

【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
 - * 长到第三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？
 - * 这是一个菲波拉契数列问题
- */

```
public class Test1 {  
    public static void main(String[] args) {  
        System.out.println("第1个月的兔子对数： 1");  
        System.out.println("第2个月的兔子对数： 1");  
        int f1 = 1, f2 = 1, f, M = 24;  
        for (int i = 3; i <= M; i++) {  
            f = f2;  
            f2 = f1 + f2;  
            f1 = f;  
            System.out.println("第" + i + "个月的兔子对数： " + f2);  
        }  
    }  
}
```

【程序2】

/**

- * 题目：判断101-200之间有多少个素数，并输出所有素数。
 - * 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 Test2 {  
    public static void main(String[] args) {  
        int count = 0;  
        for (int i = 101; i < 200; i += 2) {  
            boolean b = false;  
            for (int j = 2; j <= Math.sqrt(i); j++) {  
                if (i % j == 0) {  
                    b = false;  
                    break;  
                } else {  
                    b = true;  
                }  
            }  
            if (b == true) {  
                count++;  
                System.out.println(i);  
            }  
        }  
    }  
}
```

```

    }
    System.out.println("素数个数是: " + count);
}
}
}

```

【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 $153=1$ 的三次方+ 5 的三次方+ 3 的三次方。

```

public class Test3 {
    public static void main(String[] args) {
        int b1, b2, b3;
        for (int m = 101; m < 1000; m++) {
            b3 = m / 100;
            b2 = m % 100 / 10;
            b1 = m % 10;
            if ((b3 * b3 * b3 + b2 * b2 * b2 + b1 * b1 * b1) == m) {
                System.out.println(m + "是一个水仙花数");
            }
        }
    }
}

```

【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 n ,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 Test4 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("请键入一个正整数: ");
        int n = s.nextInt();
        int k = 2;
        System.out.print(n + "=");
        while (k <= n) {
            if (k == n) {
                System.out.println(n);
                break;
            }
        }
    }
}

```

```

        } else if (n % k == 0) {
            System.out.print(k + "*");
            n = n / k;
        } else
            k++;
    }
}
}

```

【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。

```

import java.util.*;
public class Test5 {
    public static void main(String[] args) {
        int x;
        char grade;
        Scanner s = new Scanner(System.in);
        System.out.print("请输入一个成绩: ");
        x = s.nextInt();
        grade = x >= 90 ? 'A' : x >= 60 ? 'B' : 'C';
        System.out.println("等级为: " + grade);
    }
}

```

【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。

/**在循环中，只要除数不等于 0，用较大数除以较小的数，将小的一个数作为下一轮循环的大数，取得的余数作为下一轮循环的较小的数，如此循环直到较小的数的值为 0，返回较大的数，此数即为最大公约数，最小公倍数为两数之积除以最大公约数。*/

```

import java.util.*;
public class Test6 {
    public static void main(String[] args) {
        int a, b, m;
        Scanner s = new Scanner(System.in);
        System.out.print("键入一个整数: ");
        a = s.nextInt();
        System.out.print("再键入一个整数: ");
        b = s.nextInt();
        deff cd = new deff();
        m = cd.deff(a, b);
        int n = a * b / m;
        System.out.println("最大公约数: " + m);
        System.out.println("最小公倍数: " + n);
    }
}

```

```

    }
}
class deff {
    public int deff(int x, int y) {
        int t;
        if (x < y) {
            t = x;
            x = y;
            y = t;
        }
        while (y != 0) {
            if (x == y)
                return x;
            else {
                int k = x % y;
                x = y;
                y = k;
            }
        }
        return x;
    }
}

```

【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。

```

public class Test7 {
    public static void main(String[] args) {
        int abcCount = 0; // 英文字母个数
        int spaceCount = 0; // 空格键个数
        int numCount = 0; // 数字个数
        int otherCount = 0; // 其他字符个数
        Scanner scan = new Scanner(System.in);
        System.out.println("请输入一个字符串: ");
        String str = scan.nextLine();
        char[] ch = str.toCharArray();
        for (int i = 0; i < ch.length; i++) {
            if (Character.isLetter(ch[i])) {
                // 判断是否字母
                abcCount++;
            } else if (Character.isDigit(ch[i])) {
                // 判断是否数字
                numCount++;
            } else if (Character.isSpaceChar(ch[i])) {
                // 判断是否空格键
            }
        }
    }
}

```

```

        spaceCount++;
    } else {
        // 以上都不是则认为是其他字符
        otherCount++;
    }
}
System.out.println("字母个数: " + abcCount);
System.out.println("数字个数: " + numCount);
System.out.println("空格个数: " + spaceCount);
System.out.println("其他字符个数: " + otherCount);
}
}

```

【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。

```

import java.util.*;
public class Test8 {
    public static void main(String[] args) {
        long a, b = 0, sum = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("输入数字a的值: ");
        a = s.nextInt();
        System.out.print("输入相加的项数: ");
        int n = s.nextInt();
        int i = 0;
        while (i < n) {
            b = b + a;
            sum = sum + b;
            a = a * 10;
            ++i;
        }
        System.out.println(sum);
    }
}

```

【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 Test9 {
    public static void main(String[] args) {
        System.out.println("1到1000的完数有:");
        for (int i = 1; i < 1000; i++) {
            int t = 0;
            for (int j = 1; j <= i / 2; j++) {

```

```

        if (i % j == 0) {
            t = t + j;
        }
    }
    if (t == i) {
        System.out.println(i + " ");
    }
}
}
}

```

【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 Test10 {
    public static void main(String[] args) {
        double h = 100, s = 100;
        for (int i = 1; i < 10; i++) {
            s = s + h;
            h = h / 2;
        }
        System.out.println("经过路程: " + s);
        System.out.println("反弹高度: " + h / 2);
    }
}

```

【程序 11】

题目：有 1、2、3、4 四个数字，能组成多少个互不相同且无重复数字的三位数？都是多少？

```

public class Test11 {
    public static void main(String[] args) {
        int count = 0;
        for (int x = 1; x < 5; x++) {
            for (int y = 1; y < 5; y++) {
                for (int z = 1; z < 5; z++) {
                    if (x != y && y != z && x != z) {
                        count++;
                        System.out.println(x * 100 + y * 10 + z);
                    }
                }
            }
        }
    }
}

```

```
        System.out.println("共有" + count + "个三位数");  
    }  
}
```

【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，求应发放奖金总数？

```
import java.util.*;  
public class Test12 {  
    public static void main(String[] args) {  
        double x = 0, y = 0;  
        System.out.print("输入当月利润（万）: ");  
        Scanner s = new Scanner(System.in);  
        x = s.nextInt();  
        if (x > 0 && x <= 10) {  
            y = x * 0.1;  
        } else if (x > 10 && x <= 20) {  
            y = 10 * 0.1 + (x - 10) * 0.075;  
        } else if (x > 20 && x <= 40) {  
            y = 10 * 0.1 + 10 * 0.075 + (x - 20) * 0.05;  
        } else if (x > 40 && x <= 60) {  
            y = 10 * 0.1 + 10 * 0.075 + 20 * 0.05 + (x - 40) * 0.03;  
        } else if (x > 60 && x <= 100) {  
            y = 20 * 0.175 + 20 * 0.05 + 20 * 0.03 + (x - 60) * 0.015;  
        } else if (x > 100) {  
            y = 20 * 0.175 + 40 * 0.08 + 40 * 0.015 + (x - 100) * 0.01;  
        }  
        System.out.println("应该提取的奖金是 " + y + "万");  
    }  
}
```

【程序13】

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

```
public class Test13 {  
    public static void main(String[] args) {
```

```

        for (int x = 1; x < 100000; x++) {
            if(Math.sqrt(x+100) % 1 == 0){
                if(Math.sqrt(x+268) % 1 == 0)
                {
                    System.out.println(x+"加100是一个完全平方数，再加168
是一个完全平方数");
                }
            }
        }
    }
}

```

/*按题意循环应该从-100开始（整数包括正整数、负整数、零），这样会多一个满足条件的数-99。

但是我看到大部分人解这道题目时都把题中的“整数”理解成正整数，我也就随大流了。*/

【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？

```

import java.util.*;
public class Test14 {
    public static void main(String[] args) {
        int year, month, day;
        int days = 0;
        int d = 0;
        int e;
        input fynd = new input();
        do {
            e = 0;
            System.out.print("输入年: ");
            year = fynd.input();
            System.out.print("输入月: ");
            month = fynd.input();
            System.out.print("输入天: ");
            day = fynd.input();
            if (year < 0 || month < 0 || month > 12 || day < 0 || day
> 31) {
                System.out.println("输入错误，请重新输入！");
                e = 1;
            }
        } while (e == 1);
        for (int i = 1; i < month; i++) {
            switch (i) {
                case 1:
                case 3:
                case 5:

```



```

        case 7:
        case 8:
        case 10:
        case 12:
            days = 31;
            break;
        case 4:
        case 6:
        case 9:
        case 11:
            days = 30;
            break;
        case 2:
            if ((year % 400 == 0) || (year % 4 == 0 && year % 100 !=
0)) {
                days = 29;
            } else {
                days = 28;
            }
            break;
    }
    d += days;
}
System.out.println(year + "-" + month + "-" + day + "是这年的
第" + (d + day)
    + "天。");
}
}
class input {
    public int input() {
        int value = 0;
        Scanner s = new Scanner(System.in);
        value = s.nextInt();
        return value;
    }
}

```

【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。

```

import java.util.*;
public class Test15 {
    public static void main(String[] args) {
        input fnc = new input();
        int x = 0, y = 0, z = 0;
    }
}

```

```

        System.out.print("输入第一个数字: ");
        x = fnc.input();
        System.out.print("输入第二个数字: ");
        y = fnc.input();
        System.out.print("输入第三个数字: ");
        z = fnc.input();
        if (x > y) {
            int t = x;
            x = y;
            y = t;
        }
        if (x > z) {
            int t = x;
            x = z;
            z = t;
        }
        if (y > z) {
            int t = y;
            y = z;
            z = t;
        }
        System.out.println("三个数字由小到大排列为: " + x + " " + y + " " + z);
    }
}

class input {
    public int input() {
        int value = 0;
        Scanner s = new Scanner(System.in);
        value = s.nextInt();
        return value;
    }
}

```

【程序 16】

题目: 输出 9*9 口诀。

```

public class Test16 {
    public static void main(String[] args) {
        for (int i = 1; i < 10; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print(j + "*" + i + "=" + j * i + " ");
                if (j * i < 10) {
                    System.out.print(" ");
                }
            }
            System.out.println();
        }
    }
}

```

```

        }
    }
    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。

```

public class Test17 {
    public static void main(String[] args) {
        int x = 1;
        for (int i = 2; i <= 10; i++) {
            x = (x + 1) * 2;
        }
        System.out.println("猴子第一天摘了 " + x + " 个桃子");
    }
}

```

【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，请编程序找出三队赛手的名单。

```

public class Test18 {
    static char[] m = { 'a', 'b', 'c' };
    static char[] n = { 'x', 'y', 'z' };
    public static void main(String[] args) {
        for (int i = 0; i < m.length; i++) {
            for (int j = 0; j < n.length; j++) {
                if (m[i] == 'a' && n[j] == 'x') {
                    continue;
                } else if (m[i] == 'a' && n[j] == 'y') {
                    continue;
                } else if ((m[i] == 'c' && n[j] == 'x')
                    || (m[i] == 'c' && n[j] == 'z')) {
                    continue;
                } else if ((m[i] == 'b' && n[j] == 'z')
                    || (m[i] == 'b' && n[j] == 'y')) {
                    continue;
                } else
            }
        }
    }
}

```

```

        System.out.println(m[i] + " vs " + n[j]);
    }
}
}
}

```

【程序 19】

题目：打印出如下图案（菱形）

```

    *
   ***
  *****
 *

```

```

public class Test19 {
    public static void main(String[] args) {
        int H = 7, W = 7; // 高和宽必须是相等的奇数
        for (int i = 0; i < (H + 1) / 2; i++) {
            for (int j = 0; j < W / 2 - i; j++) {
                System.out.print(" ");
            }
            for (int k = 1; k < (i + 1) * 2; k++) {
                System.out.print('*');
            }
            System.out.println();
        }
        for (int i = 1; i <= H / 2; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print(" ");
            }
            for (int k = 1; k <= W - 2 * i; k++) {
                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。

```

public class Test20 {
    public static void main(String[] args) {
        int x = 2, y = 1, t;
        double sum = 0;
    }
}

```

```

        for (int i = 1; i <= 20; i++) {
            sum = sum + (double) x / y;
            t = y;
            y = x;
            x = y + t;
        }
        System.out.println("前20项相加之和是: " + sum);
    }
}

```

【程序 21】

题目: A.求 $1+2!+3!+\dots+20!$ 的和 B.用递归求 $1+\dots+1/n!$ 的总和 或者求 $n!$

```

public class Test21 {
    public static void main(String[] args) {
        long sum = 0;
        long fac = 1;
        for (int i = 1; i <= 20; i++) {
            fac = fac * i;
            sum += fac;
        }
        System.out.println(sum);
    }
}

public class DiGui {
    public static void main(String[] args) {
        System.out.println("Result:" + jiecheng1(10));
        System.out.println("Result:" + jiecheng(5));
    }
    /**1+...+1/n!的总和*/
    public static double jiecheng1(double sum) {
        if (sum == 1) {
            return 1;
        } else {
            return jiecheng1(sum - 1) + 1 / sum;
        }
    }
    /**
     * 一个静态的方法
     * n!的阶乘
     */
    public static int jiecheng(int x) {
        if (x == 0) {
            return 1;
        } else {
            return x * jiecheng(x - 1);
        }
    }
}

```

```
    }  
    }  
}
```

【程序 22】

题目：利用递归方法求 5!。

```
public class Test22 {  
    public static void main(String[] args) {  
        int n = 5;  
        rec fr = new rec();  
        System.out.println(n + "! = " + fr.rec(n));  
    }  
}  
  
class rec {  
    public long rec(int n) {  
        long value = 0;  
        if (n == 1) {  
            value = 1;  
        } else {  
            value = n * rec(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？

```
public class Test23 {  
    public static void main(String[] args) {  
        int age = 10;  
        for (int i = 2; i <= 5; i++) {  
            age = age + 2;  
        }  
        System.out.println(age);  
    }  
}
```

【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。
//使用了长整型最多输入18位

```
import java.util.*;
```

```

public class Test24 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("请输入一个正整数: ");
        long a = s.nextLong();
        String ss = Long.toString(a);
        char[] ch = ss.toCharArray();
        int j = ch.length;
        System.out.println(a + "是一个" + j + "位数。");
        System.out.print("按逆序输出是: ");
        for (int i = j - 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.*;
public class Test25 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int a;
        do {
            System.out.print("请输入一个5位正整数: ");
            a = s.nextInt();
        } while (a < 10000 || a > 99999);
        String ss = String.valueOf(a);
        char[] ch = ss.toCharArray();
        if (ch[0] == ch[4] && ch[1] == ch[3]) {
            System.out.println("这是一个回文数");
        } else {
            System.out.println("这不是一个回文数");
        }
    }
}
//这个更好，不限位数
import java.util.*;
public class Test25_1 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        boolean is = true;
    }
}

```

```

System.out.print("请输入一个正整数: ");
long a = s.nextLong();
String ss = Long.toString(a);
char[] ch = ss.toCharArray();
int j = ch.length;
for (int i = 0; i < j / 2; i++) {
    if (ch[i] != ch[j - i - 1]) {
        is = false;
    }
}
if (is == true) {
    System.out.println("这是一个回文数");
} else {
    System.out.println("这不是一个回文数");
}
}
}

```

【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。

```

import java.util.*;
public class Test26 {
    public static void main(String[] args) {
        getChar tw = new getChar();
        System.out.println("请输入星期的第一个大写字母: ");
        char ch = tw.getChar();
        switch (ch) {
            case 'M':
                System.out.println("Monday");
                break;
            case 'W':
                System.out.println("Wednesday");
                break;
            case 'F':
                System.out.println("Friday");
                break;
            case 'T': {
                System.out.println("请输入星期的第二个字母: ");
                char ch2 = tw.getChar();
                if (ch2 == 'U') {
                    System.out.println("Tuesday");
                } else if (ch2 == 'H') {
                    System.out.println("Thursday");
                }
            }
        }
    }
}

```



```

        } else {
            System.out.println("无此写法! ");
        }
    }
    ;
    break;
case 'S': {
    System.out.println("请输入星期的第二个字母: ");
    char ch2 = tw.getChar();
    if (ch2 == 'U') {
        System.out.println("Sunday");
    } else if (ch2 == 'A') {
        System.out.println("Saturday");
    } else {
        System.out.println("无此写法! ");
    }
}
    break;
default:
    System.out.println("无此写法! ");
}
}
}

class getChar {
    public char getChar() {
        Scanner s = new Scanner(System.in);
        String str = s.nextLine();
        char ch = str.charAt(0);
        if (ch < 'A' || ch > 'Z') {
            System.out.println("输入错误, 请重新输入");
            ch = getChar();
        }
        return ch;
    }
}
}

```

【程序27】

题目: 求100之内的素数

//使用除sqrt(n)的方法求出的素数不包括2和3

```

public class Test27 {
    public static void main(String[] args) {
        boolean b = false;
        System.out.print(2 + " ");
    }
}

```

```

        System.out.print(3 + " ");
        for (int i = 3; i < 100; i += 2) {
            for (int j = 2; j <= Math.sqrt(i); j++) {
                if (i % j == 0) {
                    b = false;
                    break;
                } else {
                    b = true;
                }
            }
            if (b == true) {
                System.out.print(i + " ");
            }
        }
    }
}

