

JAVA 经典算法 40 例

【程序 1】 题目：古典问题：有一对兔子，从出生后第 3 个月起每个月都生一对兔子，小兔子长到第四个月后每个月又生一对兔子，假如兔子都不死，问每个月的兔子总数为多少？

1.程序分析： 兔子的规律为数列 1,1,2,3,5,8,13,21....

```
public class exp2{
    public static void main(String args[]){
        int i=0;
        for(i=1;i<=20;i++){
            System.out.println(f(i));
        }
    }
    public static int f(int x)
    {
        if(x==1 || x==2)
            return 1;
        else
            return f(x-1)+f(x-2);
    }
}
或
public class exp2{
    public static void main(String args[]){
        int i=0;
        math mymath = new math();
        for(i=1;i<=20;i++){
            System.out.println(mymath.f(i));
        }
    }
}
class math
{
    public int f(int x)
    {
        if(x==1 || x==2)
            return 1;
        else
            return f(x-1)+f(x-2);
    }
}
```

【程序 2】 题目：判断 101-200 之间有多少个素数，并输出所有素数。

1.程序分析：判断素数的方法：用一个数分别去除 2 到 sqrt(这个数)，如果能被整除，则表明此数不是素数，反之是素数。

```
public class exp2{
    public static void main(String args[]){
```

```
        int i=0;
        math mymath = new math();
        for(i=2;i<=200;i++)
            if(mymath.iszhishu(i)==true)
                System.out.println(i);
    }
}
class math
{
    public int f(int x)
    {
        if(x==1 || x==2)
            return 1;
        else
            return f(x-1)+f(x-2);
    }
    public boolean iszhishu(int x)
    {
        for(int i=2;i<=x/2;i++)
            if (x % i==0 )
                return false;
        return true;
    }
}
```

【程序 3】 题目：打印出所有的 "水仙花数"，所谓 "水仙花数"是指一个三位数，其各位数字立方和等于该数本身。例如：153 是一个 "水仙花数"，因为 $153=1^3+5^3+3^3$ 。

1.程序分析：利用 for 循环控制 100-999 个数，每个数分解出个位，十位，百位。

```
public class exp2{
    public static void main(String args[]){
        int i=0;
        math mymath = new math();
        for(i=100;i<=999;i++)
            if(mymath.shuixianhua(i)==true)
                System.out.println(i);
    }
}
class math
{
    public int f(int x)
    {
        if(x==1 || x==2)
            return 1;
        else
```

```

        return f(x-1)+f(x-2);
    }
    public boolean iszhishu(int x)
    {
        for(int i=2;i<=x/2;i++)
            if (x % i==0 )
                return false;
        return true;
    }
    public boolean shuixianhua(int x)
    {
        int i=0,j=0,k=0;
        i=x / 100;
        j=(x % 100) /10;
        k=x % 10;
        if(x==i*i*i+j*j*j+k*k*k)
            return true;
        else
            return false;
    }
}

```

【程序 4】 题目：将一个正整数分解质因数。例如：输入 90,打印出 90=2*3*3*5。

程序分析：对 n 进行分解质因数，应先找到一个最小的质数 k，然后按下述步骤完成：

(1)如果这个质数恰等于 n，则说明分解质因数的过程已经结束，打印出即可。

(2)如果 $n < k$ ，但 n 能被 k 整除，则应打印出 k 的值，并用 n 除以 k 的商,作为新的正整数你,重复执行第一步。

(3)如果 n 不能被 k 整除，则用 k+1 作为 k 的值,重复执行第一步。

```

public class exp2{
    public exp2(){
    }
    public void fengjie(int n){
        for(int i=2;i<=n/2;i++){
            if(n%i==0){
                System.out.print(i+"*");
                fengjie(n/i);
            }
        }
        System.out.print(n);
        System.exit(0);///不能少这句，否则结果会出错
    }
    public static void main(String[] args){
        String str="";
        exp2 c=new exp2();
        str=javax.swing.JOptionPane.showInputDialog("请输入 N 的值（输入 exit 退出）：");
    }
}

```

```
        int N;
        N=0;
        try{
            N=Integer.parseInt(str);
        }catch(NumberFormatException e){
            e.printStackTrace();
        }
        System.out.print(N+"分解质因数：" +N+"=");
        c.fengjie(N);
    }
}
```

【程序 5】 题目：利用条件运算符的嵌套来完成此题：学习成绩 ≥ 90 分的同学用 A 表示，60-89 分之间的用 B 表示，60 分以下的用 C 表示。

1.程序分析：(a > b)?a:b 这是条件运算符的基本例子。

```
import javax.swing.*;
public class ex5 {
    public static void main(String[] args){
        String str="";
        str=JOptionPane.showInputDialog("请输入 N 的值（输入 exit 退出）：");
        int N;
        N=0;
        try{
            N=Integer.parseInt(str);
        }
        catch(NumberFormatException e){
            e.printStackTrace();
        }
        str=(N>90?"A":(N>60?"B":"C"));
        System.out.println(str);
    }
}
```

【程序 6】 题目：输入两个正整数 m 和 n，求其最大公约数和最小公倍数。

1.程序分析：利用辗除法。

最大公约数：

```
public class CommonDivisor{
    public static void main(String args[])
    {
        commonDivisor(24,32);
    }
    static int commonDivisor(int M, int N)
    {
        if(N<0||M<0)
        {
            System.out.println("ERROR!");
        }
    }
}
```

```
        return -1;
    }
    if(N==0)
    {
        System.out.println("the biggest common divisor is :"+M);
        return M;
    }
    return commonDivisor(N,M%N);
}
}
```

