**1.判断素数**

**import** java.util.Scanner;

**public** **class** Prime\_palindrome {

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

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

System.*out*.println( "请输入一个整数： " );

**int** i;

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

**double** m = Math.*sqrt*( n );

**for**( i = 2;i <= m;i++ ){

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

System.*out*.println( "不是素数" );

**break**;

}

}

**if**( i > m ){

System.*out*.println( "是素数" );

}

}

}

或在for循环中用else

2.输出前100个素数（注意count++的位置）

**public** **class** Prime\_palindrome {

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

**int** i;

**for**( **int** n = 2,count = 0;count < 100;n++ ){

**double** m = Math.*sqrt*(n);

**for**( i = 2;i <= m;i++ ){

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

**break**;//能被i整除，不是素数，break

}

**if**( i > m ){

System.*out*.println(n);

count ++;

}

}

}

}

3.判断回文数。。天哪快哭了

**import** java.util.Scanner;

**public** **class** Palindrome {

**private** **static** **boolean** isPalindrome(String test){

String num = String.*valueOf*(test);

**return** **new** StringBuffer(num).reverse().toString().equalsIgnoreCase(num);

}

**private** **static** String isOrNot(**boolean** isOrNot){

**return** isOrNot ? " " : " NOT ";

}

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

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

System.*out*.println("请输入一个数： ");

String test = input.nextLine();

System.*out*.print(test+" is" + *isOrNot*(*isPalindrome*(test)));

System.*out*.println("a Plalindrome Number.");

}

}

4.

【用Java实现判断一个数是否是回文】

Code:

public class Test {

public static void main(String[] args) {

System.out.println(new Test().isPlalindrome(131));

} //此方法实现判断数字是不是回文数

public boolean isPlalindrome(int number){

String num = String.valueOf(number);

return new StringBuffer(num).reverse().toString().equalsIgnoreCase(num);

}

}

解析：

valueOf()返回参数的字符串表示形式

StringBuffer(String str)

构造一个字符串缓冲区，并将其内容初始化为指定的字符串内容。

reverse()将此字符序列用其反转形式取代。

toString()返回此序列中数据的字符串表示形式。

equalsIgnoreCase(num)将此 String 与另一个 String 比较，不考虑大小写。

5.输出前100个回文数

**public** **class** Palindrome {

**private** **static** **boolean** Palindrome(**int** n){

String num = String.*valueOf*(n);

**return** **new** StringBuffer(num).reverse().toString().equalsIgnoreCase(num);

}

**private** **static** String isOrNot(**boolean** isOrNot){

**return** isOrNot ? " " : " Not " ;

}

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

**for**( **int** n = 2,count = 0;count < 100;n++ ){

**if**( *isOrNot*(*Palindrome*(n)) == " " ){

System.*out*.println( n );

count ++;

}

}

}

}

5.数组合并 小vv

import java.util.Scanner;

public class Array {

public static void main(String []args){

Scanner x = new Scanner(System.in);

int list1[] = null, list2[] = null;

int n = x.nextInt();

list1 = new int[n];

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

list1[i] = x.nextInt();

}

int m = x.nextInt();

list2 = new int [m];

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

list2[j] = x.nextInt();

}

int list[] = new int[n+m];

list = merge(list1,list2);

for(int k = 0;k < list.length;k ++){

System.out.print(list[k]+" ");

}

}

public static int[]merge(int []list1,int []list2){

int i = 0,j = 0,k = 0;

int list[] = new int[list1.length+list2.length];

while(i < list1.length && j < list2.length)

{

if(list1[i] <= list2[j]){

list[k] = list1[i];

i++;

}

else{

list[k] = list2[j];

j++;

}

k++;

}

for(;i < list1.length;i++){

list[k]=list1[i];

k++;

}

for(;j < list2.length;j++){

list[k]=list2[j];

k++;

}

return list;

}

}

6.

**import** java.util.Scanner;

**public** **class** Merge\_Sorted\_Array {

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

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

System.*out*.println("请分别输入数组a的长度及其元素值&数组b的长度及其元素值： ");

**int** a[] = **new** **int**[100];

**int** b[] = **new** **int**[100];

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

**int** i,j;

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

a[i] = input.nextInt();

//System.out.print( a[i] );

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

}

//System.out.println();

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

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

b[j] = input.nextInt();

//System.out.print( b[j] );

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

}

**int** c[] = **new** **int**[200];

i=0;j=0;

**while**(i<x&&j<y){

//System.out.println(j);

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

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

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

i++;

//System.out.println(i);

}

**else** {

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

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

j++;

}

}

**for**( ; i < x ; i++ ){

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

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

}

**for**( ; j < y ; j++ ){

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

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

}

}

}

**import java.util.Scanner;**

**public class Array {**

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

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

**int list1[] = null, list2[] = null;**

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

**list1 = new int[n];**

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

**list1[i] = x.nextInt();**

**}**

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

**list2 = new int [m];**

**for(int j = 0;j < m;j++){**

**list2[j] = x.nextInt();**

**}**

**int list[] = new int[n+m];**

**list = merge(list1,list2);**

**for(int k = 0;k < list.length;k ++){**

**System.out.print(list[k]+" ");**

**}**

**}**

**public static int[]merge(int []list1,int []list2){**

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

**int list[] = new int[list1.length+list2.length];**

**while(i < list1.length && j < list2.length)**

**{**

**if(list1[i] <= list2[j]){**

**list[k] = list1[i];**

**i++;**

**}**

**else{**

**list[k] = list2[j];**

**j++;**

**}**

**k++;**

**}**

**for(;i < list1.length;i++){**

**list[k]=list1[i];**

**k++;**

**}**

**for(;j < list2.length;j++){**

**list[k]=list2[j];**

**k++;**

**}**

**return list;**

**}**

**}**

7.

**import** java.util.Scanner;

**public** **class** Merge\_Sorted\_Array {

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

System.*out*.println("请分别输入数组a的长度及其元素值&数组b的长度及其元素值： ");

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

**int** c[] = **new** **int**[200];

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

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

**for**( **int** k = 0;k < p+q; k++ )

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

}

**public** **static** **int**[]merge(**int** []list1,**int** []list2){

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

**int** a[] = **new** **int**[100];

**int** b[] = **new** **int**[100];

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

**int** i,j;

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

a[i] = input.nextInt();

//System.out.print( a[i] );

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

}

//System.out.println();

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

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

b[j] = input.nextInt();

//System.out.print( b[j] );

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

}

**int** c[] = **new** **int**[200];

i=0;j=0;

**while**(i<x&&j<y){

//System.out.println(j);

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

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

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

i++;

//System.out.println(i);

}

**else** {

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

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

j++;

}

}

**for**( ; i < x ; i++ ){

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

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

}

**for**( ; j < y ; j++ ){

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

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

}

**return** c;

}

}