// 该程序使用除1位素数得2位方法，运行效率高通用性差
public class Test27_1 {
    public static void main(String[] args) {
        int[] a = new int[] { 2, 3, 5, 7 };
        for (int j = 0; j < 4; j++)
            System.out.print(a[j] + " ");
        boolean b = false;
        for (int i = 11; i < 100; i += 2) {
            for (int j = 0; j < 4; j++) {
                if (i % a[j] == 0) {
                    b = false;
                    break;
                } else {
                    b = true;
                }
            }
            if (b == true) {
                System.out.print(i + " ");
            }
        }
    }
}

```

【程序28】

题目：对10个数进行排序

```
import java.util.*;
```

```

public class Test28 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int[] a = new int[10];
        System.out.println("请输入10个整数: ");
        for (int i = 0; i < 10; i++) {
            a[i] = s.nextInt();
        }
        for (int i = 0; i < 10; i++) {
            for (int j = i + 1; j < 10; j++) {
                if (a[i] > a[j]) {
                    int t = a[i];
                    a[i] = a[j];
                    a[j] = t;
                }
            }
        }
        for (int i = 0; i < 10; i++) {
            System.out.print(a[i] + " ");
        }
    }
}

```

【程序29】

题目：求一个3*3矩阵对角线元素之和

```

import java.util.*;
public class Test29 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int[][] a = new int[3][3];
        System.out.println("请输入9个整数: ");
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                a[i][j] = s.nextInt();
            }
        }
        System.out.println("输入的3 * 3 矩阵是:");
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                System.out.print(a[i][j] + " ");
            }
            System.out.println();
        }
        int sum = 0;
    }
}

```

```

        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                if (i == j) {
                    sum += a[i][j];
                }
            }
        }
        System.out.println("对角线之和是: " + 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。
 //此程序不好，没有使用折半查找插入

```

import java.util.*;
public class Test30 {
    public static void main(String[] args) {
        int[] a = new int[] { 1, 2, 6, 14, 25, 36, 37, 55 };
        int[] b = new int[a.length + 1];
        int t1 = 0, t2 = 0;
        int i = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("请输入一个整数: ");
        int num = s.nextInt();
        if (num >= a[a.length - 1]) {
            b[b.length - 1] = num;
            for (i = 0; i < a.length; i++) {
                b[i] = a[i];
            }
        } else {
            for (i = 0; i < a.length; i++) {
                if (num >= a[i]) {
                    b[i] = a[i];
                } else {
                    b[i] = num;
                    break;
                }
            }
            for (int j = i + 1; j < b.length; j++) {
                b[j] = a[j - 1];
            }
        }
        for (i = 0; i < b.length; i++) {

```

```
        System.out.print(b[i] + " ");
    }
}
}
```

【程序31】

题目：将一个数组逆序输出。

```
import java.util.*;
public class Test31 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int a[] = new int[20];
        System.out.println("请输入多个正整数（输入-1表示结束）：");
        int i = 0, j;
        do {
            a[i] = s.nextInt();
            i++;
        } while (a[i - 1] != -1);
        System.out.println("你输入的数组为：");
        for (j = 0; j < i - 1; j++) {
            System.out.print(a[j] + " ");
        }
        System.out.println("\n数组逆序输出为：");
        for (j = i - 2; j >= 0; j = j - 1) {
            System.out.print(a[j] + " ");
        }
    }
}
```

【程序32】

题目：取一个整数a从右端开始的4~7位。

```
import java.util.*;
public class Test32 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("请输入一个7位以上的正整数：");
        long a = s.nextLong();
        String ss = Long.toString(a);
        char[] ch = ss.toCharArray();
        int j = ch.length;
        if (j < 7) {
            System.out.println("输入错误！");
        }
    }
}
```

```

    } else {
        System.out.println("截取从右端开始的4~7位是: " + ch[j - 7] +
            ch[j - 6]
                + ch[j - 5] + ch[j - 4]);
    }
}
}
}

```

【程序33】

题目：打印出杨辉三角形（要求打印出10行如下图）

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
.....

```

```

public class Test33 {
    public static void main(String[] args) {
        int[][] a = new int[10][10];
        for (int i = 0; i < 10; i++) {
            a[i][i] = 1;
            a[i][0] = 1;
        }
        for (int i = 2; i < 10; i++) {
            for (int j = 1; j < i; j++) {
                a[i][j] = a[i - 1][j - 1] + a[i - 1][j];
            }
        }
        for (int i = 0; i < 10; i++) {
            for (int k = 0; k < 2 * (10 - i) - 1; k++) {
                System.out.print(" ");
            }
            for (int j = 0; j <= i; j++) {
                System.out.print(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。

```

import java.util.Scanner;
public class Test34 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.println("请输入3个整数: ");
        int a = s.nextInt();
        int b = s.nextInt();
        int c = s.nextInt();
        if (a < b) {
            int t = a;
            a = b;
            b = t;
        }
        if (a < c) {
            int t = a;
            a = c;
            c = t;
        }
        if (b < c) {
            int t = b;
            b = c;
            c = t;
        }
        System.out.println("从大到小的顺序输出:");
        System.out.println(a + " " + b + " " + c);
    }
}

```

【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 Test35 {
    public static void main(String[] args) {
        int N = 8;
        int[] a = new int[N];
        Scanner s = new Scanner(System.in);
        int idx1 = 0, idx2 = 0;
        System.out.println("请输入8个整数: ");
        for (int i = 0; i < N; i++) {
            a[i] = s.nextInt();
        }
        System.out.println("你输入的数组为: ");
        for (int i = 0; i < N; i++) {

```

```

        System.out.print(a[i] + " ");
    }
    int max = a[0], min = a[0];
    for (int i = 0; i < N; i++) {
        if (a[i] > max) {
            max = a[i];
            idx1 = i;
        }
        if (a[i] < min) {
            min = a[i];
            idx2 = i;
        }
    }
    if (idx1 != 0) {
        int temp = a[0];
        a[0] = a[idx1];
        a[idx1] = temp;
    }
    if (idx2 != N - 1) {
        int temp = a[N - 1];
        a[N - 1] = a[idx2];
        a[idx2] = temp;
    }
    System.out.println("\n交换后的数组为: ");
    for (int i = 0; i < N; i++) {
        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

```

import java.util.Scanner;
public class Test36 {
    public static void main(String[] args) {
        int N = 10;
        int[] a = new int[N];
        Scanner s = new Scanner(System.in);
        System.out.println("请输入10个整数: ");
        for (int i = 0; i < N; i++) {
            a[i] = s.nextInt();
        }
        System.out.print("你输入的数组为: ");
    }
}

```



```

    for (int i = 0; i < N; i++) {
        System.out.print(a[i] + " ");
    }
    System.out.print("\n请输入向后移动的位数: ");
    int m = s.nextInt();
    int[] b = new int[m];
    for (int i = 0; i < m; i++) {
        b[i] = a[N - m + i];
    }
    for (int i = N - 1; i >= m; i--) {
        a[i] = a[i - m];
    }
    for (int i = 0; i < m; i++) {
        a[i] = b[i];
    }
    System.out.print("位移后的数组是: ");
    for (int i = 0; i < N; i++) {
        System.out.print(a[i] + " ");
    }
}
}

```

【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 Test37 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("请输入排成一圈的人数: ");
        int n = s.nextInt();
        boolean[] arr = new boolean[n];
        for (int i = 0; i < arr.length; i++) {
            arr[i] = true;
        }
        int leftCount = n;
        int countNum = 0;
        int index = 0;
        while (leftCount > 1) {
            if (arr[index] == true) {
                countNum++;
                if (countNum == 3) {
                    countNum = 0;
                }
            }
            index++;
            if (index == arr.length) {
                index = 0;
            }
            leftCount--;
        }
    }
}

```

```

        arr[index] = false;
        leftCount--;
    }
}
index++;
if (index == n) {
    index = 0;
}
}
for (int i = 0; i < n; i++) {
    if (arr[i] == true) {
        System.out.println("原排在第" + (i + 1) + "位的人留下了。");
    }
}
}
}
}

```

【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。

/*.....

*.....题目意思似乎不能用length()函数 */

```

import java.util.*;
public class Test38 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.println("请输入一个字符串: ");
        String str = s.nextLine();
        System.out.println("字符串的长度是: " + str.length());
    }
}

```

【程序39】

题目：编写一个函数，输入n为偶数时，调用函数求 $1/2+1/4+...+1/n$ ，当输入n为奇数时，调用函数 $1/1+1/3+...+1/n$ (利用指针函数)

//没有利用指针函数

```

public class Test39 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("请输入一个正整数 n= ");
        int n = s.nextInt();
    }
}

```

```

        System.out.println("相应数列的和为: " + sum(n));
    }
    public static double sum(int n) {
        double res = 0;
        if (n % 2 == 0) {
            for (int i = 2; i <= n; i += 2) {
                res += (double) 1 / i;
            }
        } else {
            for (int i = 1; i <= n; i += 2) {
                res += (double) 1 / i;
            }
        }
        return res;
    }
}

```

【程序40】

题目：字符串排序。

```

public class Test40 {
    public static void main(String[] args) {
        int N = 5;
        String temp = null;
        String[] s = new String[N];
        s[0] = "matter";
        s[1] = "state";
        s[2] = "solid";
        s[3] = "liquid";
        s[4] = "gas";
        for (int i = 0; i < N; i++) {
            for (int j = i + 1; j < N; j++) {
                if (compare(s[i], s[j]) == false) {
                    temp = s[i];
                    s[i] = s[j];
                    s[j] = temp;
                }
            }
        }
        for (int i = 0; i < N; i++) {
            System.out.println(s[i]);
        }
    }
    static boolean compare(String s1, String s2) {

```

```

        boolean result = true;
        for (int i = 0; i < s1.length() && i < s2.length(); i++) {
            if (s1.charAt(i) > s2.charAt(i)) {
                result = false;
                break;
            } else if (s1.charAt(i) < s2.charAt(i)) {
                result = true;
                break;
            } else {
                if (s1.length() < s2.length()) {
                    result = true;
                } else {
                    result = false;
                }
            }
        }
        return result;
    }
}

```

【程序41】

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 Test41 {
    public static void main(String[] args) {
        int i, m, j = 0, k, count;
        for (i = 4; i < 10000; i += 4) {
            count = 0;
            m = i;
            for (k = 0; k < 5; k++) {
                j = i / 4 * 5 + 1;
                i = j;
                if (j % 4 == 0)
                    count++;
                else
                    break;
            }
            i = m;
            if (count == 4) {
                System.out.println("原有桃子 " + j + " 个");
                break;
            }
        }
    }
}

```

```

    }
}
}
}

```

【程序42】

题目： $809*??=800*??+9*??+1$ 其中??代表的两位数, $8*??$ 的结果为两位数, $9*??$ 的结果为3位数。求??代表的两位数, 及 $809*??$ 后的结果。

//题目错了! $809x=800x+9x+1$ 这样的方程无解。去掉那个1就有解了。

```

public class Test42 {
    public static void main(String[] args) {
        int a = 809, b, i;
        for (i = 10; i < 13; i++) {
            b = i * a;
            if (8 * i < 100 && 9 * i >= 100)
                System.out.println("809*" + i + "=" + "800*" + i + "+"
+ "9*"
                                + i + "=" + b);
        }
    }
}

```

【程序43】

题目：求0—7所能组成的奇数个数。

//组成1位数是4个。

//组成2位数是7*4个。

//组成3位数是7*8*4个。

//组成4位数是7*8*8*4个。

//.....

```

public class Test43 {
    public static void main(String[] args) {
        int sum = 4;
        int j;
        System.out.println("组成1位数是 " + sum + " 个");
        sum = sum * 7;
        System.out.println("组成2位数是 " + sum + " 个");
        for (j = 3; j <= 9; j++) {
            sum = sum * 8;
            System.out.println("组成" + j + "位数是 " + sum + " 个");
        }
    }
}