最小公倍数和最大公约数：

```
import java.util.Scanner;
public class CandC
{
    //下面的方法是求出最大公约数
    public static int gcd(int m, int n)
    {
        while (true)
        {
            if ((m = m % n) == 0)
                return n;
            if ((n = n % m) == 0)
                return m;
        }
    }
    public static void main(String args[]) throws Exception
    {
        //取得输入值
        //Scanner chin = new Scanner(System.in);
        //int a = chin.nextInt(), b = chin.nextInt();
        int a=23; int b=32;
        int c = gcd(a, b);
        System.out.println("最小公倍数：" + a * b / c + "\n最大公约数：" + c);
    }
}
```

【程序 7】 题目：输入一行字符，分别统计出其中英文字母、空格、数字和其它字符的个数。

1.程序分析：利用 while 语句,条件为输入的字符不为 '\n'.

```
import java.util.Scanner;
public class ex7 {
    public static void main(String args[])
    {
        System.out.println("请输入字符串：");
        Scanner scan=new Scanner(System.in);
        String str=scan.next();
```

```
String E1="[\u4e00-\u9fa5]";
String E2="[a-zA-Z]";
int countH=0;
int countE=0;
char[] arrChar=str.toCharArray();
String[] arrStr=new String[arrChar.length];
for (int i=0;i<arrChar.length ;i++ )
{
    arrStr[i]=String.valueOf(arrChar[i]);
}
for (String i: arrStr )
{
    if (i.matches(E1))
    {
        countH++;
    }
    if (i.matches(E2))
    {
        countE++;
    }
}
System.out.println("汉字的个数"+countH);
System.out.println("字母的个数"+countE);
}
```

【程序 8】 题目：求 $s=a+aa+aaa+aaaa+aa...a$ 的值，其中 a 是一个数字。例如 $2+22+222+2222+22222$ (此时共有 5 个数相加)，几个数相加有键盘控制。

1.程序分析：关键是计算出每一项的值。

```
import java.io.*;
public class Sumloop {
    public static void main(String[] args) throws IOException
    {
        int s=0;
        String output="";
        BufferedReader stadin = new BufferedReader(new InputStreamReader(System.in));
        System.out.println("请输入 a 的值");
        String input =stadin.readLine();
        for(int i =1;i<=Integer.parseInt(input);i++)
        {
            output+=input;
            int a=Integer.parseInt(output);
            s+=a;
        }
        System.out.println(s);
    }
}
```

```

    }
}
另解:
import java.io.*;
public class Sumloop {
    public static void main(String[] args) throws IOException
    {
        int s=0;
        int n;
        int t=0;
        BufferedReader stadin = new BufferedReader(new InputStreamReader(System.in));
        String input = stadin.readLine();
        n=Integer.parseInt(input);
        for(int i=1;i<=n;i++){
            t=t*10+n;
            s=s+t;
            System.out.println(t);
        }
        System.out.println(s);
    }
}

```

【程序 9】 题目：一个数如果恰好等于它的因子之和，这个数就称为“完数”。例如 $6=1+2+3$ 。编程找出 1000 以内的所有完数。

```

public class Wanshu {
    public static void main(String[] args)
    {
        int s;
        for(int i=1;i<=1000;i++)
        {
            s=0;
            for(int j=1;j<i;j++)
                if(i % j==0)
                    s=s+j;
            if(s==i)
                System.out.print(i+" ");
        }
        System.out.println();
    }
}

```

【程序 10】 题目：一球从 100 米高度自由落下，每次落地后反跳回原高度的一半；再落下，求它在第 10 次落地时，共经过多少米？第 10 次反弹多高？

```

public class Ex10 {
    public static void main(String[] args)
    {

```

```
double s=0;
double t=100;
for(int i=1;i<=10;i++)
{
    s+=t;
    t=t/2;
}
System.out.println(s);
System.out.println(t);

}
}
```

【程序 11】 题目：有 1、2、3、4 个数字，能组成多少个互不相同且无重复数字的三位数？都是多少？

1.程序分析：可填在百位、十位、个位的数字都是 1、2、3、4。组成所有的排列后再去 掉不满足条件的排列。

```
public class Wanshu {
    public static void main(String[] args)
    {
        int i=0;
        int j=0;
        int k=0;
        int t=0;
        for(i=1;i<=4;i++)
            for(j=1;j<=4;j++)
                for(k=1;k<=4;k++)
                    if(i!=j && j!=k && i!=k)
                        {t+=1;
                            System.out.println(i*100+j*10+k);
                        }
        System.out.println (t);
    }
}
```

【程序 12】 题目：企业发放的奖金根据利润提成。利润(I)低于或等于 10 万元时，奖金可提 10%；利润高于 10 万元，低于 20 万元时，低于 10 万元的部分按 10%提成，高于 10 万元的部分，可提成 7.5%；20 万到 40 万之间时，高于 20 万元的部分，可提成 5%；40 万到 60 万之间时高于 40 万元的部分，可提成 3%；60 万到 100 万之间时，高于 60 万元的部分，可提成 1.5%，高于 100 万元时，超过 100 万元的部分按 1%提成，从键盘输入当月利润 I，求应发放奖金总数？

1.程序分析：请利用数轴来分界，定位。注意定义时需把奖金定义成长整型。

```
import java .util.*;
public class test {
    public static void main (String[]args){
        double sum;//声明要储存的变量应发的奖金
        Scanner input =new Scanner (System.in);//导入扫描器
        System.out.print ("输入当月利润");
```



```
double lirun=input.nextDouble();//从控制台录入利润
if(lirun<=100000){
    sum=lirun*0.1;
}else if (lirun<=200000){
    sum=10000+lirun*0.075;
}else if (lirun<=400000){
    sum=17500+lirun*0.05;
}else if (lirun<=600000){
    sum=lirun*0.03;
}else if (lirun<=1000000){
    sum=lirun*0.015;
} else{
    sum=lirun*0.01;
}
System.out.println("应发的奖金是"+sum);
}
```

后面其他情况的代码可以由读者自行完善.

【程序 13】

题目：一个整数，它加上 100 后是一个完全平方数，加上 168 又是一个完全平方数，请问该数是多少？

1.程序分析：在 10 万以内判断，先将该数加上 100 后再开方，再将该数加上 268 后再开方，如果开方后的结果满足如下条件，即是结果。请看具体分析：

```
public class test {
    public static void main (String[]args){
        long k=0;
        for(k=1;k<=100000;k++)
            if(Math.floor(Math.sqrt(k+100))==Math.sqrt(k+100)           &&
Math.floor(Math.sqrt(k+168))==Math.sqrt(k+168))
                System.out.println(k);
    }
}
```

【程序 14】 题目：输入某年某月某日，判断这一天是这一年的第几天？

1.程序分析：以 3 月 5 日为例，应该先把前两个月的加起来，然后再加上 5 天即本年的第几天，特殊情况，闰年且输入月份大于 3 时需考虑多加一天。

```
import java.util.*;
public class test {
    public static void main (String[]args){
        int day=0;
        int month=0;
        int year=0;
        int sum=0;
        int leap;
        System.out.print("请输入年,月,日\n");
```

```
Scanner input = new Scanner(System.in);
year=input.nextInt();
month=input.nextInt();
day=input.nextInt();
switch(month) /*先计算某月以前月份的总天数*/
{
case 1:
    sum=0;break;
case 2:
    sum=31;break;
case 3:
    sum=59;break;
case 4:
    sum=90;break;
case 5:
    sum=120;break;
case 6:
    sum=151;break;
case 7:
    sum=181;break;
case 8:
    sum=212;break;
case 9:
    sum=243;break;
case 10:
    sum=273;break;
case 11:
    sum=304;break;
case 12:
    sum=334;break;
default:
    System.out.println("data error");break;
}
sum=sum+day; /*再加上某天的天数*/
if(year%400==0||(year%4==0&&year%100!=0))/*判断是不是闰年*/
    leap=1;
else
    leap=0;
if(leap==1 && month>2)/*如果是闰年且月份大于 2,总天数应该加一天*/
    sum++;
System.out.println("It is the the day:"+sum);
}
```

【程序 15】 题目：输入三个整数 x,y,z，请把这三个数由小到大输出。

1.程序分析：我们想办法把最小的数放到x上，先将x与y进行比较，如果x>y则将x与y的值进行交换，然后再用x与z进行比较，如果x>z则将x与z的值进行交换，这样能使x最小。

```
import java.util.*;
public class test {
    public static void main (String[]args){
        int i=0;
        int j=0;
        int k=0;
        int x=0;
        System.out.print("请输入三个数\n");
        Scanner input = new Scanner(System.in);
        i=input.nextInt();
        j=input.nextInt();
        k=input.nextInt();
        if(i>j)
        {
            x=i;
            i=j;
            j=x;
        }
        if(i>k)
        {
            x=i;
            i=k;
            k=x;
        }
        if(j>k)
        {
            x=j;
            j=k;
            k=x;
        }
        System.out.println(i+", "+j+", "+k);
    }
}
```

【程序 16】 题目：输出 9*9 口诀。

1.程序分析：分行与列考虑，共 9 行 9 列，i 控制行，j 控制列。

```
public class jiujiu {
    public static void main(String[] args)
    {
        int i=0;
        int j=0;
        for(i=1;j<=9;i++)
        {    for(j=1;j<=9;j++)
```

```
        System.out.print(i+"*"+j+"="+i*j+"\t");
        System.out.println();
    }
}
}
不出现重复的乘积(下三角)
public class jiujiu {
    public static void main(String[] args)
    {
        int i=0;
        int j=0;
        for(i=1;i<=9;i++)
        {
            for(j=1;j<=i;j++)
                System.out.print(i+"*"+j+"="+i*j+"\t");
            System.out.println();
        }
    }
}
```

```
上三角
public class jiujiu {
    public static void main(String[] args)
    {
        int i=0;
        int j=0;
        for(i=1;i<=9;i++)
        {
            for(j=i;j<=9;j++)
                System.out.print(i+"*"+j+"="+i*j+"\t");
            System.out.println();
        }
    }
}
```

【程序 17】 题目：猴子吃桃问题：猴子第一天摘下若干个桃子，当即吃了一半，还不瘾，又多吃了一个 第二天早上又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 的一半零一个。到第 10 天早上想再吃时，见只剩下一个桃子了。求第一天共摘了多少。

1.程序分析：采取逆向思维的方法，从后往前推断。

```
public class 猴子吃桃 {
    static int total(int day){
        if(day == 10){
            return 1;
        }
        else{
            return (total(day+1)+1)*2;
        }
    }
}
```

```
public static void main(String[] args)
{
    System.out.println(total(1));
}
}
```

【程序 18】 题目：两个乒乓球队进行比赛，各出三人。甲队为 a,b,c 三人，乙队为 x,y,z 三人。已抽签决定比赛名单。有人向队员打听比赛的名单。a 说他不和 x 比，c 说他不和 x,z 比，请编程找出三队赛手的名单。

1.程序分析：判断素数的方法：用一个数分别去除 2 到 sqrt(这个数)，如果能被整除， 则表明此数不是素数，反之是素数。

```
import java.util.ArrayList;
public class pingpang {
    String a,b,c;
    public static void main(String[] args) {
        String[] op = { "x", "y", "z" };
        ArrayList<pingpang> arrayList=new ArrayList<pingpang>();
        for (int i = 0; i < 3; i++)
            for (int j = 0; j < 3; j++)
                for (int k = 0; k < 3; k++) {
                    pingpang a=new pingpang(op[i],op[j],op[k]);
                    if(!a.a.equals(a.b)&&!a.b.equals(a.c)&&!a.a.equals("x")
                        &&!a.c.equals("x")&&!a.c.equals("z")){
                        arrayList.add(a);
                    }
                }
        for(Object a:arrayList){
            System.out.println(a);
        }
    }
    public pingpang(String a, String b, String c) {
        super();
        this.a = a;
        this.b = b;
        this.c = c;
    }
    @Override
    public String toString() {
        // TODO Auto-generated method stub
        return "a 的对手是"+a+","+"b 的对手是"+b+","+"c 的对手是"+c+"\n";
    }
}
```

【程序 19】 题目：打印出如下图案（菱形）

```
*
***
```

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

1.程序分析：先把图形分成两部分来看待，前四行一个规律，后三行一个规律，利用双重 for 循环，第一层控制行，第二层控制列。

三角形：

```
public class StartG {
    public static void main(String [] args)
    {
        int i=0;
        int j=0;
        for(i=1;i<=4;i++)
        {   for(j=1;j<=2*i-1;j++)
            System.out.print("*");
            System.out.println("");
        }
        for(i=4;i>=1;i--)
        { for(j=1;j<=2*i-3;j++)
            System.out.print("*");
            System.out.println("");
        }
    }
}
```

菱形：

```
public class StartG {
    public static void main(String [] args)
    {
        int i=0;
        int j=0;
        for(i=1;i<=4;i++)
        {
            for(int k=1; k<=4-i;k++)
                System.out.print(" ");
            for(j=1;j<=2*i-1;j++)
                System.out.print("*");
            System.out.println("");
        }
        for(i=4;i>=1;i--)
        {
            for(int k=1; k<=5-i;k++)
                System.out.print(" ");
```

```
        for(j=1;j<=2*i-3;j++)
            System.out.print("*");
        System.out.println("");
    }
}
}
```

【程序 20】 题目：有一分数序列：2/1, 3/2, 5/3, 8/5, 13/8, 21/13...求出这个数列的前 20 项之和。

1.程序分析：请抓住分子与分母的变化规律。

```
public class test20 {
    public static void main(String[] args) {
        float fm = 1f;
        float fz = 1f;
        float temp;
        float sum = 0f;
        for (int i=0;i<20;i++){
            temp = fm;
            fm = fz;
            fz = fz + temp;
            sum += fz/fm;
            //System.out.println(sum);
        }
        System.out.println(sum);
    }
}
```

【程序 21】 题目：求 1+2!+3!+...+20!的和

1.程序分析：此程序只是把累加变成了累乘。

```
public class Ex21 {
    static long sum = 0;
    static long fac = 0;
    public static void main(String[] args) {
        long sum = 0;
        long fac = 1;
        for(int i=1; i<=10; i++) {
            fac = fac * i;
            sum += fac;
        }
        System.out.println(sum);
    }
}
```

【程序 22】 题目：利用递归方法求 5!。

1.程序分析：递归公式：fn=fn_1*4!

```
import java.util.Scanner;

public class Ex22 {

    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int n = s.nextInt();
        Ex22 tfr = new Ex22();
        System.out.println(tfr.recursion(n));
    }

    public long recursion(int n) {
        long value = 0;
        if(n == 1 || n == 0) {
            value = 1;
        } else if(n > 1) {
            value = n * recursion(n-1);
        }
        return value;
    }
}
```

【程序 23】 题目：有 5 个人坐在一起，问第五个人多少岁？他说比第 4 个人大 2 岁。问第 4 个人岁数，他说比第 3 个人大 2 岁。问第三个人，又说比第 2 人大两岁。问第 2 个人，说比第一个人大两岁。最后问第一个人， he 说是 10 岁。请问第五个人多大？

1.程序分析：利用递归的方法，递归分为回推和递推两个阶段。要想知道第五个人岁数，需知道第四人的岁数，依次类推，推到第一人（10 岁），再往回推。

```
public class Ex23 {

    static int getAge(int n){
        if (n==1){
            return 10;
        }
        return 2 + getAge(n-1);
    }

    public static void main(String[] args) {
        System.out.println("第五个的年龄为:"+getAge(5));
    }
}
```

