

一、请编写一个程序将一个整数转换为汉字读法字符串。比如 “1123” 转换为 “一千一百二十三” 。

源程序：

```
// 王荣荣2016/11/6
import java.util.Scanner;
public class Daxie
{
    private String[] hanArr = {"零", "一", "二", "三", "四",
        "五", "六", "七", "八", "九"};
    private String[] unitArr = {"十", "百", "千", "万", "十万", "百万"};
    /**
     * 把一个四位的数字字符串变成汉字字符串
     * @param numStr 需要被转换的四位的数字字符串
     * @return 四位的数字字符串被转换成的汉字字符串。
     */
    private String toHanStr(String numStr)
    {
        String result = "";
        int numLen = numStr.length();
        //依次遍历数字字符串的每一位数字
        for (int i = 0 ; i < numLen ; i++)
        {
            //把char型数字转换成的int型数字，因为它们的ASCII码值恰好相差48
            //因此把char型数字减去48得到int型数字，例如'4'被转换成4。
            int num = numStr.charAt(i) - 48;
            //如果不是最后一位数字，而且数字不是零，则需要添加单位（千、百、十）
            if ( i != numLen - 1 && num != 0)
            {
                result += hanArr[num] + unitArr[numLen - 2 - i];
            }
            //否则不要添加单位
            else
            {
                //上一个数是否为“零”，不为“零”时就添加
                if(result.length()>0 && hanArr[num].equals("零") && result.charAt(result.length()-1)=='零')
                    continue;
                result += hanArr[num];
            }
        }
        //只有个位数，直接返回
        if(result.length()==1)
            return result;
        int index=result.length()-1;
        while(result.charAt(index)=='零'){
            index--;
        }
    }
}
```

```

    }
    if(index!=result.length()-1)
        return result.substring(0, index+1);
    else {
        return result;
    }
}
public static void main(String[] args)
{
    Daxie nr = new Daxie();
    Scanner scan=new Scanner(System.in);
    System.out.println("请输入整数（只支持整数（0~百万））:");
    String number= scan.next();
    System.out.println("\n"+number+"的汉字读法为:"+nr.toHanStr(number));
}
}

```

结果:



3、更进一步，能否将数字表示的金额改为“汉字表达？比如将“¥ 123.52”转换为“壹佰贰拾叁元伍角贰分”。

源程序:

```

// 王荣荣2016/11/6
import java.util.Scanner;
public class Daxie2
{
    private String[] hanArr = {"零", "壹", "贰", "叁", "肆",
        "伍", "陆", "柒", "捌", "玖"};
    private String[] unitArr = {"十", "百", "千", "万", "十万", "百万"};
    /**
     * 把一个四位的数字字符串变成汉字字符串
     * @param numStr 需要被转换的四位的数字字符串
     * @return 四位的数字字符串被转换成的汉字字符串。
     */
    private String toHanStr(String numStr)

```

```

{
String result = "";
int numLen = numStr.length();
//依次遍历数字字符串的每一位数字
for (int i = 0 ; i < numLen ; i++ )
{
    //把char型数字转换成的int型数字，因为它们的ASCII码值恰好相差48
    //因此把char型数字减去48得到int型数字，例如'4'被转换成4。
    int num = numStr.charAt(i) - 48;
    //如果不是最后一位数字，而且数字不是零，则需要添加单位（千、百、十）
    if ( i != numLen - 1 && num != 0)
    {
        result += hanArr[num] + unitArr[numLen - 2 - i];
    }
    //否则不要添加单位
    else
    {
        //上一个数是否为“零”，不为“零”时就添加
        if(result.length()>0 && hanArr[num].equals("零") && result.charAt(result.length()-1)=='零')
            continue;
        result += hanArr[num];
    }
}
//只有个位数，直接返回
if(result.length()==1)
    return result;
int index=result.length()-1;
while(result.charAt(index)=='零'){
    index--;
}
if(index!=result.length()-1)
    return result.substring(0,index+1);
else {
    return result;
}
}
public static void main(String[] args)
{
    Daxie2 nr = new Daxie2();
    Scanner scan=new Scanner(System.in);
    System.out.println("请输入整数（只支持整数（0~百万））:");
    String number= scan.next();
    System.out.println("\n"+number+"的汉字读法为:"+nr.toHanStr(number)+"元");
}
}

```

结果:

```
Console X
<terminated> Daxie2 [Java Application] C:\Program
请输入整数（只支持整数（0~百万））：
563298

563298的汉字读法为:伍十万陆万叁千贰百玖十捌元
```