```

【程序44】

题目：一个偶数总能表示为两个素数之和。

//由于用除sqrt(n)的方法求出的素数不包括2和3，

//因此在判断是否是素数程序中人为添加了一个3。

```
import java.util.*;
public class Test44 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int n, i;
        do {
            System.out.print("请输入一个大于等于6的偶数: ");
            n = s.nextInt();
        } while (n < 6 || n % 2 != 0); // 判断输入是否是>=6偶数,不是,重新
输入
        fun fc = new fun();
        for (i = 2; i <= n / 2; i++) {
            if ((fc.fun(i)) == 1 && (fc.fun(n - i) == 1)) {
                int j = n - i;
                System.out.println(n + " = " + i + " + " + j);
            } // 输出所有可能的素数对
        }
    }
}

class fun {
    public int fun(int a) // 判断是否是素数的函数
    {
        int i, flag = 0;
        if (a == 3) {
            flag = 1;
            return (flag);
        }
        for (i = 2; i <= Math.sqrt(a); i++) {
            if (a % i == 0) {
                flag = 0;
                break;
            } else
                flag = 1;
        }
        return (flag);
    }
}

// 不是素数,返回0,是素数,返回1}}
```

```
}
```

//解法二

```
import java.util.*;
public class Test4S {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int n;
        do {
            System.out.print("请输入一个大于等于6的偶数: ");
            n = s.nextInt();
        } while (n < 6 || n % 2 != 0); // 判断输入是否是>=6偶数,不是,重新
输入
        for (int i = 3; i <= n / 2; i += 2) {
            if (fun(i) && fun(n - i)) {
                System.out.println(n + " = " + i + " + " + (n - i));
            } // 输出所有可能的素数对
        }
    }
    static boolean fun(int a) {
        // 判断是否是素数的函数
        boolean flag = false;
        if (a == 3) {
            flag = true;
            return (flag);
        }
        for (int i = 2; i <= Math.sqrt(a); i++) {
            if (a % i == 0) {
                flag = false;
                break;
            } else
                flag = true;
        }
        return (flag);
    }
}
```

【程序45】

题目：判断一个素数能被几个9整除

//题目错了吧？能被9整除的就不是素数了！所以改成整数了。

```
import java.util.*;
public class Test45 {
```

```

public static void main(String[] args) {
    Scanner s = new Scanner(System.in);
    System.out.print("请输入一个整数: ");
    int num = s.nextInt();
    int tmp = num;
    int count = 0;
    for (int i = 0; tmp % 9 == 0;) {
        tmp = tmp / 9;
        count++;
    }
    System.out.println(num + " 能够被 " + count + " 个9整除。");
}

```

【程序46】

题目：两个字符串连接程序

```

import java.util.Scanner;
public class Test46 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("请输入一个字符串: ");
        String str1 = s.nextLine();
        System.out.print("请再输入一个字符串: ");
        String str2 = s.nextLine();
        String str = str1 + str2;
        System.out.println("连接后的字符串是" + str);
    }
}

```

【程序47】

题目：读取7个数（1—50）的整数值，每读取一个值，程序打印出该值个数的*。

```

public class Test47 {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int n = 1, num;
        while (n <= 7) {
            do {
                System.out.print("请输入一个1--50之间的整数: ");
                num = s.nextInt();
            } while (num < 1 || num > 50);
            for (int i = 1; i <= num; i++) {
                System.out.print("*");
            }
            n++;
        }
    }
}

```



```

        }
        System.out.println();
        n++;
    }
}
}

```

【程序48】

题目：某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下：每位数字都加上5,然后用和除以10的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。

```

import java.util.*;
public class Test48 {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        int num = 0, temp;
        do {
            System.out.print("请输入一个4位正整数: ");
            num = s.nextInt();
        } while (num < 1000 || num > 9999);
        int a[] = new int[4];
        a[0] = num / 1000; // 取千位的数字
        a[1] = (num / 100) % 10; // 取百位的数字
        a[2] = (num / 10) % 10; // 取十位的数字
        a[3] = num % 10; // 取个位的数字
        for (int j = 0; j < 4; j++) {
            a[j] += 5;
            a[j] %= 10;
        }
        for (int j = 0; j <= 1; j++) {
            temp = a[j];
            a[j] = a[3 - j];
            a[3 - j] = temp;
        }
        System.out.print("加密后的数字为: ");
        for (int j = 0; j < 4; j++)
            System.out.print(a[j]);
    }
}

```

【程序49】

题目：计算字符串中子串出现的次数

```

import java.util.Scanner;
public class Test49 {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        System.out.print("请输入字符串: ");
        String str1 = s.nextLine();
        System.out.print("请输入子串: ");
        String str2 = s.nextLine();
        int count = 0;
        if (str1.equals("") || str2.equals("")) {
            System.out.println("你没有输入字符串或子串,无法比较!");
            System.exit(0);
        } else {
            {
                for(int i=0;i<=str1.length()-str2.length();i++){
                    if(str2.equals(str1.substring(i, str2.length()+i)))
                        //这种比法有问题,会把"aaa"看成有2个"aa"子串。
                        count++;
                } System.out.println("子串在字符串中出现: "+count+" 次");
            }
        }
    }
}

```

【程序 50】

题目: 有五个学生, 每个学生有 3 门课的成绩, 从键盘输入以上数据 (包括学生号, 姓名, 三门课成绩), 计算出平均成绩, 把原有的数据和计算出的平均分数存放在磁盘文件 "stud" 中.

```

import java.io.*;
import java.util.*;
public class Test50 {
    public static void main(String[] args) throws IOException {
        Scanner ss = new Scanner(System.in);
        String[][] a = new String[5][6];
        for (int i = 1; i < 6; i++) {
            System.out.print("请输入第" + i + "个学生的学号: ");
            a[i - 1][0] = ss.nextLine();
            System.out.print("请输入第" + i + "个学生的姓名: ");
            a[i - 1][1] = ss.nextLine();
            for (int j = 1; j < 4; j++) {
                System.out.print("请输入该学生的第" + j + "个成绩: ");
                a[i - 1][j + 1] = ss.nextLine();
            }
        }
    }
}