【程序 24】 题目：给一个不多于 5 位的正整数，要求：一、求它是几位数，二、逆序打印出各位数字。

```
import java.util.Scanner;

public class Ex24 {

    public static void main(String[] args) {
        Ex24 tn = new Ex24();
    }
}
```



```
Scanner s = new Scanner(System.in);
long a = s.nextLong();
if(a < 0 || a > 100000) {
    System.out.println("Error Input, please run this program Again");
    System.exit(0);
}
if(a >= 0 && a <= 9) {
    System.out.println(a + "是一位数");
    System.out.println("按逆序输出是" + '\n' + a);
} else if(a >= 10 && a <= 99) {
    System.out.println(a + "是两位数");
    System.out.println("按逆序输出是" );
    tn.converse(a);
} else if(a >= 100 && a <= 999) {
    System.out.println(a + "是三位数");
    System.out.println("按逆序输出是" );
    tn.converse(a);
} else if(a >= 1000 && a <= 9999) {
    System.out.println(a + "是四位数");
    System.out.println("按逆序输出是" );
    tn.converse(a);
} else if(a >= 10000 && a <= 99999) {
    System.out.println(a + "是五位数");
    System.out.println("按逆序输出是" );
    tn.converse(a);
}
}
}
public void converse(long l) {
    String s = Long.toString(l);
    char[] ch = s.toCharArray();
    for(int i=ch.length-1; i>=0; i--) {
        System.out.print(ch[i]);
    }
}
}
```

【程序 25】 题目：一个 5 位数，判断它是不是回文数。即 12321 是回文数，个位与万位相同，十位与千位相同。

```
import java.util.Scanner;
public class Ex25 {
    static int[] a = new int[5];
    static int[] b = new int[5];
    public static void main(String[] args) {
        boolean is = false;
        Scanner s = new Scanner(System.in);
```

```
long l = s.nextLong();
if (l > 99999 || l < 10000) {
    System.out.println("Input error, please input again!");
    l = s.nextLong();
}
for (int i = 4; i >= 0; i--) {
    a[i] = (int) (l / (long) Math.pow(10, i));
    l = (l % (long) Math.pow(10, i));
}
System.out.println();
for(int i=0,j=0; i<5; i++, j++) {
    b[j] = a[i];
}
for(int i=0,j=4; i<5; i++, j--) {
    if(a[i] != b[j]) {
        is = false;
        break;
    } else {
        is = true;
    }
}
}
if(is == false) {
    System.out.println("is not a Palindrom!");
} else if(is == true) {
    System.out.println("is a Palindrom!");
}
}
}
```

【程序 26】 题目：请输入星期几的第一个字母来判断一下是星期几，如果第一个字母一样，则继续判断第二个字母。

1.程序分析：用情况语句比较好，如果第一个字母一样，则判断用情况语句或 if 语句判断第二个字母。

```
import java.util.Scanner;
public class Ex26 {
    public static void main(String[] args){
        //保存用户输入的第二个字母
        char weekSecond;
        //将 Scanner 类示例化为 input 对象，用于接收用户输入
        Scanner input = new Scanner(System.in);
        //开始提示并接收用户控制台输入
        System.out.print("请输入星期值英文的第一个字母，我来帮您判断是星期几：");
        String letter = input.next();
        //判断用户控制台输入字符串长度是否是一个字母
        if (letter.length() == 1){
            //利用取第一个索引位的字符来实现让 Scanner 接收 char 类型输入
```

```
char weekFirst = letter.charAt(0);
switch (weekFirst){
case 'm':
    //当输入小写字母时，利用 switch 结构特性执行下一个带 break 语句的 case 分支，以实现忽略用户
    控制台输入大小写敏感的功能
    case 'M':
        System.out.println("星期一(Monday)");
        break;
    case 't':
        //当输入小写字母时，利用 switch 结构特性执行下一个带 break 语句的 case 分支，以实现忽略用户
        控制台输入大小写敏感的功能
        case 'T':
            System.out.print("由于星期二(Tuesday)与星期四(Thursday)均以字母 T 开头，故需输入第二个字
            母才能正确判断：");
            letter = input.next();
            //判断用户控制台输入字符串长度是否是一个字母
            if (letter.length() == 1){
                //利用取第一个索引位的字符来实现让 Scanner 接收 char 类型输入
                weekSecond = letter.charAt(0);
                //利用或 ( || ) 运算符来实现忽略用户控制台输入大小写敏感的功能
                if (weekSecond == 'U' || weekSecond == 'u'){
                    System.out.println("星期二(Tuesday)");
                    break;
                }
                //利用或 ( || ) 运算符来实现忽略用户控制台输入大小写敏感的功能
                } else if (weekSecond == 'H' || weekSecond == 'h'){
                    System.out.println("星期四(Thursday)");
                    break;
                }
                //控制台错误提示
                } else{
                    System.out.println("输入错误，不能识别的星期值第二个字母，程序结束！");
                    break;
                }
            } else {
                //控制台错误提示
                System.out.println("输入错误，只能输入一个字母，程序结束！");
                break;
            }
        }
    case 'w':
        //当输入小写字母时，利用 switch 结构特性执行下一个带 break 语句的 case 分支，以实现忽略用户
        控制台输入大小写敏感的功能
        case 'W':
            System.out.println("星期三(Wednesday)");
            break;
        case 'f':
```

//当输入小写字母时，利用 switch 结构特性执行下一个带 break 语句的 case 分支，以实现忽略用户控制台输入大小写敏感的功能

```
case 'f':  
    System.out.println("星期五(Friday)");  
    break;
```

```
case 's':
```

//当输入小写字母时，利用 switch 结构特性执行下一个带 break 语句的 case 分支，以实现忽略用户控制台输入大小写敏感的功能

```
case 'S':
```

