**1.编写程序输出，找出2~200000之间的所有质数。希望用尽可能快的方法实现。**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int**[]prime=**new** **int** [300000];

prime[0]=2;prime[1]=3;prime[2]=5;

**int** total=3; //找到素数的个数

**int** i,n=200000; //要查找的范围(>=6)

total=*creat\_prime*(prime,n,total);

**for**(i=0;i<total;i++)

{

System.*out*.print(prime[i]+" ");

**if**((i+1)%10==0)

System.*out*.println();

}

}

**public** **static** **int** creat\_prime(**int** prime[ ],**int** n,**int** total)

{

**int** i, j, gab=2, count;

**for**(i=7;i<=n;i+=gab)

{

count=1;

gab=6-gab;

**for**(j=0;prime[j]\*prime[j]<=i;j++)

{

**if**(i%prime[j]==0)

{

count=0;

**break**;

}

}

**if**(count==1)

{

prime[total]=i;

total++;

}

}

**return** total;

}

}

**2.编写程序输出3到10000内的可逆素数。可逆素数是指：一个素数将其各位数字的顺序倒过来构成的反序数也是素数。如157和751均为素数，它们是可逆素数。**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int** i;

**int** count=0;

**for** (i = 3; i < 10000; i++) // 1分

{

**if** (*prime*(i)==**true**) // 1分

**if** (*prime*( *invert*(i))==**true**)

{

System.*out*.print(i+" ");//2分

count++;

**if**(count%10==0)

System.*out*.println();

}

}

}

**public** **static** **int** invert(**int** a)

{

**int** s = 0;

**while** (a > 0)

{

s = 10\*s+ a % 10 ;

a /= 10;

}

**return** s;

}

**public** **static** **boolean** prime(**int** a)

{

**int** i;

**for** (i = 2; i <= (**int**)Math.*sqrt*(a); i++)

**if** ( a % i==0)

**return** **false**;

**return** **true**;

}

}

**3.歌德巴赫猜想问题是指一个不小于6的偶数n能够表示为两个素数之和。如6＝3＋3，8＝3＋5，10＝3＋7等。**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

System.*out*.println("请输入一个大于6的偶数：");

Scanner s1 = **new** Scanner(System.*in*);

**int** n = s1.nextInt();

**if**(n<6||n%2!=0)

System.*out*.println("输入错误！请重新输入！");

**else**

{

System.*out*.println(n+"的歌德巴赫猜想是：");

*gotbaha*(n);

}

}

**public** **static** **void** gotbaha(**int** n)

{

**int** i;

**for** (i = 3; i < n; i++)

{

**if** (*prime*(i)==**true**&&*prime*(n-i)==**true**&&i<(n-i))

System.*out*.println(n+"="+i+"+"+(n-i));

}

}

**public** **static** **boolean** prime(**int** a)

{

**int** i;

**for** (i = 2; i <= (**int**)Math.*sqrt*(a); i++)

**if** ( a % i==0)

**return** **false**;

**return** **true**;

}

}

请输入一个大于6的偶数：

16

16的歌德巴赫猜想是：

16=3+13

16=5+11

**4.-1找出范围2--100000间的所有回文质数; 从左到右和从右到左是看一样的素数。如1，2，5，151等**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int** i;

**int** count=0;

**for** (i = 2; i < 100000; i++)

{

**if** (*invert*(i)==**true**)

{

**if** (*prime*(i)==**true**)

{

System.*out*.print(i+" ");

count++;

**if**(count%10==0)

System.*out*.println();

}

}

}

}

**public** **static** **boolean** invert(**int** a)

{

**int** s = 0;

**int** t = a;

**while** (a > 0)

{

s = 10\*s+ a % 10 ;

a /= 10;

}

**if**(s==t)

**return** **true**;

**return** **false**;

}

**public** **static** **boolean** prime(**int** a)

{

**int** i;

**for** (i = 2; i <= (**int**)Math.*sqrt*(a); i++)

**if** ( a % i==0)

**return** **false**;

**return** **true**;

}

}

**4-2.判断一个整数是不是回文**

**import** java.util.Scanner;

**public** **class** PanduanHuiwenShuzi

{

**public** **static** **void** main(String[] args)

{

System.*out*.println("请输入一个整数：");

Scanner s1 = **new** Scanner(System.*in*);

**int** n = s1.nextInt();

**if**(*invert*(n)==**true**)

System.*out*.println(n+"是回文");

**else**

System.*out*.println(n+"不是回文");

}

**public** **static** **boolean** invert(**int** a)

{

**int** s = 0;

**int** t = a;

**while** (a > 0)

{

**s = 10\*s+ a % 10 ;**

**a /= 10;**

}

**if**(s==t)

**return** **true**;

**return** **false**;

}

}

**4-3.判断一个字符串是不是回文**

**import** java.util.Scanner;

**public** **class** HuiwenZifu

{

**public** **static** **void** main(String[] args)

{

System.*out*.print("请输入一串字符：");

Scanner s = **new** Scanner(System.*in*);

String n = s.nextLine();

StringBuffer s1=**new** StringBuffer(n);//将字符缓存中的内容倒置

s1.reverse();//计算出str与sb中对应位置字符相同的个数n

**int** t=0;

**for**(**int** i=0;i<n.length();i++)

{

**if**(n.charAt(i)==s1.charAt(i))

t++;

}

**if**(t==n.length())

System.*out*.println(n+"是回文!");

**else**

System.*out*.println(n+"不是回文!");

}

}

**5-1.找出三个整数的所有因子数。 一个整数n的因子数为包含它自身的所有因子的个数。例如：12的因子数为6(1,2,3,4,6,12)。**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int** i,n;

Scanner s = **new** Scanner(System.*in*);

System.*out*.println("输入三个整数:");

**for**(i=1;i<4;i++)

{

n = s.nextInt();

System.*out*.println(n+"的因子个数"+ *facter* (n));

}

}

**public** **static** **int** facter (**int** n)

{

**int** i,count=2;

**for** (i = 2; i <= n/2; i++)

**if** ( n % i==0)

count++;

**return** count;

}

}输入三个整数:

3 8 6

3的因子个数2

8的因子个数4

6的因子个数4

**5-2.一个数如果恰好等于除它本身外的因子之和，这个数就称“完数”。例如6=1＋2＋3.**

**import** java.util.Scanner;

**public** **class** WanYin {

**public** **static** **void** main(String[] args)

{

**int** i,n;

Scanner s = **new** Scanner(System.*in*);

System.*out*.println("输入一个整数:");

n = s.nextInt();

**if**(n==*facter*(n))

System.*out*.println(n+"是完全因子数");

**else**

System.*out*.println(n+"不是完全因子数");

}

**public** **static** **int** facter (**int** n)

{

**int** i,sum=1;

**for** (i = 2; i <= n/2; i++)

**if** ( n % i==0)

sum=sum+i;

**return** sum;

}

}

输入一个整数:

8

8不是完全因子数

输入一个整数:

6

6是完全因子数

**5-3.输出1000以内所以的“完数”**

**import** java.util.Scanner;

**public** **class** WanYin {

**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*.print(i + " ");

}

}

}

}

**1到1000的完数有：**

**6 28 496**

**6.如果一个自然数比所有比它小的自然数的约数个数都要多，那么我们就称这个数为一个反素数。例如，1、2、4、6、12和24都是反素数。**

**import** java.util.Scanner;

**public** **class** FanSushu

{

**public** **static** **void** main(String[] args)

{

System.*out*.println("Please input three numbers:");

Scanner sc = **new** Scanner(System.*in*);

**int** a,b,c;

**int** m = sc.nextInt();

**int** n = sc.nextInt();

**int** j = sc.nextInt();

a=*su*(m);

b=*su*(n);

c=*su*(j);

System.*out*.println(a+" "+b+" "+c);

}

**public** **static** **int** yue(**int** a)

{

**int** b,j=0;

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

{

**if**(a%b==0)

{j++;}

}

**return** j;

}

**public** **static** **int** su(**int** a)

{

**int** i,b,c;

b=1;c=1;

**for**(i=1;i<=a;i++)

{

**if**(b<=*yue*(i))

{

b=*yue*(i);

c=i;

}

}

**return** c;

}

}

**Please input three numbers:**

**10 100 1000**

**10 96 840**

**7.输入两个正整数m和n，求其最大公约数和最小公倍数。**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input=**new** Scanner(System.*in*);

System.*out*.print("请输入两个数：");