```

```

    }
    System.out.println("\n");
} // 以下 计算平均分
float avg;
int sum;
for (int i = 0; i < 5; i++) {
    sum = 0;
    for (int j = 2; j < 5; j++) {
        sum = sum + Integer.parseInt(a[i][j]);
    }
    avg = (float) sum / 3;
    a[i][5] = String.valueOf(avg);
}
// 以下写磁盘文件
String s1;
File f = new File("C:\\stud");
if (f.exists()) {
    System.out.println("文件存在");
} else {
    System.out.println("文件不存在，正在创建文件");
    f.createNewFile(); // 不存在则创建
}
BufferedWriter output = new BufferedWriter(new FileWriter(f));
for (int i = 0; i < 5; i++) {
    for (int j = 0; j < 6; j++) {
        s1 = a[i][j] + "\r\n";
        output.write(s1);
    }
}
output.close();
System.out.println("数据已写入c盘文件stud中！");
}
}

```

【程序 51】

用程序处理一个字符串倒序输出如：ABC----->输出结果为 CBA

```

public class DaoXu {
    /**
     * @param args
     */
    public static void main(String[] args) {

```

```

        System.out.println("-----方法一-----");
        str1("你爱我");
        System.out.println("-----方法二-----");
        System.out.println(str2("你爱我"));
        System.out.println("-----方法三-----");
        System.out.println(str3("你爱我"));
        System.out.println("-----方法四-----");
        System.out.println(str4("你爱我"));
        System.out.println("-----方法五-----");
        str5("你爱我");
    }
    /**
     * charAt
     */
    public static void str1(String s) {
        String str = " ";
        for (int i = s.length() - 1; i >= 0; i--) {
            str += s.charAt(i);
        }
        System.out.println(str);
    }
    /**
     * charAt
     */
    public static String str2(String s) {
        String str = " ";
        for (int i = s.length() - 1; i >= 0; i--) {
            str += s.charAt(i);
        }
        return str;
    }
    /**
     * StringBuffer append
     */
    public static String str3(String s) {
        StringBuffer sb = new StringBuffer("");
        for (int i = s.length() - 1; i >= 0; i--) {
            sb.append(s.charAt(i));
        }
        return sb.toString();
    }
    /**

```

* substring(beginIndex,endIndex) beginIndex是提取开始位置, 而且是从0开始,

```

    * endIndex是提取结束位置
    * Java是基于字符串的字符首字为1出来
    */
    public static String str4(String st) {
        String str = "";
        int s, a;
        for (int i = 0; i < st.length(); i++) {
            s = st.length() - i - 1;
            a = st.length() - i;
            str = str + st.substring(s, a);
        }
        return str;
    }
    /**
    * 面试最好不要用.
    * @param str
    */
    public static void str5(String str) {
        StringBuffer sb = new StringBuffer(str);
        System.out.println(sb.reverse().toString());
    }
}

```

【程序 52】

请你写一个程序打印出昨天的当前时间

```

import java.text.SimpleDateFormat;
import java.util.Date;
public class MyDate {
    /**
    * 请你写一个程序打印出昨天的当前时间..
    * @param args
    */
    public static void main(String[] args) {
        String str = "";
        // 创建一个时间格式对象
        SimpleDateFormat sd = new SimpleDateFormat("yyyy-MM-dd
HH:mm:ss");
        // 创建一个时间对象
        Date d = new Date();
        // 设置一个时间值并且取到当前时间再减一天
        d.setDate(d.getDate() - 1);
        str = sd.format(d);
        System.out.println("打印出昨天的当前时间:" + str);
    }
}

```

```
}
```

【程序 53】

请你写一个程序生成 9 个不重复的随机数

```
public class Rand {  
    public static void main(String[] args) {  
        Random rand = new Random(System.nanoTime());  
        Set<Integer> set = new HashSet<Integer>();  
        while (set.size() < 9) {  
            set.add(rand.nextInt(100));  
        }  
        System.out.println(set);  
    }  
}
```

【程序 53】求 0 到 1000 的水仙花数有哪些, 请你用程序打印出来

```
public static void main(String[] args) {  
    for (int i = 0; i < 1000; i++) {  
        int a = i / 100;  
        int b = i / 10 % 10;  
        int c = i % 10;  
        if (Math.pow(a, 3) + Math.pow(b, 3) + Math.pow(c, 3) == i)  
            System.out.println(i + "是水仙花数");  
    }  
}
```

【程序 54】单例模式

```
public class Singleton {  
    /**  
     * 一个构造方法  
     */  
    public Singleton() {  
    }  
    /**  
     * 方法一  
     */  
    public static Singleton singleton = null;  
    public static Singleton getInstance() {  
        if (singleton == null) {  
            singleton = new Singleton();  
        }  
        return singleton;  
    }  
    /**
```

```

* 方法二
*/
public static Singleton sing = new Singleton();

synchronized public static Singleton getInst() {
    return sing;
}
}

```

【程序 55】Java 的==与 equals 的区别

“==”比较两个变量本身的值，即两个对象在内存中的首地址。

“equals()”比较字符串中所包含的内容是否相同

```

package com.jse.org.core.eq;
public class Test1 {
    public static void main(String[] args) {
        String a = "abc";
        String b = "abc";
        //这个是创建了两个对象,一个是匿名,一个是new实例创建
        String c = new String("abC");
        if (a == b) {
            System.out.println("==是用来对比地址是否相等!");
        }
        if(a.equals(b)) {
            if(b.equalsIgnoreCase(c)) //忽略大小写
                System.out.println("equals用来对比空间内容是否相等!");
        }
    }
}

```

1 字符串的判断

```

public static void main(String[] args) {
    String a = "123";
    String b = "123";
    String e = "321";
    String c = "d";
    String f=new String("abc");
    System.out.println(a.equals(b)); //true
    System.out.println(a == b); //true
    System.out.println((a+c)==(c+b)); //false
    System.out.println(e == b); //false
    System.out.println(e.equals(b)); //false
    System.out.println(b.intern()==a.intern()); //true
    System.out.println(f.equals(a)); //false
    System.out.println(f=="abc"); //false
}

```

```
}
```

【程序 56】如何将 a 乘以 b 的结果,6.875000000000001 改为 6.875

```
public class DoubleTest {
    /**
     * 如何将a乘以b的结果,6.875000000000001改为 6.875
     *
     * @param args
     */
    public static void main(String[] args) {
        double a = 12.5;
        double b = 0.55;
        System.out.println(a * b);
        // .3f是 向后面的小数点取三位数printf在C++使用比较多
        System.out.printf("%.3f", a * b);
    }
}
```

【程序57】

请你编写一个截取字符串的函数,输入为一个字符串和字节数,输出为按字节截取的字符串,但要保证汉字不被截取半个,如"我ABC",4,应该截取"我AB",输入"我ABC汉DEF",6,应该输出"我ABC",而不是"我ABC+汉的半个".

```
public class JieziJie {
    /**
     * 请你编写一个截取字符串的函数,输入为一个字符串和字节数,输出为按字节截取的字符串,但要保证汉字不被截取半个,
     * 如"我ABC",4,应该截取"我AB",输入"我ABC汉DEF",6,应该输出"我ABC",而不是"我ABC+汉的半个".
     */
    public static void main(String[] args) throws Exception {
        @SuppressWarnings("unused")
        String str = "我a爱中华abc我爱你的def";
        String str1 = "我ABC汉";
        int num = trimGBK(str1.getBytes("GBK"), 5);
        System.out.println(str1.substring(0, num));
    }

    public static int trimGBK(byte[] buf, int n) {
        int num = 0;
        boolean bChineseFirstHalf = false;
        for (int i = 0; i < n; i++) {
            if (buf[i] < 0 && !bChineseFirstHalf) {
                bChineseFirstHalf = true;
            } else {
                num++;
            }
        }
    }
}
```



```

        bChineseFirstHalf = false;
    }
}
return num;
}
}

```

【程序58】 移位运算符的使用

```

public class SuanfaFu {
    /**
     * Author:jilongliang
     * 移位运算符
     * 移位运算符的面向对象也是二进制的"位".可以单独用移位运算符来处理 int 型数据.它主要包括: 左移位运算符 (<<)
     * "有符号"右移位运算符 (>>)、"无符号"右移位运算符 (>>>)
     *
     * 1) 左移位运算符
     * 左移位运算符,用符号"<<"表示.它是将运算符左边的对象向左移动运算符右边指定的位数(在低位补0).
     *
     * 2) "有符号"右移运算符
     * "有符号"右移运算符
     * ,用符号">>"表示.它是将运算符左边的运算对象向右移动运算符右侧指定的位数.它使用了"符号扩展"机制,也就是说,如果值为正,在高位补0
     * ,若为负,则在高位补1.
     *
     * 3) "无符号"右移运算符
     * "无符号"右移运算符,用符号">>>"表示.它同"有符号"右移运算符的移动规则是一样的,惟一的区别就是
     * : "无符号"右移运算符,它采用了"零扩展",也就是说,无论值为正负,都在高位补0.
     * 请你写出x y z 的值
     * @param args
     */
    public static void main(String[] args) {
        int a = 15;
        int b = 2;
        int x = a << b;
        int y = a >> b;
        int z = a >>> b;
        System.out.println("result:"+x);//result:60
        System.out.println("result:"+y);//result:3
        System.out.println("result:"+z);//result:3
    }
}

```

【程序 59】Integer 的使用

```
public class TestInteger {
    public static void main(String[] args) {
        Integer a = 130;
        Integer b = new Integer(127);
        Integer c = 126;
        for (int i = 125; i < 135; i++) {
            if (c == i) {
                System.out.println(126);
            }
            if (b == i) {
                System.out.println(127);
            }
            if (a == i) {
                System.out.println(130);
            }
        }
    }
}
```

Result:126,127,130

【程序 60】冒泡

```
public class MaoPao {

    /**
     * 冒(冒泡) 择(选择) 路(插入) 兮(希尔) 快(快速) 归(归并) 堆(堆排序)
     * @author: jilongliang
     */

    public static void main(String[] args) {
        int i[] = { 3, 3232323, 4232 };
        Str(i);
        for (int k : i) {
            System.out.println(k);
        }
    }

    public static void Str(int s[]) {
        int temp;
        for (int i = 0; i < s.length; i++) {
            for (int j = i; j < s.length; j++) {
                if (s[i] > s[j]) {
                    temp = s[i];
                    s[i] = s[j];
                    s[j] = temp;
                }
            }
        }
    }
}
```

```

}
    }
    }
}

```

```
String s1="你好"; int i=3;s1+=i; //
String s2="你好"; int y=3; if (i==s2){} //Error
String s3="你好"; int k=3; s3=k+s3;
String s4="你好"; int z=3; s4=z+; //Error
String s5=null; int o=(s5!=null)&&(s5.length()>0)?s5.length():0;
```

```
public class ThreadTest1 {
    private int j;
    public static void main(String args[]) {
        ThreadTest1 tt = new ThreadTest1();
        Inc inc = tt.new Inc();
        Dec dec = tt.new Dec();
        for (int i = 0; i < 2; i++) {
            Thread t = new Thread(inc);
            t.start();
            t = new Thread(dec);
            t.start();
        }
    }
    private synchronized void inc() {
        j++;
        System.out.println(Thread.currentThread().getName()
        + "-inc:" + j);
    }
    private synchronized void dec() {
        j--;
        System.out.println(Thread.currentThread().getName()
        + "-dec:" + j);
    }
    class Inc implements Runnable {
        public void run() {
            for (int i = 0; i < 100; i++) {
                inc();
            }
        }
    }
}
```

```

    }
    class Dec implements Runnable {
        public void run() {
            for (int i = 0; i < 100; i++) {
                dec();
            }
        }
    }
}

```

【程序 63】我们在 web 应用开发过程中经常遇到输出某种编码的字符，如 iso8859-1 等，如何输出一个某种编码的字符串

```

public class Gbk {
    public static void main(String[] args) {
        String str = "zijie";
        try {
            str = new String(str.getBytes("ISO-8859-1"), "GBK");
            str = new String(str.getBytes("UTF-8"), "UTF-8");
        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }
        System.out.println(str);
    }
}

```