二、前面几讲介绍过JDK所提供的BigInteger能完成大数计算，如果不用它，直接使用数组表达大数，你能实现相同的功能吗？

要求：

- (1) 用你的大数类实现加和减两个功能
- (2) 阅读BigInteger类源码，弄清楚它是使用什么算法实现加减乘除四种运算的？
- (3) 通过互联网查找大数运算的相关资料，给你的大数类添加乘、除、求阶乘等其它功能。

1、数组大数类实现加法运算源程序：

```
//王荣荣2016/11/6
import java.util.Scanner;
public class Sum{
    public static int[] add(int []a, int []b){
        int digit=0;//位数
        int[]c=new int[a.length];
        for(int i=a.length-1;i>=0;i--){
            c[i]=a[i]+b[i]+digit;
            if(c[i]<10)
                digit=0;
            else
            {
                c[i]=c[i]-10;
                digit=1;
            }
        }
        return c;
    }
    public static int [] sub(int []a, int []b, int w)
    {
        int digit=0;
        int[]c=new int[a.length];
```

```

for(int i=a.length-1;i>=0;i--)
{
    if(w<=0)
    {
        c[i]=b[i]-a[i]-digit;
        if(c[i]>=0)
        {    digit=0;}
        else
        {
            c[i]=c[i]+10;
            digit=1;
        }
    }
    else
    {
        c[i]=a[i]-b[i]-digit;
        if(c[i]>=0)
        {digit=0;}
        else
        {
            c[i]=c[i]+10;
            digit=1;
        }
    }
}
return c;
}

public static void main(String[]args){
    int a[]=new int[50];
    int b[]=new int[50];
    int m=0;
    int n=0;
    int s=0;
    int t=0;int w=0;
    Scanner in=new Scanner(System.in);
    System.out.println("请输入第一个大数:");
    String aa=in.next();
    System.out.println("请输入第二个大数:");
    String bb=in.next();
    m=a.length-aa.length();
    n=b.length-bb.length();
    if(aa.length()>bb.length())
    {
        w=1;
    }
    else if (aa.length()<bb.length())
    {

```

```

        w=-1;
    }
    else
    {w = aa.compareTo(bb);}
    for (int i = 0; i < aa.length(); i++)
    {
        a[m++] = aa.charAt(i) - 48;
    }
    for (int j = 0; j < bb.length(); j++)
    {
        b[n++] = bb.charAt(j) - 48;
    }

    int[] c = Test1.add(a, b);
    for (int k = 0; k < c.length; k++)
    {
        if (c[k] > 0)
        {
            s = k;
            break;
        }
    }
    System.out.print("大数相加的结果为: ");
    for (int i = s; i < c.length; i++) {
        System.out.print(c[i]);
    }
    System.out.println();
    int[] d = Test1.sub(a, b, w);
    for (int k = 0; k < d.length; k++)
    {
        if (d[k] > 0)
        {
            t = k;
            break;
        }
    }
    System.out.print("大数相减的结果为: ");
    if (w < 0)
        System.out.print("-");
    for (int i = t; i < d.length; i++)
    {
        System.out.print(d[i]);
    }
    System.out.println();
    System.out.println();
    System.out.println();
    System.out.println();

```

```
    }  
}
```

```
请输入第一个大数:  
2354  
请输入第二个大数:  
5632  
大数相加的结果为: 7986  
大数相减的结果为: -3278
```

2、上网百度的用数组大数类实现加减乘运算源程序:

```
/*自定义大数类，用数组实现任意超大整数的加减乘运算*/  
import java.util.Random;  
import java.util.Scanner;  
public class Bignumber {  
    private int[] num;  
    // 无参构造函数  
    public Bignumber() {  
    }  
    // 有参构造函数  
    public Bignumber(int i) {  
        num = new int[i];  
        Random random = new Random();  
        int j;  
        for (j = 0; j < i; j++)  
            num[j] = random.nextInt(10);  
        // 当生成的数首位是0的话，让重新生成  
        while (num[i - 1] == 0) {  
            num[i - 1] = random.nextInt(10);  
        }  
    }  
    // 加法  
    public static Bignumber add(Bignumber bigA, Bignumber bigB) {  
        int alen = bigA.num.length;  
        int blen = bigB.num.length;  
        int clen = Math.max(alen, blen);  
        Bignumber result = new Bignumber();  
        result.num = new int[clen];  
        if (alen >= blen) {  
            for (int i = 0; i < blen; i++)
```

```

        result.num[i] = bigA.num[i] + bigB.num[i];
    for (int j = blen; j < alen; j++)
        result.num[j] = bigA.num[j];
} else {
    for (int i = 0; i < alen; i++)
        result.num[i] = bigA.num[i] + bigB.num[i];
    for (int j = alen; j < blen; j++)
        result.num[j] = bigB.num[j];
}
for (int k = 0; k < clen - 1; k++) {
    if (result.num[k] >= 10) {
        result.num[k] -= 10;
        result.num[k + 1]++;
    }
}
return result;
}
// 减法
public static Bignumber subtract(Bignumber bigA, Bignumber bigB) {
    int alen = bigA.num.length;
    int blen = bigB.num.length;
    int clen = Math.max(alen, blen);
    Bignumber result = new Bignumber();
    result.num = new int[clen];
    if (alen > blen) {
        for (int i = 0; i < blen; i++)
            result.num[i] = bigA.num[i] - bigB.num[i];
        for (int j = blen; j < alen; j++)
            result.num[j] = bigA.num[j];
    } else if (alen < blen) {
        for (int i = 0; i < alen; i++)
            result.num[i] = bigB.num[i] - bigA.num[i];
        for (int j = alen; j < blen; j++)
            result.num[j] = bigB.num[j];
    } else {
        if (isBigger(bigA, bigB)) {
            for (int i = 0; i < clen; i++) {
                result.num[i] = bigA.num[i] - bigB.num[i];
            }
        } else {
            for (int i = 0; i < clen; i++) {
                result.num[i] = bigB.num[i] - bigA.num[i];
            }
        }
    }
    for (int k = 0; k < clen - 1; k++) {
        if (result.num[k] < 0) {