**int** m=input.nextInt();

**int** n=input.nextInt();

System.*out*.println("这两个数的最小公约数是："+*gac*(m,n));

System.*out*.println("这两个数的最小公倍数是："+m\*n/*gac*(m,n));

}

**public** **static** **int** gac(**int** a,**int** b)

{

**if**(a%b==0)

**return** b;

**else**

**return** *gac*(b,(a%b));

}

}**键入一个整数： 25**

**再键入一个整数： 30**

**最大公约数: 5**

**最小公倍数: 150**

**题目：判断闰年**

**import** java.util.Scanner;

**public** **class** Test{

**public** **static** **void** main(String[] args)

{

System.*out*.print("请输入年份：");

**int** year;

Scanner input = **new** Scanner(System.*in*);

year = input.nextInt();

**if** (year<0)

System.*out*.println("年份有误，程序退出！");

**else**

**if**((year%4==0)&&(year%100!=0)||(year%400==0&&year%3200!=0)||year%3200==0&&year%172800==0)

System.*out*.println(year+"是闰年");

**else**

System.*out*.println(year+"不是闰年");

}

}请输入年份：86400

86400不是闰年

**题目：杨辉三角形**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int**[][] a = **new** **int**[12][12];

**for**(**int** i=0; i<12; i++)

{a[i][i] = 1;a[i][0] = 1;}

**for**(**int** i=1; i<12; 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<12; i++)

{

**for**(**int** k=0; k<(12-i); k++)

System.*out*.print(" ");

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

{

**if**(a[i][j]<10)

System.*out*.print(a[i][j]+" " );

**else**

**if**(a[i][j]<100)

System.*out*.print(a[i][j]+" " );

**else**

System.*out*.print(a[i][j]+" " );

}

System.*out*.println();

}

}

}

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

1 8 28 56 70 56 28 8 1

1 9 36 84 126 126 84 36 9 1

1 10 45 120 210 252 210 120 45 10 1

1. 11 55 165 330 462 462 330 165 55 11 1

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

1 8 28 56 70 56 28 8 1

1 9 36 84 126 126 84 36 9 1

1 10 45 120 210 252 210 120 45 10 1

1. 11 55 165 330 462 462 330 165 55 11 1

**题目：打印出如下图案（菱形）**  
     \*     
   \*\*\*     
 \*\*\*\*\*     
\*\*\*\*\*\*\*     
 \*\*\*\*\*     
   \*\*\*     
    \*     
**public** **class** Test

{

**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();

}

}

}

**题目：矩阵对角线之和**

**import** java.util.Scanner;

**public** **class** Test

{

**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);

}

}

**题目：矩阵转置**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

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

**int** i,j;

**for**(i = 1;i < 3;i++)

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

{

**int** t;

t = a[i][j];

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

a[j][i] = t;

}

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

{

**for**(j = 0;j < 3;j++)

System.*out*.print(a[i][j] + " ");

System.*out*.println();

}

}

}

1 4 7

2 5 8

3 6 9

**题目：矩阵相加（一）**

**public** **class** Test {

**public** **static** **void** main(String[] args) {

**int** a[][] = {{1,2,3},{3,4,5}};

**int** b[][] = {{4,5,6},{6,3,2}};

**int**[][] c = **new** **int**[2][3];

**int** i,j;

**for**(i = 0;i<2;i++)

{

**for**(j = 0;j < 3;j++)

{

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

System.*out*.print(c[i][j] + " ");

}

System.*out*.println();

}

}

}

5 7 9

9 7 7

**题目：矩阵相加（二）**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner s = **new** Scanner(System.*in*);

System.*out*.println("请输入矩阵的行和列:");

**int** x=s.nextInt();

**int** y=s.nextInt();

**int**[][] a = **new** **int**[x][y];

**int**[][] b = **new** **int**[x][y];

**int**[][] c = **new** **int**[x][y];

**int** i,j;

System.*out*.println("请输入 "+x+"行"+y+"列矩阵中的数值(第一个矩阵)：");

**for**(i=0; i<x; i++)

**for**(j=0; j<y; j++)

a[i][j] = s.nextInt();

System.*out*.println("请输入 "+x+"行"+y+"列矩阵中的数值(第二个矩阵)：");

**for**(i=0; i<x; i++)

**for**(j=0; j<y; j++)

b[i][j] = s.nextInt();

System.*out*.println("两个矩阵相加得如下矩阵：");

**for**(i = 0;i<x;i++)

{

**for**(j = 0;j < y;j++)

{

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

System.*out*.print(c[i][j] + " ");

}

System.*out*.println();

}

}

}

请输入矩阵的行和列:

2 3

请输入 2行3列矩阵中的数值(第一个矩阵)：

1 3 5

6 8 2

请输入 2行3列矩阵中的数值(第二个矩阵)：

3 7 9

2 5 1

两个矩阵相加得如下矩阵：

4 10 14

8 13 3

**题目：矩阵相乘**

**public** **class** Test {

**public** **static** **void** main(String[] args) {

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

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

**int**[][] c = **new** **int**[3][3];

**int** i,j,k;

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

{

**for**(j = 0;j < 3;j++)

{

c[i][j]=0;

**for**(k = 0;k < 3;k++)

c[i][j] = c[i][j] + a[i][k] \* b[k][j];

System.*out*.print(c[i][j] + " ");

}

System.*out*.println();

}

}

}

30 36 42

66 81 96

102 126 150

**题目：冒泡排序**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int** p[]={2,8,30,10,23,31};

**int** k, j = 0 ,d;

**for**( k = 1; k < 6; k++ ) //扫描数组趟数控制

{ **for**( j = 0; j < 6-k; j++)

**if** ( p[ j+1 ] < p[ j ] ) //两两比较

{ d = p[ j ]; p[ j ] = p [ j+1 ]; //交换

p[ j+1 ] = d; }

}

System.*out*.print("冒泡排序后的结果：");

**for**( j = 0; j < 6; j++)

System.*out*.print(p[j]+" ");

}

}

**题目：插入排序**

public class ghj

{

public static void main(String[] args)

{

int p[]={2,8,30,10,23,31};

int j, k,t ;

for ( j = 1; j < 6; j++ )

{ t = p[ j ]; k = j-1;

while ( ( k >= 0 ) && (p[ k ] > t) )

{ p[ k+1 ] = p[ k ]; //准备比较下一个

k = k-1; }

p[ k+1 ] = t ; // p[k] <= t

}

System.out.print("插入排序后的结果：");

for( j = 0; j < 6; j++)

System.out.print(p[j]+" ");

}

}

插入排序后的结果：2 8 10 23 30 31

**题目：选择排序**

设数组内存放了n个待排数字，数组下标从1开始，到n结束。首先i=1。从数组的第i个元素

开始到第n个元素，寻找最小的元素。将上一步找到的最小元素和第i位元素交换。

如果i=n－1算法结束，否则回到第3步。选择排序的平均时间复杂度也是O(n&sup2;)的。

public class Test

{

public static void main(String[] args)

{

int p[]={2,8,30,10,23,31};

int i, j, k,d;

for (i=0; i<5; i++)

{ k=i;

for ( j=i+1; j<6; j++)

if (p[j]<p[k]) k=j;

if ( k != i ) { d=p[i]; p[i]=p[k]; p[k]=d; }

}

System.out.print("选择排序后的结果：");

for( j = 0; j < 6; j++)

System.out.print(p[j]+" ");

}

}

选择排序后的结果：2 8 10 23 30 31

**题目：10进制转化成其他进制**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input=**new** Scanner(System.*in*);

System.*out*.print("请输入一个整数(10进制)：");

**int** n=input.nextInt();

System.*out*.print(n+"的二进制是:");

*Decton*(n,2);

System.*out*.print(" "+n+"的八进制是:");

*Decton*(n,8);

System.*out*.print(" "+n+"的十六进制是:");

*Decton*(n,16);

}

**public** **static** **void** Decton(**int** m,**int** n)

{

**int** t=0,k=1,sum=0;

**while**(m>0)

{

t=m%n;

sum=t\*k+sum;

k=k\*10;

m=m/n;

}

System.*out*.print(sum+" ");

}

}

请输入一个整数(10进制)：24

24的二进制是:11000 24的八进制是:30 24的十六进制是:18

**题目：16进制转化成其他进制**

**import** java.math.BigInteger;

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input = **new** Scanner(System.*in*);

BigInteger bi = **null**;

String enter = **null**;

