)/3600%24;
    tim.minute=(tim.minute*60+tim.second+n)/60%60;
    tim.second=(n+tim.second)%60;
    printf("%d:%d:%d",tim.hour,tim.minute,tim.second);

    return 0;
}
```

2. 计算两个复数之积：编写程序，利用结构变量求解两个复数之积： $(3+4i) \times (5+6i)$ 。

提示：求解 $(a_1+a_2i) \times (b_1+b_2i)$ ，乘积的实部为： $a_1 \times b_1 - a_2 \times b_2$ ，虚部为： $a_1 \times b_2 + a_2 \times b_1$ 。

解答：

```
#include<stdio.h>
struct complex{
    int real,im;
};
struct complex cmult(struct complex a,struct complex b);
int main(void)
{
    struct complex a={3,4},b={5,6},c;

    c=cmult(a,b);
}
```

```

        printf("(%d+%di)(%d+%di)=%d+%di\n",a.real,a.im,b.real,b.im,c.real,c.im);

    return 0;
}
struct complex cmult(struct complex a,struct complex b)
{
    struct complex c;

    c.real=a.real*b.real-a.im*b.im;
    c.im=a.real*b.im+a.im*b.real;

    return c;
}

```

3. 平面向量加法：输入两个二维平面向量 $V1=(x_1, y_1)$ 和 $V2=(x_2, y_2)$ 的分量，计算并输出两个向量的和向量。试编写相应程序。

解答：

```

#include<stdio.h>
struct vector{
    float x,y;
};
struct vector vec(struct vector v1,struct vector v2);
int main(void)
{
    struct vector v1,v2,v;

    printf("x1 and y1:");
    scanf("%f%f",&v1.x,&v1.y);
    printf("x2 and y2:");
    scanf("%f%f",&v2.x,&v2.y);
    v=vec(v1,v2);
    printf("v=(%f,%f)",v.x,v.y);

    return 0;
}
struct vector vec(struct vector v1,struct vector v2)
{
    struct vector v;

    v.x=v1.x+v2.x;
    v.y=v1.y+v2.y;

    return v;
}

```



```
}
```

4. 查找书籍：从键盘输入 10 本书的名称和定价并存入结构数组中，从中查找定价最高和最低的书的名称和定价，并输出。试编写相应程序。

解答：

```
#include<stdio.h>
#define NUMBER 10
struct book{
    char name[10];
    float price;
};
int main(void)
{
    int i,maxl,minl,n;
    struct book test[NUMBER];

    for(i=0;i<NUMBER;i++)
        scanf("%s%f",test[i].name,&test[i].price);
    maxl=minl=0;
    for(i=1;i<n;i++){
        if(test[maxl].price<test[i].price) maxl=i;
        if(test[minl].price>test[i].price) minl=i;
    }
    printf("Max Price:%f,%s\n", test[maxl].price,test[maxl].name);
    printf("Min Price:%f,%s\n", test[minl].price,test[minl].name);
    return 0;
}
```

5. 通讯录排序：建立一个通讯录，通讯录的结构包括：姓名、生日、电话号码；其中生日又包括三项：年、月、日。编写程序，定一个嵌套的结构类型，输入n(n ≤ 10)个联系人信息，再按照他们的年龄从大到小依次输出其信息。试编写相应程序。

解答：

```
#include<stdio.h>
struct date{
    int year;
    int month;
    int day;
};
struct friends_list{
    char name[10];
    struct date birthday;
    char phone[15];
};
void sort(struct friends_list s[],int n);
int main(void)
```

```

{
    int i,n;
    struct friends_list friends[10];

    printf("Input n:");
    scanf("%d",&n);
    printf("Input %d friends' name ,birthday and phone:",n);
    for(i=0;i<n;i++)
        scanf("%s%d%d%d%s",friends[i].name,&friends[i].birthday.year,
&friends[i].birthday.month,&friends[i].birthday.day,&friends[i].phone
);
    sort(friends,n);
    for(i=0;i<n;i++)
        printf("%s%d/%d/%d%s\n",friends[i].name,friends[i].birthday.y
ear,friends[i].birthday.month,friends[i].birthday.day,friends[i].phon
e);

    return 0;
}
void sort(struct friends_list s[],int n)
{
    int i,j,index;
    struct friends_list temp;

    for(i=0;i<n-1;i++){
        index=i;
        for(j=i+1;j<n;j++){
            if(s[index].birthday.year>s[j].birthday.year)
                index=j;
            else if((s[index].birthday.year==s[j].birthday.year)
&&(s[index].birthday.month>s[j].birthday.month))
                index=j;
            else if((s[index].birthday.year==s[j].birthday.year)
&&(s[index].birthday.month==s[j].birthday.month)&&(s[index].birthday.
day>s[j].birthday.day)){
                index=j;
                temp=s[i];
                s[i]=s[index];
                s[index]=temp;
            }
        }
    }
}

```

6．按等级统计学生成绩：输入 10 个学生的学号、姓名和成绩，输出学生的成绩等级和不及格人数。每个学生的记录包括学号、姓名、成绩和等级，要求定义

和调用函数 set_grade() 根据学生成绩设置其等级，并统计不及格人数，等级设置：85 - 100 为 A，70 - 84 为 B，60 - 69 为 C，0 - 59 为 D。试编写相应程序。

解答：

```
#include<stdio.h>
#define N 10
struct students{
    int num;
    char name[20];
    int score;
    char grade;
};
int set_grade(struct students *p);
int main(void)
{
    struct students student[N];
    int i,count;

    printf("Input 10 students' number,name and score: \n");
    for(i=0;i<N;i++){
        printf("No.%d:",i+1);
        scanf("%d%s%d",&student[i].num,student[i].name,&student[i].score);
    }
    count=set_grade(student);
    printf("The count (<60): %d\n",count);
    printf("The student grade:\n");
    for(i=0;i<N;i++)
        printf("%d %s %c\n",student[i].num,student[i].name,student[i].grade);

    return 0;
}
int set_grade(struct students *p)
{
    int i,n=0;

    for(i=0;i<N;i++,p++){
        if(p->score>=85)
            p->grade='A';
        else if(p->score>=70&& p->score<85)
            p->grade='B';
        else if(p->score>=60&& p->score<70)
            p->grade='C';
        else{
```

```
        p->grade='D';  
        n++;  
    }  
}  
  
return n;  
}
```

第 10 章 函数与程序结构

【练习 10-1】使用递归函数计算 1 到 n 之和：若要用递归函数计算 $sum=1+2+3+\dots+n$ (n 为正整数)，请写出该递归函数的递归式子及递归出口。试编写相应程序。

解答：

递归式子： $sum(i) = sum(i-1) + i$;

递归出口： $sum(i) = 0$;

【练习 10-2】请完成下列宏定义：

MIN(a,b)	求 a,b 的最小值
ISLOWER(c)	判断 c 是否为小写字母
ISLEAP(y)	判断 y 是否为闰年
CIRFER(r)	计算半径为 r 的圆周长

解答：

MIN(a, b) : 求 a,b 的最小值。

```
#define MIN(a,b) (a<b)? a: b
```

ISLOWER(c) : 判断 c 是否为小写字母。

```
#define ISLOWER(c) (c>='a'&&c<='z')
```

ISLEAP(y) : 判断 y 是否为闰年。

```
#define ISLEAP(y) (y%4==0&&((y%100!=0)||y%400==0))
```

CIRFER(r) : 计算半径为 r 的圆周长。

```
#define PI 3.14159
```

```
#define CIRFER(r) 2*PI*(r)
```

【练习 10-3】分别用函数和带参宏实现从 3 个数中找出最大数，请比较两者在形式上和使用上的区别。

解答：

(1) 函数实现

```
int max(int x,int y,int z)
```

```
{
    int t;

    if(x>=y)
    if(x>=z) t=x;
    else t=z;
    else
        if(y>=z) t=y;
    else t=z;

    return t;
}
```

(2) 宏实现

```
#define MAX( x, y, z ) x>=y? (x>=z? x:z) : (y>=z? y:z)
```

两者在定义形

式上完全不同。使用上函数是在执行时，从主调函数转到函数 `max()`，然后再返回到主调函数，函数体始终存在；而宏是在编译预处理时，用条件表达式去替换 `MAX(int x, int y, int z)`，等程序执行时，执行的是条件表达式，而不再存在 `MAX(int x, int y, int z)` 的式子。

习题 10

一、选择题

1. 要调用数学函数时，在 `#include` 命令行中应包含 C 。

- A. " `stdio.h` " B. " `string.h` " C. " `math.h` "
D. " `ctype.h` "

2. 对于以下递归函数 `f`，调用 `f(4)`，其返回值为 A 。

```
int f(int n)
{
    if (n) return f(n - 1) + n;
    else return n;
}
```

- A. 10 B. 4 C. 0 D. 以上均不是

3. 执行下列程序：

```
#define MA(x, y) (x*y)
i = 5;
i = MA(i, i + 1) - 7;
```

后变量 `i` 的值应为 B 。

- A. 30 B. 19 C. 23
D. 1

4. 宏定义 " `#define DIV(a, b) a/b` "，经 `DIV(x + 5, y - 5)` 引用，替换展开后是 A 。

- A. `x + 5 / y - 5` B. `(x + 5 / y - 5)`
C. `(x + 5) / (y - 5)` D. `(x + 5) / (y - 5);`

5. 定义带参数的宏 " `#define JH(a,b,t) t = a; a = b; b = t` "，对两个参数 `a`、`b` 的值进行交换，下列表述中哪个是正确的 C 。

- A. 不定义参数 `a` 和 `b` 将导致编译错误 B. 不定义参数 `a`、`b`、`t` 将导致编译错误
C. 不定义参数 `t` 将导致运行错误 D. 不需要定义参数 `a`、`b`、`t` 类型

6. 执行下面程序，正确的输出是 A 。

```

int x = 5, y = 7;
void swap ( )
{   int z ;
    z = x ; x = y ; y = z ;
}
int main(void)
{
    int x = 3, y = 8;
    swap ( ) ;
    printf ( " %d , %d \n", x , y ) ;
    return 0 ;
}

```

A . 3, 8 B . 8, 3 C . 5, 7 D . 7, 5

7 . 下面说法中正确的是 A 。

- A .若全局变量仅在单个 C 文件中访问，则可以将这个变量修改为静态全局变量，以降低模块间的耦合度
- B . 若全局变量仅由单个函数访问，则可以将这个变量改为该函数的静态局部变量，以降低模块间的耦合度
- C . 设计和访问动态全局变量、静态全局变量、静态局部变量的函数时，需要考虑变量生命周期问题
- D . 静态全局变量使用过多，可那会导致动态存储区（堆栈）溢出

二、填空题

1.C 语言的编译预处理功能主要包括 _条件编译_、_宏定义_和_文件包含_。

2 . 执行完下列语句段后，i 的值为 5 。

```

int i;
int f(int x)
{
    return ((x>0)? f(x-1)+f(x-2):1);
}
i=f(3);

```

3 . 下列程序段 A 与 B 功能等价，请填写程序段 B 中相应语句。

程序段 A：

```

int f( int n )
{   if(n<=1)
    return n;
    else
    return f(n-1)+f(n-2);
}

```

程序 B：

```

int f( int n )

```

```

{   _int t, t0, t1;_
    t0=0; t1=1; t=n;
    while ( _n>1_){
        t = _t0+t1_;
        t0 = t1;
        t1 = t;
        n - -;
    }
    return _t;_
}

```

5 . 下面程序用于计算 $f(k, n)=1^k+2^k+\dots+n^k$, 其中 $\text{power}(m, n)$ 求 m^n 。请填写程序中相应语句。

```

#include <stdio.h>
int power(int m , int n)
{   int i ;
    int p=1;
    for( i=1 ; i<=n ; i++)
        p = p * m ;
    return p ;
}
int f(int k , int n){
    int i ;
    int s=0 ;
    for( i=1 ; i<=n ; i++)
        s = s + power(i, k) ;
    return s ;
}
int main(void)
{   int k , n ;
    scanf( " %d%d", &k, &n ) ;
    printf( " f(%d, %d)=%ld " , k, n, f(k, n)) ;
    return 0;
}

```

5 . 下列递归程序的输出结果为 $g=4, g=3, k=6$ 。

```

#include <stdio.h>
int fib(int g)
{   switch(g){
        case 0: return 0;
        case 1:
        case 2: return 2;
    }
    printf("g=%d,", g);
}

```



```

        return fib(g-1) + fib(g-2);
    }

```

```

int main(void)
{
    int k;
    k = fib(4);
    printf("k=%ld\n", k);
    return 0;
}

```

6 . 下列语句的运算结果为 7,-13 。

```

#define F(x)  x - 2
#define D(x)  x * F(x)
printf("%d,%d", D(3), D(D(3))) ;

```

三、程序设计题

1 . 判断满足条件的三位数：编写一个函数，利用参数传入一个 3 位数 n，找出 101 ~ n 间所有满足下列两个条件的数：它是完全平方数，又有两位数字相同，如 144、676 等，函数返回找出这样的数据的个数。试编写相应程序。

解答：

```

#include<stdio.h>
#include<math.h>
int fun(int n);
int main(void)
{
    int n;

    printf("Input n: ");
    scanf("%d",&n);
    printf("total=%d\n",fun(n));

    return 0;
}
int fun(int n)
{
    int i,d=0;

    for(i=101;i<=n;i++)
        if(((int)sqrt(i)*(int)sqrt(i))==i){
            if(i/100==(i/10)%10||i/100==i%10||(i/10)%10==i%10)
                d++;
        }

    return d;
}

```

2. 递归求阶乘和：输入一个整数 n ($n > 0$ 且 $n \leq 10$)，求 $1! + 2! + 3! + \dots + n!$ 。定义并调用函数 $\text{fact}(n)$ 计算 $n!$ ，函数类型是 double 。试编写相应程序。

解答：

```
#include<stdio.h>
double fact(int n);
int main(void)
{
    int n;

    printf("Input n:");
    scanf("%d",&n);
    printf("1~%d 的阶乘和 :%.0lf",n,fact(n));

    return 0;
}
double fact(int n)
{
    int i;
    double result=1;

    if(n==1)
        result=1;
    else{
        for(i=1;i<=n;i++)
            result=result*i;
        result=result+fact(n-1);
    }

    return result;
}
```

3. 递归实现计算 x^n ：输入实数 x 和正整数 n ，用递归函数计算 x^n 的值。试编写相应程序。

解答：

```
#include<stdio.h>
double power(double x,int n);
int main(void)
{
    double x;
    int n;

    printf("Input x:");
    scanf("%lf",&x);
    printf("Input n:");
```

```

scanf("%d",&n);
printf("%lf    的%d次方=%lf",x,n,power(x,n));

return 0;
}
double power(double x,int n)
{
    double result;

    if(n==1)
        result=x;
    else
        result=x*power(x,(n-1));

    return result;
}

```

4 . 递归求式子和：输入实数 x 和正整数 n , 用递归的方法对下列计算式子编写一个函数。

$$f(x, n) = x - x^2 + x^3 - x^4 + \dots + (-1)^{n-1} x^n \quad (n > 0)$$

试编写相应程序。

解答：

```

#include<stdio.h>
#include<math.h>
double f(double x,int n);
int main(void)
{
    int n;
    double x;

    printf("Input x:");
    scanf("%lf",&x);
    printf("Input n:");
    scanf("%d",&n);
    printf("f(%lf,%d)=%lf\n",x,n,f(x,n));

    return 0;
}
double f(double x, int n)
{
    if(n==1)
        return x;
    else
        return pow(-1,n-1)*pow(x,n)+f(x,n-1);
}

```

}

5. 递归计算函数 $\text{ack}(m, n)$: 输入 m 和 n 。编写递归函数计算 Ackermann 函数的值:

$$\text{ack}(m, n) = \begin{cases} n+1 & m=0 \\ \text{ack}(m-1, 1) & n=0 \ \& \ m>0 \\ \text{ack}(m-1, \text{ack}(m, n-1)) & m>0 \ \& \ n>0 \end{cases}$$

试编写相应程序。

解答:

```
#include<stdio.h>
int Ack(int m,int n);
int main(void)
{
    int m,n;
    int result;

    scanf("%d%d", &m, &n);
    result=Ack(m,n);
    printf("Ackerman(%d,%d)=%d\n",m,n,result);

    return 0;
}
int Ack(int m, int n)
{
    if (m==0) return n+1;
    else if(n==0&&m>0)
        return Ack(m-1,1);
    else if(m>0&&n>0)
        return Ack(m-1,Ack(m,n-1));
}
```

6. 递归实现求 Fibonacci 数列: 用递归方法编写求斐波那契数列的函数, 函数类型为整型, 并写出相应主函数。斐波那契数列的定义如下。试编写相应程序。

$$f(n) = f(n-2) + f(n-1) \quad (n > 1) \text{ 其中 } f(0) = 0, f(1) = 1。$$

解答:

```
#include<stdio.h>
long fib(int n);
int main(void)
{
    int n;

    printf("Enter n:");
    scanf("%d",&n);
    printf("fib(%d)=%ld\n",n,fib(n));

    return 0;
}
```

```

}
long fib(int n)
{
    long res;

    if(n==0) res=0;
    else if(n==1)
        res=1;
    else
        res=fib(n-2)+fib(n-1);

    return res;
}

```

7. 递归实现十进制转换二进制： 输入一个正整数 n ，将其转换为二进制输出。 要求定义并调用函数 `dectobini(n)`， 它的功能是输出 n 的二进制。试编写相应程序。

解答：

```

#include<stdio.h>
int dectobini(int n);
int main(void)
{
    int n;

    printf("Input n(    十进制整数 ):");
    scanf("%d",&n);
    printf("    输出二进制 :%d",dectobini(n));

    return 0;
}
int dectobini(int n)
{
    int result;
    if(n==0)
        result=0;
    else
        result=dectobini(n/2)*10+n%2;
    return result;
}

```

8. 递归实现顺序输出整数：输入一个正整数 n ，编写递归函数实现对其进行按位顺序输出的递归函数。试编写相应程序。

解答：

```

#include <stdio.h>
int inorder(int n);
int main(void)
{

```

```

int n;

printf("Input n:");
scanf("%d",&n);
printf("    按位顺序输出： %d\n",inorder(n));

return 0;
}
int inorder(int n)
{
    if(n<10) return n;
    else
        return(inorder(n/10)*10+n%10);
}

```

9. 输入 n (n<10) 个整数，统计其中素数的个数。要求程序有两个文件组成，一个文件中编写 main 函数，另一个文件中编写素数判断的函数。使用文件包含的方式实现。试编写相应程序。

解答：

10 . 三角形面积为：

$$\text{area} = \sqrt{s \times (s-a) \times (s-b) \times (s-c)} \quad s=(a+b+c)/2$$

其中 a、b、c 分别是三角形的 3 条边。请分别定义计算 s 和 area 的宏。再使用函数实现，比较两者在形式上和使用上的区别。

解答：

(1) 使用宏实现

```

#include <stdio.h>
#include <math.h>
#define S(a,b,c) ((a)+(b)+(c))/2
#define AREA(s,a,b,c) sqrt((s)*((s)-(a))*((s)-(b))*((s)-(c)))
int main(void)
{
    double a, b, c, s;
    printf("input a, b, c: ");
    scanf("%lf%lf%lf", &a, &b, &c);
    s = S(a,b,c);
    printf("s = %lf, area = %lf\n", s, AREA(s,a,b,c));
    return 0;
}

```

(2) 使用函数实现

```

#include <stdio.h>
#include <math.h>
double f1(double a, double b, double c)

```

```
{
    return (a+b+c)/2;
}
double f2(double s, double a, double b, double c)
{
    return sqrt(s*(s-a)*(s-b)*(s-c));
}
int main(void)
{
    double a, b, c, s;
    printf("input a, b, c: ");
    scanf("%lf%lf%lf", &a, &b, &c);
    s = f1(a,b,c);
    printf("s = %lf, area = %lf\n", s, f2(s,a,b,c));
    return 0;
}
```

第 11 章 指针进阶

【练习 11-1】如何理解指针数组，它与指针、数组有何关系？为何可以用二级指针数组进行操作？

解答：

指针数组——存放指针的一个数组。

指针数组名也是指向常量二级指针，因为指针数组名总是指向数组的第一个元素。

【练习 11-2】用指针数组处理多个字符串有何优势？可以直接输入多个字符串给为初始化的指针数组吗？为什么？

解答：

C语言中字符串的地址是按顺序存放的，用指针指向字符串的首地址，然后取出这个地址里面的值，然后指针加一，取出里面的值，这样就能输出整个字符串了。用指针数组可以接收多个字符串，对这些字符串的操作可以做到相互隔离，一个指针数组的元素作为一个指针对应一个字符串。

【练习 11-3】参考例 11-3，使用二级指针操作改写例 11-4 中的程序 A。

解答：

```
#include<stdio.h>
#include<string.h>
void fsort(char **ch,int n);
int main(void)
{
    int i;
    char *pcolor[]={"red","blue","yellow","green","black"};
    char **ch;
    ch=pcolor;
    fsort(ch,5);
    for(i=0;i<5;i++)
        printf("%s ",*(ch+i));
    return 0;
}
void fsort(char **ch,int n)
{
    int k,j;
    char *temp;
    for(k=1;k<n;k++)
        for(j=0;j<n-k;j++)
            if(strcmp(*(ch+j),*(ch+j+1))>0){
                temp=*(ch+j);
                *(ch+j)=*(ch+j+1);
```



```

        *(ch+j+1)=temp;
    }
}

```

【练习 11-4】改写例 11-8 中的函数 match(), 要求返回字符串 s 中最后一个字符 ch 的位置 (地址)。

解答：

```

#include<stdio.h>
char *match(char *s,char ch);
int main (void)
{
    char ch,str[80],*p=NULL;

    printf("Please Input the string:\n");
    scanf("%s",str);
    getchar();
    ch=getchar();
    if((p=match(str,ch))!=NULL)
        printf("%s\n",p);
    else
        printf("Not Found\n");
    return 0;
}
char *match(char *s,char ch)
{
    char *k=NULL;
    while(*s!='\0'){
        if(*s==ch)
            k=s;
        s++;
    }
    return k;
}

```

【练习 11-5】前面章节中介绍的指针变量都可以进行算数运算，请思考：指向函数的指针变量可以进行算术运算吗？

解答：

函数指针变量不能进行算术运算，这是与数组指针变量不同的。数组指针变量加减一个整数可使指针移动指向后面或前面的数组元素，而函数指针的移动是毫无意义的。

【练习 11-6】运行例 11-10，试执行程序中间函数的功能，观察结果。

解答：

主函数包含选择链表新建、插入、删除和遍历的功能，而新建链表函数则是

用链表的方式建立一整套有关学生的信息，插入操作函数则是可以在链表其中添加新的学生信息，删除操作函数则是可以在链表中删除要求删除的学生信息，最后的遍历操作函数则是显示所有的学生信息。

【练习 11-7】改写例 11-10 中的函数 DelectDoc ()，要求删除链表中成绩小于 60 分的学生结点。

解答：

```
struct stud_node* DeleteDoc(struct stud_node *head,int score)
{
    struct stud_node *ptr1,*ptr2;
    /* 要被删除结点为表头结点 */
    while(head!=NULL&&head->score==score){
        ptr2=head;
        head=head->next;
        free(ptr2);
    }
    if(head==NULL)
        return NULL;
    /* 要被删除结点为非表头结点 */
    ptr1=head;
    ptr2=head->next;
    while(ptr2!=NULL){
        if(ptr2->score==score){
            ptr1->next=ptr2->next;
            free(ptr2);
        }
        else
            ptr1=ptr2;
        ptr2=ptr1->next;
    }
    return head;
}
```

【练习 11-8】在例 11-10 的基础上，再编写一个函数 UpdateDoc ()，实现对链表中某结点信息（成绩）的修改。函数原型为：void UpdateDoc(struct stud_node *head,int num,int score)，其中，num为需要修改信息的学生学号，score 为需要修改的成绩值。

解答：

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
struct stud_node{
    int num;
    char name[20];
```

```

    int score;
    struct stud_node *next;
};
struct stud_node *Create_Stu_Doc();/*      新建链表 */
struct stud_node *InsertDoc(struct stud_node *head,struct stud_node
*stud);/*      插入结点 */
struct stud_node *DeleteDoc(struct stud_node *head,int num); /*      删除结
点 */
void Print_Stu_Doc(struct stud_node *head);/*      链表遍历 */
void UpdateDoc(struct stud_node *head,int num,int score);/*      修改成绩 */
int main (void)
{
    struct stud_node *head,*p;
    int choice,num,score;
    char name[20];
    int size=sizeof(struct stud_node);

    do{
        printf("1:Create 2:Insert 3>Delete 4:Update 5:Print 0:Exit\n");
        scanf("%d",&choice);
        switch(choice){
            case 1:
                head=Create_Stu_Doc();
                break;
            case 2:
                printf("Input num,name and score:\n");
                scanf("%d%s%d",&num,name,&score);
                p=(struct stud_node*)malloc(size);
                p->num=num;
                strcpy(p->name,name);
                p->score=score;
                head=InsertDoc(head,p);
                break;
            case 3:
                printf("Input num:\n");
                scanf("%d",&num);
                head=DeleteDoc(head,num);
                break;
            case 4:
                printf("Input num:\n");
                scanf("%d",&num);
                printf("Input score:\n");
                scanf("%d",&score);
                UpdateDoc(head,num,score);

```

```

        break;
    case 5:
        Print_Stu_Doc(head);
        break;
    case 0:
        break;
    }
    }while(choice!=0);

    return 0;
}
/* 新建链表 */
struct stud_node *Create_Stu_Doc()
{
    struct stud_node *head,*p;
    int num,score;
    char name [20] ;
    int size=sizeof(struct stud_node);

    head=NULL;
    printf("Input num,name and score: \n");
    scanf("%d%s%d",&num,name,&score);
    while(num!=0){
        p=(struct stud_node*)malloc(size);
        p->num=num;
        strcpy(p->name,name);
        p->score=score;
        head=InsertDoc(head,p);
        scanf("%d%s%d",&num,name,&score);
    }
    return head;
}
/* 插入操作 */
struct stud_node *InsertDoc(struct stud_node *head,struct stud_node
*stud)
{
    struct stud_node *ptr,*ptr1,*ptr2;

    ptr2=head;
    ptr=stud;
/* 原链表为空时的插入 */
    if(head==NULL){
        head=ptr;
        head->next=NULL;

```

```

    }
    else{
        while((ptr->num>ptr2->num)&&(ptr2->next!=NULL)){
            ptr1=ptr2;
            ptr2=ptr2->next;
        }
        if(ptr->num<=ptr2->num){
            if(head==ptr2) head=ptr;
            else ptr1->next=ptr;
            ptr->next=ptr2;
        }
        else{
            ptr2->next=ptr;
            ptr->next=NULL;
        }
    }
    return head;
}
/* 删除操作 */
struct stud_node* DeleteDoc(struct stud_node *head,int num)
{
    struct stud_node *ptr1,*ptr2;
    while(head!=NULL&&head->num==num){
        ptr2=head;
        head=head->next;
        free(ptr2);
    }
    if(head==NULL)
        return NULL;
    ptr1=head;
    ptr2=head->next;
    while(ptr2!=NULL){
        if(ptr2->num==num){
            ptr1->next=ptr2->next;
            free(ptr2);
        }
        else
            ptr1=ptr2;
        ptr2=ptr1->next;
    }
    return head;
}
/* 遍历操作 */
void Print_Stu_Doc(struct stud_node *head)

```

```

{
    struct stud_node*ptr;
    if(head==NULL){
        printf("\nNo Records\n");
        return;
    }
    printf("\nThe Students' Records Are: \n");
    printf("Num\tName\tScore\n");
    for(ptr=head;ptr!=NULL;ptr=ptr->next)
        printf("%d\t%s\t%d\n",ptr->num,ptr->name,ptr->score);
}
/* 修改操作 */
void UpdateDoc(struct stud_node *head,int num,int score)
{
    struct stud_node *ptr;
    if(head==NULL){
        printf("\nNo Records\n");
        return;
    }
    for(ptr=head;ptr!=NULL;ptr=ptr->next)
        if(ptr->num==num){
            ptr->score=score;
            break;
        }
    printf("\nThe Students' Records Are: \n");
    printf("Num\tName\tScore\n");
    for(ptr=head;ptr!=NULL;ptr=ptr->next)
        printf("%d\t%s\t%d\n",ptr->num,ptr->name,ptr->score);
}

```

习题 11

一、选择题

1. 下面程序的运行结果是 ____B____。

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

```
int main (void)
```

```

{
    int x[5]={2,4,6,8,10},*p,**pp;
    p=x;
    pp=&p;
    printf ("%d",*(p++));
    printf (    " %d\n",**pp);
    return 0;
}

```

A.4 4 B.2 4 C.2 2 D.4 6

2. 对于以下变量定义，正确的赋值是 ____D____。

```
int *p[3],a [3];
```

A.p=a B.* p=a [0] C.p=&a [0] D.p[0]=&a [0]

3. 下列程序的输出是 ____D____。

```
#include<stdio.h>
int main (void)
{
    int a[12]={1,2,3,4,5,6,7,8,9,10,11,12},*p[4],i;
    for(i=0;i<4;i++)
        p[i]=&a [i*3];
    printf ("%d\n",p[3] [2] );
    return 0;
}
```

A. 上述程序有错误 B.6

C.8 D.12

4. 下列程序的输出结果是 ____C____。

```
#include<stdio.h>
#include<string.h>
char * fun (char *t)
{   char *p=t;
    return (p+strlen(t) /2);
}
int main (void)
{   char *str=      " abcdefgh" ;
    str=fun (str);
    puts(str);
    return 0;
}
```

A. , abcdefgh B.abcd C efth D.fgh

5. 设有如下定义的链表，则值为 7 的表达式是 ____C____。

```
struct st {
    int n;
    struct st *next;
} a[3]={ 5,&a[1],7,&a[2].9,NULL},*p=&a;
```

A.p->n B.(p->n) ++ C.(++p)->n D.p->next->n

二、填空题

1. 下面程序可以逐行输出由 language 数组元素所指向的 5 个字符中。请填写程序中相应语句。

```
#include<stdio.h>
int main (void)
```

```

{
    char * language[ ]= {"BASIC",      " FORTRAN", " PROLOG", "JAVA ,C++ " };
    char ____*q____;
    int k
    for(k=0;k<5;k++){
        q=_language[k]_;
        printf ("%s \n",*q);
    }
    return 0;
}

```

2. 下面的程序输入 10 个成绩，查找最高分并输出。请填写程序中相应语句。

```

#include<stdio.h>
int *GetMax (int score[ ],int n);
int main (void)
{
    int i,score [10],*p;
    for (i=0;i<10; i++)
        scanf ( " %d",&score[ i] );
    p=_GetMax(score,10)_;
    printf ("Max:%d\ n",*p);
    return 0;
}
int *GetMax (int score[ ],int n)
{
    int i,temp,pos=0;
    temp=score [0] ;
    for(i=0;i<10;i++)
        if(score[i]>temp)
        { temp=score[i];
          pos=i;
        }
    return __&temp____;
}

```

3. 下面程序的运行结果是 1 0 0 1

```

    0 1 1 0
    0 1 1 0
    __ 1 0 0 1 __.