System.out.print("由于星期六(Saturday)与星期日(Sunday)均以字母 S 开头，故需输入第二个字母才能正确判断：");

```
letter = input.next();
```

//判断用户控制台输入字符串长度是否是一个字母

```
if (letter.length() == 1){
```

//利用取第一个索引位的字符来实现让 Scanner 接收 char 类型输入

```
weekSecond = letter.charAt(0);
```

//利用或 (||) 运算符来实现忽略用户控制台输入大小写敏感的功能

```
if (weekSecond == 'A' || weekSecond == 'a'){
```

```
    System.out.println("星期六(Saturday)");
```

```
    break;
```

//利用或 (||) 运算符来实现忽略用户控制台输入大小写敏感的功能

```
} else if (weekSecond == 'U' || weekSecond == 'u'){
```

```
    System.out.println("星期日(Sunday)");
```

```
    break;
```

//控制台错误提示

```
} else{
```

```
    System.out.println("输入错误，不能识别的星期值第二个字母，程序结束！");
```

```
    break;
```

```
}
```

```
} else{
```

//控制台错误提示

```
System.out.println("输入错误，只能输入一个字母，程序结束！");
```

```
break;
```

```
}
```

```
default:
```

//控制台错误提示

```
System.out.println("输入错误，不能识别的星期值第一个字母，程序结束！");
```

```
break;
```

```
}
```

```
} else{
```

//控制台错误提示

```
System.out.println("输入错误，只能输入一个字母，程序结束！");
```

```
}
```

```
}
```

```
}
```

【程序 27】 题目：求 100 之内的素数

```
public class Ex27 {  
    public static void main(String args[])  
    {  
        int sum,i;  
        for(sum=2;sum<=100;sum++)  
        {  
            for(i=2;i<=sum/2;i++)  
            {  
                if(sum%i==0)  
                    break;  
            }  
            if(i>sum/2)  
                System.out.println(sum+"是素数");  
        }  
    }  
}
```

【程序 28】 题目：对 10 个数进行排序

1.程序分析：可以利用选择法，即从后 9 个比较过程中，选择一个最小的与第一个元素交换，下次类推，即用第二个元素与后 8 个进行比较，并进行交换。

```
import java.util.Arrays;  
import java.util.Random;  
import java.util.Scanner;  
public class Ex28 {  
    public static void main(String[] args) {  
        int arr[] = new int[11];  
        Random r=new Random();  
        for(int i=0;i<10;i++){  
            arr[i]=r.nextInt(100)+1;//得到 10 个 100 以内的整数  
        }  
        Arrays.sort(arr);  
        for(int i=0;i<arr.length;i++){  
            System.out.print(arr[i]+"\\t");  
        }  
        System.out.print("\\nPlease Input a int number: ");  
        Scanner sc=new Scanner(System.in);  
        arr[10]=sc.nextInt();//输入一个 int 值  
        Arrays.sort(arr);  
        for(int i=0;i<arr.length;i++){  
            System.out.print(arr[i]+"\\t");  
        }  
    }  
}
```

```
}
```

【程序 29】 题目：求一个 3*3 矩阵对角线元素之和

1.程序分析：利用双重 for 循环控制输入二维数组，再将 a[i][i]累加后输出。

```
public class Ex29 {  
    public static void main(String[] args){  
        double sum=0;  
        int array[][]={{1,2,3},{4,5, 6},{7,7,8}};  
        for(int i=0;i<3;i++){  
            for(int j=0;j<3;j++){  
                if(i==j)  
                    sum=sum + array[i][j];  
            }  
            System.out.println( sum);  
        }  
    }  
}
```

【程序 30】 题目：有一个已经排好序的数组。现输入一个数，要求按原来的规律将它插入数组中。

1. 程序分析：首先判断此数是否大于最后一个数，然后再考虑插入中间的数的情况，插入后此元素之后的数，依次后移一个位置。

```
import java.util.Random;  
public class ArraySort {  
    public static void main(String[] args)  
    {  
        int temp=0;  
        int myarr[] = new int[12];  
        Random r=new Random();  
        for(int i=1;i<=10;i++)  
            myarr[i]=r.nextInt(1000);  
        for (int k=1;k<=10;k++)  
            System.out.print(myarr[k]+",");  
        for(int i=1;i<=9;i++)  
            for(int k=i+1;k<=10;k++)  
                if(myarr[i]>myarr[k])  
                {  
                    temp=myarr[i];  
                    myarr[i]=myarr[k];  
                    myarr[k]=temp;  
                }  
        System.out.println("");  
        for (int k=1;k<=10;k++)  
            System.out.print(myarr[k]+",");  
  
        myarr[11]=r.nextInt(1000);  
        for(int k=1;k<=10;k++)  
            if(myarr[k]>myarr[11])  
            {
```

```
        temp=myarr[11];
        for(int j=11;j>=k+1;j--)
            myarr[j]=myarr[j-1];
        myarr[k]=temp;
    }
    System.out.println("");
    for (int k=1;k<=11;k++)
        System.out.print(myarr[k]+",");
}
}
```

【程序 31】 题目：将一个数组逆序输出。

程序分析：用第一个与最后一个交换。

其实，用循环控制变量更简单：

```
for(int k=11;k>=1;k--)
    System.out.print(myarr[k]+",");
```