System.*out*.println("请输入你要转换的十六进制数字：");

enter = input.next();

bi = **new** BigInteger(enter, 16);

String show1 = bi.toString(10);

System.*out*.println("该数转换成10进制是："+show1);

String show2 = bi.toString(8);

System.*out*.println("该数转换8进制是："+show2);

String show3 = bi.toString(2);

System.*out*.println("该数转换2进制是："+show3);

}

}

请输入你要转换的十六进制数字：

23

该数转换成10进制是：35

**题目：在歌星大奖赛中，有10个评委为参赛的选手打分，分数为1~100分。选手最后得分为：去掉一个最高分和一个最低分后其余8个分数的平均值。**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int** i,j,sum = 0,min = 0,max = 0;

**int** a[]=**new** **int**[10];

System.*out*.println("请输入10个评委的分数：");

Scanner sc = **new** Scanner(System.*in*);

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

{

a[i] = sc.nextInt();

sum += a[i];

min = max = a[0];

**for**(j = 1;j < 10;j++)

{

**if**(min >= a[j])

min = a[j];

**if**(max <= a[j])

max = a[j];

}

}

sum = sum - min - max;

System.*out*.print("该歌星的最终的分为："+sum/8);

}

}

请输入10个评委的分数：

10 20 30 40 50 60 70 80 90 100

该歌星的最终的分为：55

**题目：求555555的约数中最大的三位数**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input=**new** Scanner(System.*in*);

System.*out*.println("请输入一个6位数:");

**int** n=input.nextInt();

**int** j;

**for**(j=999;j>=100;j--)

**if**(n%j==0)

{

System.*out*.print(n+"的约数中最大的三位数是："+j);

**break**;

}

}

}

请输入一个6位数:

555555

555555的约数中最大的三位数是：777

**题目：高次方数的尾数**，如求13的13次方的最后三位数（研究乘法的规律发现：乘积的最后三位的值只与乘数和被乘数的后三位有关，与乘数和被乘数的高位无关。利用这一规律，可以大大简化程序。）

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input=**new** Scanner(System.*in*);

System.*out*.print("请输入x和Y:");

**int** x=input.nextInt();

**int** y=input.nextInt();

**int** i,last=1; //变量last保存求X的Y次方过程中的部分乘积的后三位

**for**(i=1;i<=y;i++) //X自乘Y次

last=last\*x%1000; //求积的后三位

System.*out*.print(x+"的"+y+"次方的后三位:"+last%1000);

}

}

请输入x和Y:13 13

13的13次方的后三位:253

**A、B、C、D、E五个人在某天夜里合伙去捕鱼，到第二天凌晨时都疲惫不堪，于是各自找地方睡觉。日上三杆，A第一个醒来，他将鱼分为五份，把多余的一条鱼扔掉，拿走自己的一份。B第二个醒来，也将鱼分为五份，把多余的一条鱼扔掉，保持走自己的一份。C、D、E依次醒来，也按同样的方法拿走鱼。问他们合伙至少捕了多少条鱼？**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int** n,x,i,flag=1; //flag=1可分配

**for**(n=6;flag==1;n++) /\*用试探的方法。试探值n逐步加大\*/

{

**for**(x=n,i=1,flag=1;i<=5;i++)

{

**if**((x-1)%5==0)

x=4\*(x-1)/5;

**else** flag=0; //不能分配则置标记falg=0退出分配过程

}

**if**(flag==1) **break**; //若分配过程正常结束则找到结果退出试探的过程

**else** flag=1; //否则继续试探下一个数

}

System.*out*.print("鱼的总数是："+n);

}

}

**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** Test{

**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"); }

}

}

}

153是一个水仙花数

370是一个水仙花数

371是一个水仙花数

407是一个水仙花数

**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。

**import** java.util.Scanner;

**public** **class** Test

{

**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*);//扫描器输入信息

System.*out*.println("输入一组字符");

String str=scan.nextLine();//取出输入的信息

**char**[] ch = str.toCharArray();//把取道的字符串变成一个char数组

**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);

}

}

输入一组字符

usfyhoiu9480,n/;][nkn[

字母个数：12

数字个数：4

空格个数：0

其他字符个数：6

**题目：计算“a+aa+aaa+...”**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args) {

**long** b = 0, sum = 0;

Scanner s = **new** Scanner(System.*in*);

System.*out*.print("输入数字a的值和相加的项数： ");

**long** a = s.nextLong();

**int** n = s.nextInt();

**int** i = 0;

**while**(i < n) {

b = b + a;

sum = sum + b;

a = a \* 10;

++ i;

}

System.*out*.println(n+"项该数做a+aa+aaa+...运算的结果是："+sum);

}

}

**题目：16进制转其他进制**

**import** java.math.BigInteger;

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input = **new** Scanner(System.*in*);

BigInteger bi = **null**;

String enter = **null**;

System.*out*.println("请输入你要转换的十六进制数字：");

enter = input.next();

bi = **new** BigInteger(enter, 16);

String show1 = bi.toString(10);

System.*out*.println("该数转换成10进制是："+show1);

String show2 = bi.toString(8);

System.*out*.println("该数转换8进制是："+show2);

String show3 = bi.toString(2);

System.*out*.println("该数转换2进制是："+show3);

}

}

请输入你要转换的十六进制数字：

abcd

该数转换成10进制是：43981

该数转换8进制是：125715

该数转换2进制是：1010101111001101

**题目：10进制转其他进制**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input=**new** Scanner(System.*in*);

System.*out*.print("请输入一个整数(10进制)：");

**int** n=input.nextInt();

System.*out*.print(n+"的二进制是:");

*Decton*(n,2);

System.*out*.print(" "+n+"的八进制是:");

*Decton*(n,8);

System.*out*.print(" "+n+"的十六进制是:");

*Decton*(n,16);

}

**public** **static** **void** Decton(**int** m,**int** n)

{

**int** t=0,k=1,sum=0;

**while**(m>0)

{

t=m%n;

sum=t\*k+sum;

k=k\*10;

m=m/n;

}

System.*out*.print(sum+" ");

}

}

请输入一个整数(10进制)：125

125的二进制是:1111101 125的八进制是:175 125的十六进制是:83

**题目：求1！+2！+3！+4！+5！**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int** i,sum=0;

**for**(i=1;i<=5;i++)

{sum+=*fun*(i);}

System.*out*.println(sum);

}

**public** **static** **int** fun(**int** n)

{

**if**(n==1)

**return** n;

**else**

**return** n\**fun*(n-1);

}

}

**题目：递归，等差**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input=**new** Scanner(System.*in*);

System.*out*.print("请输入差数列第一项的值和要求的项数：");

**int** m=input.nextInt();//设首项为 m

**int** n=input.nextInt();//要求第n项

System.*out*.print("该等差数列第"+n+"项的值是"+*getan*(n,m));

}

**public** **static** **int** getan(**int** n, **int** a1) // 要求第n项,a1为已知的首项

{

**if**(n==1)

**return** a1;

**else**

**return** *getan*(n-1,a1)+2 ;

}

}

请输入差数列第一项的值和要求的项数：3 5

该等差数列第5项的值是11

**题目：递归m的n次幂之和**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

**int** sum=0;

**for**(**int** i=1;i<=5;i++)

{

sum=sum+*fun*(i,i);

}

System.*out*.println(sum);

}

**public** **static** **int** fun(**int** m,**int** n)

{

**if** (n==0)

**return** 1;

**if**(n==1)

**return** m;

**else**

**return** m\**fun*(m,n-1);

}

}

**题目：Cnm=n!/(n-m)!\*m!**

**import** java.util.Scanner;

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

Scanner input=**new** Scanner(System.*in*);

System.*out*.print("请输入两个数：");

**int** m=input.nextInt();

**int** n=input.nextInt();

System.*out*.print(*fun*(n)/(*fun*(n-m)\**fun*(m)));

}

**public** **static** **int** fun(**int** n)

{

**if**(n==1)

**return** n;

**else**

**return** n\**fun*(n-1);

}

}

**题目：振兴中华**

**public** **class** Test

{

**public** **static** **void** main(String[] args)

{

System.*out*.print(*fun*(1,1));

}

**public** **static** **int** fun(**int** x, **int** y)

{

**if**(x==4&&y==5)

*s*=*s*+1;

**if**(x>4||y>5)

**return** -1;

**else**

{

*fun*(x,y+1);*fun*(x+1,y);

}

**return** *s*;

}

**static** **int** *s*=0;

}