```



```

        result.num[k] += 10;
        result.num[k + 1]--;
    }
}
return result;
}
// 乘法
public static Bignumber multiply(Bignumber bigA, Bignumber bigB) {
    int alen = bigA.num.length;
    int blen = bigB.num.length;
    int clen = alen + blen;
    int t;
    Bignumber result = new Bignumber();
    result.num = new int[clen];
    for (int i = 0; i < alen; i++) {
        Bignumber temp = new Bignumber();
        temp.num = new int[clen];
        for (int j = 0; j < blen; j++) {
            temp.num[i + j] = bigA.num[i] * bigB.num[j];
        }
        for (int k = 0; k < clen; k++) {
            if (temp.num[k] >= 10) {
                t = temp.num[k];
                temp.num[k] = t % 10;
                temp.num[k + 1] += t / 10;
            }
        }
        result = add(result, temp);
    }
    return result;
}
// 判断两个位数相同的BigNumber那个大,前面的大于或者等于返回true,后面的大返回false
public static boolean isBigger(Bignumber bigA, Bignumber bigB) {
    boolean flag = true;
    for (int i = bigA.num.length - 1; i >= 0; i--) {
        if (bigA.num[i] < bigB.num[i]) {
            flag = false;
            break;
        } else if (bigA.num[i] > bigB.num[i]) {
            break;
        } else
            continue;
    }

    return flag;
}
// 打印

```

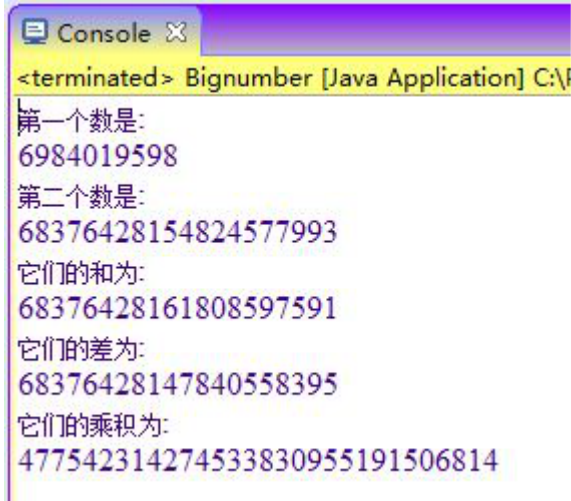
```

public void print() {
    if (num[num.length - 1] != 0)
        System.out.print(num[num.length - 1]);
    for (int i = num.length - 2; i >= 0; i--)
        System.out.print(num[i]);
    System.out.println();
}

public static void main(String[] args) {
    // TODO Auto-generated method stub
    Bignumber a = new Bignumber(10);
    System.out.println("第一个数是:");
    a.print();
    Bignumber b = new Bignumber(20);
    System.out.println("第二个数是:");
    b.print();
    System.out.println("它们的和为:");
    add(a, b).print();
    System.out.println("它们的差为:");
    subtract(a, b).print();
    System.out.println("它们的乘积为:");
    multiply(a, b).print();
}
}

```

结果:



```

<terminated> Bignumber [Java Application] C:\
第一个数是:
6984019598
第二个数是:
68376428154824577993
它们的和为:
68376428161808597591
它们的差为:
68376428147840558395
它们的乘积为:
477542314274533830955191506814