```

```

#include<stdio.h>
int main(void)
{
    static int a[4][4];
    int *p[4],i,j;
    for(i=0;i<4;i++)
        p[i]=&a[i][0];
}

```



```

for(i=0;i<4;i++){
*(p[i]+i)=1;
*(p[i]+4-(i+1))=1;
}
for(i=0;i<4;i++){
    for(j=0;j<4;j++)
        printf("%2d",p[i][j]);
    printf("\n");
}
return 0;
}

```

4. 下列函数用于将链表中各结点的数据依次输出。请填写程序中相应语句。

```

void print (struct student *head)
{
    struct student *p;
    p=head;
    if(head!= NULL)
        do{
            printf ("ld\      n " ,p->data);
            _p=p->next_____;
        }while(_p!=NULL_);
}

```

5. 已建立学生“英语”课程的成绩链表（成绩存于 score 域中，学号存于 num域中），下列函数用于输出不及格学生的学号和成绩，及补考学生人数。请填写程序中相应语句。

```

void require (struct student *head)
{
    struct student *p;
    long x;
    if( head!= NULL){
        ___head=p___;
        while (p!= NULL){
            if(_p->score<60____){
                printf ( " %7d %6.1lf \n",p ->num,p->score);
                x++;
            }
            p=p->next;
        }
        printf ( " %ld\n",x);
    }
}

```

三、程序设计题

1. 输出月份英文名：输入月份，输出对应的英文名称。要求用指针数组表示 12 个月的英文名称。例如，输入 5, 输出 May 试编写相应程序。

解答：

```
#include<stdio.h>
int main(void)
{
    char
*months[12]={"January","February","March","April","May","June","July"
,
"Aguest","September","October","November","December"};
    int month;

    printf("Input the month:");
    scanf("%d",&month);
    printf("Month in English:%s",months[month-1]);

    return 0;
}
```

2. 查找星期： 定义一个指针数组，将下表的星期信息组织起来，输入一个字符串，在表中查找，若存在，输出该字符串在表中的序号，否则输出 -1。试编写相应程序。

0	Sunday
1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday

解答：

```
#include<stdio.h>
#include<string.h>
int main()
{
    int i,ri,repeat;
    char
*date[]={"Sunday","Monday","Tuesday","Wednesday","Thursday","Friday",
"Saturday"},str[80];

    printf("Input the string:");
    scanf("%s",str);
    for(i=0;i<7;i++){
        if(strcmp(date[i],str)==0){
            printf("%d\n",i);
            break;
        }
        else
```

```

        printf("%d\n",-1);
    }

    return 0;
}

```

3. 计算最长的字符串长度：输入 n (n<10) 个字符串，输出其中最长字符串的有效长度。要求自定义函数 `int max_len (char *s[],int n)`，用于计算有 n 个元素的指针数组 s 中最长的字符串的长度。试编写相应程序。

解答：

```

#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int max_len(char *s[],int n);
int main(void)
{
    int n,i=0,j;
    char *s[10],str[10];

    printf("Enter n(n<10) :");
    scanf("%d",&n);
    printf("Input %d string:\n",n);
    while(i<n){
        scanf("%s",str);
        s[i]=(char*)calloc(sizeof(char),strlen(str)+1);
        strcpy(s[i],str);
        i++;
    };
    printf("length=%d",max_len(s,n));

    return 0;
}
int max_len(char *s[10],int n)
{
    int i,max,k;

    max=0;
    for(i=0;i<n;i++){
        k=strlen(s[i]);
        max=(max>k)?max:k;
    }
    return max;
}

```

4. 字符串的连接：输入两个字符串，输出连接后的字符串。要求自定义函数 `char strcat (char*s,char*t)`，将字符串 t 复制到字符串 s 的末端，并且返回字符串

s 的地址。试编写相应程序。

解答：

```
#include<stdio.h>
char *strcat(char *s,char *t);
int main(void)
{
    int n,i=0,j;
    char s[80],t[80];

    printf("Input first string:");
    gets(s);
    printf("Input second string:");
    gets(t);
    printf("%s",strcat(s,t));

    return 0;
}
char *strcat(char *s,char *t)
{
    int i,j=0;

    for(i=sizeof(s)/2;i<=sizeof(s)/2+sizeof(t)/2;i++,j++)
        *(s+i)=*(t+j);
    *(s+i-1)='\0';
    return s;
}
```

5. 指定位置输出字符串：输入一个字符串后再输入两个字符，输出此字符串中从与第 1 个字符匹配的位置开始到与第 2 个字符匹配的位置结束的所有字符。例如，输入字符串“ program”与 2 个字符“ r ”和“ g ”后，输出“ rog ”。要求自定义函数 char *match(char*s,char ch1, char ch2) 返回结果字符串的首地址。试编写相应程序。

解答：

```
#include<stdio.h>
#include<string.h>
char *match(char *s,char ch1,char ch2);
char p[80];
int main()
{
    char str[10],ch1,ch2,*s;

    printf("Input the string:");
    scanf("%s",str);
    getchar();
    printf("Input start_end character:");
```

```

scanf("%c%c",&ch1,&ch2);
getchar();
s=match(str,ch1,ch2);
printf("%s\n",s);

return 0;
}
char *match(char *s,char ch1,char ch2)
{
    int i=0,j,k=0;
    while(s[i]!=ch1)
        i++;
    j=i;
    while(s[j]!=ch2)
        j++;
    for(;j<=i;j++)
        p[k++]=s[j];
    return p;
}

```

// 为什么设置全局变量

// 局部变量的特点：随函数调用时创建，随函数结束时析构（销毁）。指针指向的是一个正被销毁了的对象。

6. 查找子串：输入两个字符串 s 和 t，在字符串 s 中查找子串 t，输出起始位置，若不存在，则输出 -1。要求自定义函数 char *search(char *s,char *t) 返回子串 t 的首地址，若未找到，则迟回 NULL。试编写相应程序。

解答：

/* 查找子串：输入两个字符串 s 和 t，在字符串 s 中查找子串 t，输出起始位置，若不存在，则输出 -1*/

```

#include<stdio.h>
#include<stdlib.h>
#include<string.h>
char* search(char*s,char*t);
int main(void)
{
    int count,i;
    char *s,*t,*p,ch;

    printf("Input the string(s):\n");
    if((s=(char *)malloc(sizeof(char)))==NULL){
        printf("Not able to allocate memory.\n");
        exit(1);
    };
    gets(s);
    printf("Input the string(t):\n");

```

```

    if((t=(char *)malloc(sizeof(char)))==NULL){
        printf("Not able to allocate memory.\n");
        exit(1);
    };
    gets(t);
    if((p=search(s,t))!=NULL)
        printf("        起始位置：从第 %d个字符开始 \n",p-s+1);
    else
        printf("-1\n");
    free(s);
    free(t);

    return 0;
}
char* search(char*s,char*t)
{
    int T,i,j;
    char *p=NULL;

    for(i=0;i<=strlen(s)-strlen(t);i++)
        if(*(s+i)==*t)
            break;
    p=s+i;
    for(j=0;j<strlen(t);j++,p++)
        if(*p==*(t+j)) T=0;
        else{
            T=1;
            break;
        }
    if(T==1)
        return NULL;
    else
        return s+i;
}

```

7. 奇数值结点链表：输入若干个正整数（输入 -1 为结束标志）建立一个单向链表，头指针为 L，将链表 L 中奇数值的结点重新组成一个新的链表 NEW 并输出新建链表的信息。试编写相应程序。

解答：

/* 奇数值结点链表：输入若干个正整数（输入 -1 为结束标志）建立一个单向链表，头指针为 L，将链表 L 中奇数值的结点重新组成一个新的链表 NEW 并输出新建链表的信息 */

```

#include<stdio.h>
#include<stdlib.h>
struct one_way{

```

```

    int num;
    struct one_way *next;
};
int main(void)
{
    struct one_way *L=NULL,*last,*ptr,*ptr1,*ptr2,*NEW;
    int num;
    do{
        scanf("%d",&num);
        if(num!=-1){
            struct one_way *p=(struct one_way*)malloc(sizeof(struct
one_way));
            p->num=num;
            p->next=NULL;
            if(L==NULL)
                L=p;
            else
                last->next=p;
            last=p;
        }
    }while(num!=-1);
    while(L!=NULL&&L->num%2==0){
        ptr2=L;
        L=L->next;
        free(ptr2);
    }
    if(L==NULL)
        NEW=L;
    ptr1=L;
    ptr2=L->next;
    while(ptr2!=NULL){
        if(ptr2->num%2==0){
            ptr1->next=ptr2->next;
            free(ptr2);
        }
        else
            ptr1=ptr2;
        ptr2=ptr1->next;
    }
    if(NEW==NULL){
        printf("NEW 中不含任何信息 !\n");
        return 0;
    }
    printf("NEW 链表中的信息为 : \n");

```

```

        for(ptr=L;ptr!=NULL;ptr=ptr->next)
            printf("%d ",ptr->num);

        return 0;
    }
}

8. 删除结点： 输入若干个正整数（输入 -1 为结束标志）建立一个单向链表，再输入一个整数 m，删除链表中值为 m 的所有结点。试编写相应程序。
解答：
#include<stdio.h>
#include<stdlib.h>
struct one_way{
    int num;
    struct one_way *next;
};
int main(void)
{
    struct one_way *head=NULL,*last,*ptr,*ptr1,*ptr2;
    int num,m;
    do{
        scanf("%d",&num);
        if(num!=-1){
            struct one_way *p=(struct one_way*)malloc(sizeof(struct one_way));
            p->num=num;
            p->next=NULL;
            if(head==NULL)
                head=p;
            else
                last->next=p;
            last=p;
        }
    }while(num!=-1);
    printf("Enter m:");
    scanf("%d",&m);
    while(head!=NULL&&head->num==m){
        ptr2=head;
        head=head->next;
        free(ptr2);
    }
    if(head==NULL)
        printf("链表中无信息！");
    ptr1=head;
    ptr2=head->next;
    while(ptr2!=NULL){

```



```
    if(ptr2->num==m){
        ptr1->next=ptr2->next;
        free(ptr2);
    }
    else
        ptr1=ptr2;
    ptr2=ptr1->next;
}
for(ptr=head;ptr!=NULL;ptr=ptr->next)
    printf("%d ",ptr->num);

return 0;
}
```

第 12 章 文件

【练习 12-1】读出例 12-1 学生成绩文件 f12-1.txt 内容，输出最高分和最低分及相应的学号和姓名。

解答：

```
#include<stdio.h>
#include<stdlib.h>
struct student{
    long num;
    char stname[20];
    int score;
};
int main(void)
{
    FILE *fp;
    int i,max,min,j=0,k=0;
    struct student students[5];

    if((fp=fopen("f12-1.txt","r"))==NULL) {
        printf("File open error!\n");
        exit(0);
    }
    fscanf(fp,"%ld%s%d",&students[0].num,students[0].stname,&students[0].score);
    max=min=students[0].score;
    for(i=1;i<=4;i++){

fscanf(fp,"%ld%s%d",&students[i].num,students[i].stname,&students[i].score);
        if(max<students[i].score){
            max=students[i].score;
            j=i;
        }
        if(min>students[i].score){
            min=students[i].score;
            k=i;
        }
    }
    printf("Max
score: %d,num:%d,name:%s\n",students[j].score,students[j].num,&students[j].stname);
    printf("Min
```

```

score: %d,num:%d,name:%s\n",students[k].score,students[k].num,&students[k].stname);
    if(fclose(fp)){
        printf("Can not close the file!\n");
        exit(0);
    }
    return 0;
}

```

【练习 12-2】请使用例 8-9 答电码加密函数对民吗字符串进行加密， 改写例 12-2。

解答：

```

#include<stdio.h>
#include<string.h>
#include<stdlib.h>
struct sysuser{
    char username[20];
    char password[8];
};
void encrypt(char *pwd);
int main(void)
{
    FILE *fp;
    int i;
    struct sysuser su;
    if((fp=fopen("f12-2.txt","w"))==NULL){
        printf("File open error!\n");
        exit(0);
    }
    for(i=1;i<=5;i++){
        printf("Enter %dth sysuser (name password):",i);
        scanf("%s%s",su.username,su.password);
        encrypt(su.password);
        fprintf(fp,"%s %s\n",su.username,su.password);
    }
    if(fclose(fp)){
        printf("Can not close the file!\n");
        exit(0);
    }
    return 0;
}
void encrypt(char *pwd)
{
    int i;

```

```

for(i=0;i<strlen(pwd);i++)
    if(pwd[i]=='z')
        pwd[i]='a';
    else
        pwd[i]+=1;
}

```

【练习 12-3】例 12-3 中为什么在执行 fputc (ch , fp2) 前要判断 ch 的值是否等于 EOF? 改写例 12-3 的程序，在复制用户信息文件后，再统计被复制文件中字符的数量。

解答：

文件结束符 EOF 是一个值为 -1 的常量，读文件时可用来判断从文件中读入的字符是否为 EOF 来决定循环是否继续。

```

#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    FILE *fp1,*fp2;
    char ch;
    int count=0;

    if((fp1=fopen("f12-2.txt","r"))=NULL){
        printf("File open error!\n");
        exit(0);
    }
    if((fp2=fopen("f12-3.txt","w"))==NULL){
        printf("File open error!\n");
        exit(0);
    }
    while(!feof(fp1)){
        ch=fgetc(fp1);
        if(ch!=EOF) {
            fputc(ch,fp2);
            count++;
        }
    }
    if(fclose(fp1)){
        printf("Can not close the file!\n");
        exit(0);
    }
    if(fclose(fp2)){
        printf("Can not close the file!\n");
        exit(0);
    }
}

```

```
printf("f12-2    中字符数量为： %d",count);

    return 0;
}
```

【练习 12-4】字母转换并统计行数：读取一个指定的文本文件，显示在屏幕上，如果有大写字母，则改成小写字母再输出，并根据回车符统计行数。试编写相应程序。

解答：

```
#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    char ch;
    int countline=0;
    FILE *fp;
    if((fp=fopen("    练习 12-4.txt","r"))==NULL){
        printf("Not open!");
        exit(0);
    }
    while(!feof(fp)){
        ch=fgetc(fp);
        if(ch!=EOF)
            if(ch>='A'&&ch<='Z')
                printf("%c",ch-'A'+ 'a');
            else
                printf("%c",ch);
        if(ch=='\n')
            countline++;
    }
    printf("\n");
    printf("file's line is:%d.",countline+1);
    if(fclose(fp)){
        printf("Can not close!");
        exit(0);
    }
    return 0;
}
```

【练习 12-5】写字符并验证：从键盘输入一行字符，写入到文件 f3.txt 中，并重新读出，最终在屏幕上显示验证。程序输入以读到回车符“ n”为结束，读文件时要用 EOF来控制循环。试编写相应程序。

解答：

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

```

int main(void)
{
    FILE *fp;
    char ch;
    if((fp=fopen("f3.txt","w+"))==NULL){
        printf("can not open file!");
        exit(0);
    }
    printf("Input the string:\n");
    ch=getchar();
    while(ch!='\n'){
        fputc(ch,fp);
        ch=getchar();
    }
    rewind(fp);
    while(!feof(fp)){
        ch=fgetc(fp);
        if(ch!=EOF)
            putchar(ch);
    }
    printf("\n");
    if(fclose(fp)){
        printf("can not close file!\n");
        exit(0);
    }
    return 0;
}

```

【练习 12-6】实数取整写入文件：文件 f1.txt 中有若干个实数，请分别读出，将每个实数按四舍五入取整后存入文件 f2.txt 中。试编写相应程序。

解答：

```

#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    FILE *fp1,*fp2;
    double a;
    if((fp1=fopen("f1.txt","r"))==NULL){
        printf("File open error!\n");
        exit(0);
    }
    if((fp2=fopen("f2.txt","w"))==NULL){
        printf("File open error!\n");
        exit(0);
    }
}

```

```

}
while(!feof(fp1)){
    fscanf(fp1,"%lf",&a);
    fprintf(fp2,"%0.0f ",a);
}
if(fclose(fp1)){
    printf("Can not close the file!\n");
    exit(0);
}
if(fclose(fp2)){
    printf("Can not close the file!\n");
    exit(0);
}
return 0;
}

```

【练习 12-7】修改例 12-6, 增加修改资金账户的功能。输入一个记录 ID , 如果文件中已存在该记录 , 则输入新的记录信息并更新资金账户文件中相应记录的信息。要求定义和调用函数 UpdateLog () , 其功能是修改资金账户记录。

解答 :

```

#include<stdio.h>
#include<stdlib.h>
long size;
struct LogData{
    long logid;
    char logdate[11];
    char lognote[15];
    double charge;
    double balance;
};
int inputchoice()
{
    int mychoice;
    printf("\nEnter your choice: \n");
    printf("1-Add a new cash LOG.\n2-List All Cash LOG.\n");
    printf("3-Query Last Cash LoG.\n0-End program.\n");
    scanf("%d",&mychoice);
    return mychoice;
}
long getLogcount(FILE *cfptr)
{
    long begin,end,logcount;
    fseek(cfptr,OL,SEEK_SET);
    begin=ftell(cfptr);
    fseek(cfptr,size,SEEK_END);
}

```

```

        end=ftell(cfptr);
        logcount=(end-begin)/size-1;
        return logcount;
    }
/* 列出所有收支流水账 */
void ListAllLog(FILE *cfptr)
{
    struct LogData log;
    fseek(cfptr,OL,SEEK_SET);
    fread(&log,size,1,cfptr);
    printf("logid logdate lognote charge balance \n");
    while(!feof(cfptr)){

        printf("%6ld%-11s%-15%10.2lf%10.2lf\n",log.logid,log.logdate,log.logn
ote,log.charge,log.balance);
        fread(&log,size,1,cfptr);
    };
}
/* 查询显示最后一条记录 */
void QueryLastLog(FILE *cfptr)
{
    struct LogData log;
    long logcount;
    logcount=getlogcount(cfptr);
    if(logcount>0)
    {
        fseek(cfptr,size*(logcount-1),SEEK_SET);
        fread(&log,size,1,cfptr);
        printf("The last log is:\n");

        printf("logid:%-6ld\nlogdate:%-11s\nlognote:%-15s\n",log.logid,log.lo
gdate,log.lognote);

        printf("charge:%-10.2lf\nbalance:%-10.2lf\n",log.charge,log.balance);
    }
    else printf("no logs in file!\n");
}
/* 添加新记录 */
void AddNewLog(FILE *cfptr)
{
    struct LogData log,lastlog;
    long logcount;
    printf("Input logdate (format: 2006-01-01):");
    scanf("%s",log.logdate);

```



```

printf("Input lognote:"); scanf("%s",log.lognote);
printf("Input Charge: Income+and epend-:");
scanf("%lf",&log.charge);
logcount=getLogcount(cfptr);

if(logcount>0){
    fseek(cfptr,size*(logcount-1),SEEK_SET);
fread(&lastlog,size,1,cfptr)
    log.logid=lastlog.logid+1;
    log.balance=log.charge+lastlog.balance;
}
else{
    log.logid=1;
    log.balance=log.charge;
}
rewind(cfptr);logid=lastlog.logid;
printf("logid=%ld\n",log.logid);
fwrite(&log,sizeof(struct LogData),1,cfptr);
}
/* 修改资金账户 */
void UpdateLog(FILE *cfptr)
{
    FILE *fpout;
    struct LogData user;
    char date[11];
    char note[15];
    double charge;
    double balance;
    int choice,ID;
    cfptr=fopen("r");
    if((fpout=fopen("cash.dat","w"))==NULL){
        printf("can not open the file !\n");
        exit(0);
    }
    printf("Enter LogID:");
    scanf("%d",&ID);
    while(!feof(cfptr)){
        fread(&user,sizeof(struct LogData),1,cfptr);
        if(strcmp(user.logid,ID)==0){
            printf("        请输入修改信息 :\n");
            printf("Date:");
            scanf("%s",date);
            strcpy(user.logdate,date);
            printf("Note:");

```

```

        scanf("%s",note);
        strcpy(user.lognote,note);
        printf("Charge:");
        user.charge=charge;
        printf("Balance:");
        scanf("%s",&balance);
        user.balance=balance;
        fwrite(&user,sizeof(struct LogData),1,fpout);
    }
    else
        fwrite(&user,sizeof(struct LogData),1,fpout);
    }
    if(fclose(cfptr)){
printf("can not close file!\n");
exit(0);
}
    if(fclose(fpout)){
printf("can not close file!\n");
exit(0);
}
    unlink("cashbox.dat");//remove("cashbox.dat");
    rename("cash.dat.dat","cashbox.dat");
}
/* 打开文件 */
FILE *openfile(char *openmode)
{
    FILE *fp;
    if((fp=fopen("cashbox.dat",openmode))==NULL){
        printf("can not open file cashbox.dat!\n");
        exit(0);
    }
    return (fp);
}
int main(void)
{
    FILE *fp; int choice;
    size=sizeof(struct LogData);
    while((choice=inputchoice())!=0){
        switch(choice){
            case 1:
                fp=openfile("ab+");
                AddNewLog(fp);
                break;
            /*      列出所有的收入支出情况      */

```

都是删除文件的函数

```

    case 2:
        fp=fopen("rb");
        ListAllLog(fp);
        break;
    /*    查询最后记录及余额    */
    case 3:
        fp=fopen("rb");
        QueryLastLog(fp);
        break;
    case 4:
        fp=fopen("rb");
        UpdateLog(fp);
        break;
    default:
        printf("Input Error."); break;
}
}
if(fclose(fp)){
    printf("Can not close the file!\n");
    exit(0);
}
return 0;
}

```

习题 12

一、选择题

1. 以下语句将输出 ____B____。

```
#include<stdio.h>
printf ("%d %d %d",NULL,    '\0',EOF);
```

A.0 0 1 B.0 0 -1 C.NULL EOF D.1 0 EOF

2. 如果二进制文件 a.dat 已经存在，现在要写入全新数据，应以 ____B____方式打开。

A."w" B."wb" C."w+" D. "wb+"

3. 定义 " FILE *fp;"， 则文件指针 fp 指向的是 ____D____。

A. 文件在磁盘上的读写位置 B 文件在缓冲区上的读写位置
C. 整个磁盘文件 D. 文件类型结构

4. 缓冲文件系统的文件缓冲区位于 ____C____。

A. 磁盘缓冲区中 B. 磁盘文件中
C. 内存数据区中 D. 程序文件中

5. 使文件指针重新定位到文件读写的首地址的函数是 ____C____。

A.ftell () B.fseek () C.rewind () D.ferror()

二、填空题

1. 函数 fopen () 的返回值是 _指向文件缓冲区的首地址的文件结构类型指针_。
2. 文件的三大特征是 _名称_、_大小_和_类型_。
3. 缓冲文件系统与非缓冲文件系统的不同点在于 _系统是否为文件自动分配一块文件缓冲区 (内存单元)_。
4. 只能向指定文件写入一个字符的函数是 _____fputc() 函数_____。
5. 判断文件指针是否已经到了文件尾部的函数是 _____feof() 函数_____。
6. 阅读程序，以下程序完成的功能是 _文件 infile 内容复制到文件_。

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

```
int main (void)
```

```
{ char infile [10],outfile [10];
```

```
FILE * fpa,* fpb;
```

```
gets (infile);
```

```
gets (outfile);
```

```
fpa=fopen (infile,"r" );
```

```
fpb=fopen (outfile," w" );
```

```
while (! feof (fpa) )
```

```
    fputc(fgetc(fpa),fpb);
```

```
fclose (fpa) ;
```

```
fclose (fpb) ;
```

```
return 0;
```

```
}
```

三、程序设计题

1. 统计文本文件中各类字符个数：分别统计一个文本文件中字母、数字及其他字符的个数。试编写相应程序。

解答：

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

```
#include<stdlib.h>
```

```
int main(void)
```

```
{
```

```
FILE *fp;
```

```
char ch;
```

```
int letter=0,digiter=0,other=0;
```

```
if((fp=fopen("12-3.1.txt","r"))==NULL){
```

```
    printf("File open error!\n");
```

```
    exit(0);
```

```
}
```

```
while((ch=fgetc(fp))!=EOF){
```

```
    if(ch>='A'&&ch<='Z'||ch>='a'&&ch<='z')
```

```
        letter++;
```

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

```
        digiter++;
```

```

        else
            other++;
    }
    printf("letter=%d,digiter=%d,other=%d",letter,digiter,other);
    if(fclose(fp)){
        printf("Can not close the file!\n");
        exit(0);
    }

    return 0;
}

```

2. 将实数写入文件：从键盘输入若干实数（以特殊数值 -1 结束），分别写到一个文本文件中。试编写相应程序。

解答：

```

#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    FILE *fp;
    int number;

    if((fp=fopen("12-3.2.txt","w"))==NULL){
        printf("can not open file!\n");
        exit(0);
    }
    printf("Input number:\n");
    scanf("%d",&number);
    while(number!=-1){
        fprintf(fp,"%d",number);
        scanf("%d",&number);
    }
    if(fclose(fp)){
        printf("Can not close the file!\n");
        exit(0);
    }
    return 0;
}

```

3. 比较两个文本文件是否相等：比较两个文本文件的内容是否相同，并输出两个文件中第一次出现不同字符内容的行号及列值。试编写相应程序。

解答：

```

#include<stdio.h>
#include<stdlib.h>

```

```

int main(void)
{
    FILE *fp1,*fp2;
    int i=1,j=1;
    char ch1,ch2;

    if((fp1=fopen("12-3.3.1.txt","r"))==NULL){
        printf("can not open file!\n");
        exit(0);
    }
    if((fp2=fopen("12-3.3.2.txt","r"))==NULL){
        printf("can not open file!\n");
        exit(0);
    }
    ch1=fgetc(fp1);
    ch2=fgetc(fp2);
    while(ch1!=EOF&&ch2!=EOF){
        if(ch1!=ch2)
            break;
        else if(ch1!='\n'){
            j++;
            ch1=fgetc(fp1);
            ch2=fgetc(fp2);
        }
        else{
            i++;
            j=1;
            ch1=fgetc(fp1);
            ch2=fgetc(fp2);
        }
    }
    printf("    首次不同的字符在第  %d行、第 %d列。 \n",i,j);
    if(fclose(fp1)){
        printf("can not close file!\n");
        exit(0);
    }
    if(fclose(fp2)){
        printf("can not close file!\n");
        exit(0);
    }
}

```

4. 将文件中的数据求和并写入文本文件尾 ： 文件 Int_Data.dat 中存放了若干整数，将文件中所有数据相加， 并把累加和写入该文件的最后。 试编写相应程序。

解答：

```
#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    FILE *fp;
    int x,sum=0;

    if((fp=fopen("Int_Date.dat","ab+"))==NULL){
        printf("can not open file!");
        exit(0);
    }
    while(!feof(fp)){
        fscanf(fp,"%d",&x);
        sum+=x;
    }
    fprintf(fp," %d",sum);
    if(fclose(fp)){
        printf("can not close the file!\n");
        exit(0);
    }

    return 0;
}
```

5. 输出含 for 的行：将文本文件 test.txt 中所有包含字符串“ for ”的行输出。试编写相应程序。

解答：

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
char s[999];
int main(void)
{
    FILE* fp;
    int i;
    if((fp=fopen("test.txt","r"))==NULL){
        printf("can not open file!\n");
        exit(0);
    }
    while(!feof(fp)){
        fgets(s,999,fp);
        if(strlen(s)>3){
            for(i=0;i<strlen(s)-2;i++)
```

```

        if(s[i]=='f'&&s[i+1]=='o'&&s[i+2]=='r')
            printf("%s",s);
    }
}
if(fclose(fp)){
    printf("can not close file!\n");
    exit(0);
}
return 0;
}

```

6. 删除文件中的注释：将 C 语言源程序 (hello.c) 文件中的所有注释去掉后存入另一个文件 (new_hello.c)。试编写相应程序。

解答：

```

#include<stdio.h>
#include<stdlib.h>
int main(void)
{
    FILE *fp1,*fp2;
    char ch,ch1,ch2,s[99];

    if((fp1=fopen("hello.c","r"))==NULL){
        printf("can not open file!\n");
        exit(0);
    }
    if((fp2=fopen("new_hello.c","w"))==NULL){
        printf("can not open file!\n");
        exit(0);
    }
    while(!feof(fp1)){
        ch=fgetc(fp1);
        if(ch=='/'){
            if((ch1=fgetc(fp1))=='*')
                while(fgetc(fp1)!='*'&&(ch1=fgetc(fp1))!='/')
                    fseek(fp1,-sizeof(ch1),1);
            else if(ch1=='/'){
                ch='\0';
                for(;ch1!='\n';ch1=fgetc(fp1))
                    ch1='\0';
            }
        }
        else if(ch!=EOF)
            fputc(ch,fp2);
    }
}

```



```
if(fclose(fp2)){
    printf("can not close file!\n");
    exit(0);
}

return 0;
}
```

7.(选做) 账户余额管理： 创建一个随机文件，用来存储银行账户和余额信息，程序要求能够查询某个账户的余额，当客户发生交易额时（正表示存入，负表示取出）能够更新余额。账户信息包括账号、账号名和余额三个数据项。试编写相应程序。

文件部分内容如下：

AcctNo	AcctName	Balance
1	zhangsan	1000.00
2	lisi	300.00
3	wangwu	-100.00
.....		

解答：

```
#include <stdio.h>
#include<string.h>
#include <stdlib.h>
long size;
struct account{
    char no[10];
    char acctname[50];
    double balance;
};
FILE *openfile(char *openmode){
    FILE *fp;
    if((fp=fopen("accout.dat",openmode))==NULL){
        printf("Can not open the file!");
        exit(0);
    }
    return fp;
}
double userbalance(FILE *fp,char *name);
void pay(FILE *fp,char *name,double count);
int main()
{
    FILE *fp;
    int choice;
    char name[50];
    double balance;
```

```

double count;

printf("    请输入选择类型： \n");
printf("1.    查账户余额 \n");
printf("2.    账户交易 \n");
printf("    退出按 exit\n");
printf("    选择类型： ");
scanf("%d",&choice);
    switch(choice){
        case 1:
            printf("        输入名字 ");
            scanf("%s",name);
            fp=fopen("r+");
            balance=userbalance(fp,name);
            printf("%.2f",balance);
            break;
        case 2:
            printf("        输入名字 ");
            scanf("%s",name);
            fp=fopen("r+");
            printf("        输入交易金额 ");
            scanf("%lf",&count);
            pay(fp,name,count);
            break;
    }
    default:
        exit(0);
        break;
}

return 0;
}
double userbalance(FILE *fp,char *name){
    struct account user;
    double balance;

    rewind(fp);
    while(!feof(fp)){
        fscanf(fp,"%s %s %lf",user.no,user.acctname,&user.balance);
        if(strcmp(user.acctname,name)==0){
            balance=user.balance;
            break;
        }
    }
}

```

```

    return balance;
}
void pay(FILE *fp,char *name,double count){

FILE *fpout;
    struct account user;
    double balance;
    balance=userbalance(fp,name);
    balance=balance+count;
    rewind(fp);
    fpout=fopen("temp.dat","w");
    while(!feof(fp)){
        fscanf(fp,"%s %s %lf",user.no,user.acctname,&user.balance);
        if(strcmp(user.acctname,name)==0){
            user.balance=balance;

fprintf(fpout,"%s %s %lf\n",user.no,user.acctname,user.balance);
        }
        else

fprintf(fpout,"%s %s %lf\n",user.no,user.acctname,user.balance);
    }
    if(fclose(fp)){
        printf("can not close file!\n");
        exit(0);
    }
    if(fclose(fpout)){
        printf("can not close file!\n");
        exit(0);
    }
    unlink("accout.dat");//remove("accout.dat");
    rename("temp.dat","accout.dat");
}

```

都是删除文件的函数