【程序 32】 题目：取一个整数 a 从右端开始的 4~7 位。

程序分析：可以这样考虑：

(1)先使 a 右移 4 位。

(2)设置一个低 4 位全为 1,其余全为 0 的数。可用 $\sim(\sim 0 < 4)$

(3)将上面二者进行 & 运算。

```
public class Ex32 {
    public static void main(String[] args)
    {
        int a=0;
        long b=18745678;
        a=(int) Math.floor(b % Math.pow(10,7)/Math.pow(10, 3));
        System.out.println(a);
    }
}
```

【程序 33】

题目：打印出杨辉三角形（要求打印出 10 行如下图）

1.程序分析：

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
```

```
public class Ex33 {
    public static void main(String args[]){
        int i,j;
```

```
        int a[][];
        a=new int[8][8];
        for(i=0;i<8;i++){
            a[i][i]=1;
            a[i][0]=1;
        }
        for(i=2;i<8;i++){
            for(j=1;j<=i-1;j++){
                a[i][j]=a[i-1][j-1]+a[i-1][j];
            }
        }
        for(i=0;i<8;i++){
            for(j=0;j<i;j++){
                System.out.printf("  "+a[i][j]);
            }
            System.out.println();
        }
    }
}
```

【程序 34】 题目：输入 3 个数 a,b,c，按大小顺序输出。

1.程序分析：利用指针方法。

```
public class Ex34 {
    public static void main(String[] args)
    {
        int []arrays = {800,56,500};
        for(int i=arrays.length;--i>=0;)
        {
            for(int j=0;j<i;j++)
            {
                if(arrays[j]>arrays[j+1])
                {
                    int temp=arrays[j];
                    arrays[j]=arrays[j+1];
                    arrays[j+1]=temp;
                }
            }
        }
        for(int n=0;n<arrays.length;n++)
            System.out.println(arrays[n]);
    }
}
```

【程序 35】 题目：输入数组，最大的与第一个元素交换，最小的与最后一个元素交换，输出数组。


```
import java.util.*;
public class Ex35 {
    public static void main(String[] args) {
        int i, min, max, n, temp1, temp2;
        int a[];
        System.out.println("输入数组的长度:");
        Scanner keyboard = new Scanner(System.in);
        n = keyboard.nextInt();
        a = new int[n];
        for (i = 0; i < n; i++) {
            System.out.print("输入第" + (i + 1) + "个数据");
            a[i] = keyboard.nextInt();
        }
        //以上是输入整个数组
        max = 0;
        min = 0;
        //设置两个标志,开始都指向第一个数
        for (i = 1; i < n; i++) {
            if (a[i] > a[max])
                max = i; //遍历数组,如果大于 a[max],就把他的数组下标赋给 max
            if (a[i] < a[min])
                min = i; //同上,如果小于 a[min],就把他的数组下标赋给 min
        }
        //以上 for 循环找到最大值和最小值, max 是最大值的下标, min 是最小值的下标
        temp1 = a[0];
        temp2 = a[min]; //这两个 temp 只是为了在交换时使用

        a[0] = a[max];
        a[max] = temp1; //首先交换 a[0]和最大值 a[max]

        if (min != 0) { //如果最小值不是 a[0], 执行下面
            a[min] = a[n - 1];
            a[n - 1] = temp2; //交换 a[min]和 a[n-1]
        } else { //如果最小值是 a[0],执行下面
            a[max] = a[n - 1];
            a[n - 1] = temp1;
        }

        for (i = 0; i < n; i++) { //输出数组
            System.out.print(a[i] + " ");
        }
    }
}
```

【程序 36】 题目：有 n 个整数，使其前面各数顺序向后移 m 个位置，最后 m 个数变成最前面的 m 个

数

【程序 37】

题目：有 n 个人围成一圈，顺序排号。从第一个人开始报数（从 1 到 3 报数），凡报到 3 的人退出圈子，问最后留下的是原来第几号的那位。

```
import java.util.Scanner;

public class Ex37 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int n = s.nextInt();
        boolean[] arr = new boolean[n];
        for(int i=0; i<arr.length; i++) {
            arr[i] = true; //下标为 TRUE 时说明还在圈里
        }
        int leftCount = n;
        int countNum = 0;
        int index = 0;
        while(leftCount > 1) {
            if(arr[index] == true) { //当在圈里时
                countNum++; //报数递加
                if(countNum == 3) { //报道 3 时
                    countNum = 0; //从零开始继续报数
                    arr[index] = false; //此人退出圈子
                    leftCount--; //剩余人数减一
                }
            }
            index++; //每报一次数，下标加一
            if(index == n) { //是循环数数，当下标大于 n 时，说明已经数了一圈，
                index = 0; //将下标设为零重新开始。
            }
        }
        for(int i=0; i<n; i++) {
            if(arr[i] == true) {
                System.out.println(i);
            }
        }
    }
}
```

【程序 38】

题目：写一个函数，求一个字符串的长度，在 main 函数中输入字符串，并输出其长度。

```
import java.util.Scanner;

public class Ex38 {
```

```
public static void main(String [] args)
{
    Scanner s = new Scanner(System.in);
    System.out.println("请输入一个字符串");
    String mys= s.next();
    System.out.println(str_len(mys));
}
public static int str_len(String x)
{
    return x.length();
}
}
```

题目：编写一个函数，输入 n 为偶数时，调用函数求 $1/2+1/4+\dots+1/n$ ，当输入 n 为奇数时，调用函数 $1/1+1/3+\dots+1/n$

【程序 39】

题目：字符串排序。

```
import java.util.*;
public class test{
    public static void main(String[] args)
    {
        ArrayList<String> list=new ArrayList<String>();
        list.add("010101");
        list.add("010003");
        list.add("010201");
        Collections.sort(list);
        for(int i=0;i<list.size();i++){
            System.out.println(list.get(i));
        }
    }
}
```

【程序 40】

题目：海滩上有一堆桃子，五只猴子来分。第一只猴子把这堆桃子凭据分为五份，多了一个，这只猴子把多的一个扔入海中，拿走了一份。第二只猴子把剩下的桃子又平均分成五份，又多了-一个，它同样把多的一个扔入海中，拿走了一份，第三、第四、第五只猴子都是这样做的，问海滩上原来最少有多少个桃子？

```
public class Dg {
    static int ts=0;//桃子总数
    int fs=1;//记录分的次数
```

```
static int hs=5;//猴子数...
int tsscope=5000;//桃子数的取值范围.太大容易溢出.
public int fT(int t){
    if(t==tsscope){
        //当桃子数到了最大的取值范围时取消递归
        System.out.println("结束");
        return 0;
    }
    else{
        if((t-1)%hs==0 && fs <=hs){
            if(fs==hs)
            {
                System.out.println("桃子数 = "+ts + " 时满足分桃条件");
            }
            fs+=1;
            return fT((t-1)/5*4);// 返回猴子拿走一份后的剩下的总数
        }
        else
        {
            //没满足条件
            fs=1;//分的次数重置为 1
            return fT(ts+=1);//桃子数加+1
        }
    }
}

public static void main(String[] args) {
    new Dg().fT(0);
}

}
```

【程序 41】

java 排序算法的比较

```
import java.util.*;
import java.io.*;
public class SortAlgorithm
{
    static Random rand = new Random();
    void bubbleSort(int[] numlist) // 冒泡排序算法
    {
        int temp;
        for(int j=1;j<numlist.length;j++)
```

```
for(int i=0;i<numlist.length-j;i++)
if(numlist[i]>numlist[i+1])
{
temp = numlist[i+1];
numlist[i+1] = numlist[i];
numlist[i] = temp;
}
}
void selectionSort (int[] numlist) //选择排序算法
{
int temp;
for(int i=0;i<numlist.length-1;i++)
for(int j=i+1;j<numlist.length;j++)
if(numlist[i]>numlist[j])
{
temp = numlist[j];
numlist[j] = numlist[i];
numlist[i] = temp;
}
}
void insertSort (int[] numlist) //插入排序算法
{
int temp,in,out;
for(out=1;out<numlist.length;out++)
{
temp=numlist[out];
in=out;
while(in>0 && numlist[in-1]>=temp)
{
numlist[in]=numlist[in-1];
--in;
}
numlist[in]=temp;
}
}
void display (int[] num) // 打印出排序结果
{
for(int i = 0;i<num.length;i++)
System.out.print(num[i] + " ");
System.out.println("");
}
static int pRand(int mod) // 生成随即数组
{
return Math.abs(rand.nextInt())%mod;
}
```

```
}
public static void main(String args[])throws IOException
{
    SortAlgorithm sortAlgorithm = new SortAlgorithm();
    int[] numList = new int[10];
    for(int i = 0;i<numList.length;i++)
        numList[i] = pRand(100); //调用 pRand 方法，把随即生成的数据输入到
    // 数组中
    System.out.println("随即生成的数组是：");
    // 打印出原数组，
    for(int j = 0;j<numList.length;j++)
        System.out.print(numList[j]+ " ");
    System.out.println("");
    long begin = System.currentTimeMillis(); //排序开始时间，调用系统的当前时间
    sortAlgorithm.bubbleSort(numList); //执行冒泡排序
    long end = System.currentTimeMillis(); //排序结束时间，调用系统当前时间
    System.out.println("冒泡排序用时为：" + (end-begin)); //排序用时
    System.out.println("排序后的数组为：");
    sortAlgorithm.display(numList);
    begin = System.currentTimeMillis();
    sortAlgorithm.selectionSort(numList);
    end = System.currentTimeMillis();
    System.out.println("选择排序用时为：" + (end-begin));
    System.out.println("排序后的数组为：");
    sortAlgorithm.display(numList);
    begin = System.currentTimeMillis();
    sortAlgorithm.insertSort(numList);
    end = System.currentTimeMillis();
    System.out.println("插入排序用时为：" + (end-begin));
    System.out.println("排序后的数组为：");
    sortAlgorithm.display(numList);
}
}
```

【程序 42】

题目如下：用 1、2、2、3、4、5 这六个数字，用 java 写一个 main 函数，打印出所有不同的排列，如：512234、412345 等，要求："4"不能在第三位，"3"与"5"不能相连。

```
static int[] bits = new int[] { 1, 2, 3, 4, 5 };
/**
 * @param args
 */
public static void main(String[] args) {
    sort("", bits);
}
private static void sort(String prefix, int[] a) {
```

```
if (a.length == 1) {  
    System.out.println(prefix + a[0]);  
}  
for (int i = 0; i < a.length; i++) {  
    sort(prefix + a, copy(a, i));  
}  
}  
private static int[] copy(int[] a,int index){  
    int[] b = new int[a.length-1];  
    System.arraycopy(a, 0, b, 0, index);  
    System.arraycopy(a, index+1, b, index, a.length-index-1);  
    return b;  
}
```