```

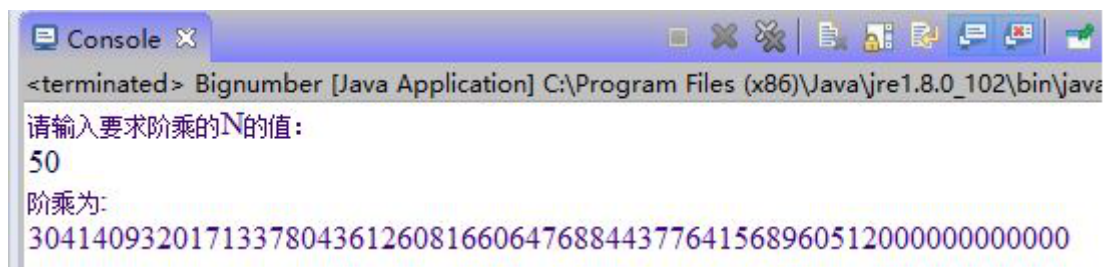
3、上网百度了求阶乘源程序:

```

import java.util.Scanner;
public class Bignumber {
    private static int[] resultArray = new int[10000000];
    static int resultJinwei = 0;
    static long index = 0;
    public static void main(String[] args) {
        System.out.println("请输入要求阶乘的N的值：");
        Scanner sin = new Scanner(System.in);
        int number = sin.nextInt();
        long maxIndex = method(number);
        System.out.println("阶乘为：");
        for (long i = maxIndex-1; i >= 0; i--) {
            System.out.print(resultArray[(int) i]);
            if(i % 100 == 0) { //此处对输出格式做处理时因为eclipse编译器的控制台每行输出的长度有限定，所以处理成每行输出100个数
                System.out.println();
            }
        }
    }
    public static long method(long number) {
        long maxIndex = 1;
        int temp = 0;
        //int tempMaxIndex = 0;
        resultArray[0] = 1;
        for (long i = 1; i <= number; i++) {
            for (long j = 0; j < maxIndex; j++) {
                resultArray[(int) j] *= i;
                resultArray[(int) j] += resultJinwei;
                temp = resultArray[(int) j];
                if (temp >= 10) {
                    resultArray[(int) index] = temp % 10;
                    resultJinwei = temp / 10;
                    index++;
                    if(maxIndex<index+1)
                        maxIndex = index+1;
                } else {
                    index++;
                    resultJinwei = 0;
                }
            }
            index = 0;
        }
        return maxIndex;
    }
}

```

结果：

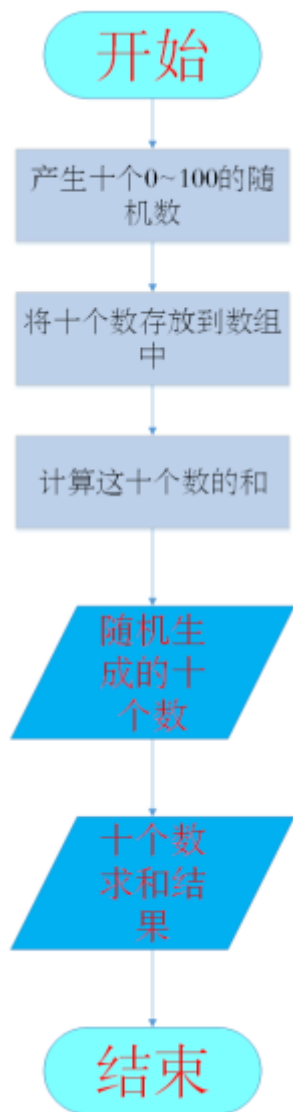


```
<terminated> Bignumber [Java Application] C:\Program Files (x86)\Java\jre1.8.0_102\bin\java
请输入要求阶乘的N的值:
50
阶乘为:
30414093201713378043612608166064768844377641568960512000000000000
```

三、随机生成10个数，填充一个数组，然后用消息框显示数组内容，接着计算数组元素的和，将结果也显示在消息框中。要求将设计思路、程序流程图、源程序代码、结果截图、编程总结等发表到博客园，并备份到课堂派。

1、程序设计思想：先定义一个一维数组，用for循环存放十个随机生成的0~100之间的数字，再逐个输出并求和。

2、程序流程图：



3、源代码：

```
// 王荣荣2016/11/6
import javax.swing.*.*;
public class Sum {
public static void main(String[] args) {
    String output= "随机生成的10个数为： \n";
    int sum=0;
```

```
int a[]=new int [10];
for(int i = 0;i<10;i++)
{
    a[i]=(int) (Math.random()*100);
    output += " "+a[i]+"\\n";
    sum=sum+a[i];
}
output += "这十个数的和为: "+sum;
JOptionPane.showMessageDialog(null,output,"结果",JOptionPane.PLAIN_MESSAGE);
}
}
```

结果:

