比较器

import java.util.Arrays;

import java.util.Scanner;

class Student implements Comparable<Student>{

String name;

int number;

int score;

@Override

public int compareTo(Student o) {

if (this.score!=o.score){

return o.score-this.score;

}else {

return this.number-o.number;

}

}

}

public class Main {

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()) {

int n=cin.nextInt();

Student[] students=new Student[n];

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

students[i] = new Student();

students[i].name=cin.next();

students[i].number=cin.nextInt();

students[i].score=cin.nextInt();

}

Arrays.sort(students,Student::compareTo);

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

System.out.print("name:"+students[i].name);

System.out.print(";number:"+students[i].number);

System.out.println(";score:"+students[i].score);

}

}

}

}

import java.util.Scanner;

//A 子串相加

public class SonNNumbers {

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()) {

int n = cin.nextInt();

int m = cin.nextInt();

int ans = 0;

int[] a = new int[n];

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

a[i] = cin.nextInt();

if (i < m) {

ans += a[i];

}

}

int temp = ans;

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

temp = a[i] + temp - a[i - m];

if (temp > ans) {

ans = temp;

}

}

System.out.println(ans);

}

}

}

import java.util.Scanner;

//B 字符串分割 遍历String中的每一个char

public class charDivide {

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()) {

String str=cin.next();

for (int i=0;i<str.length();i++){

for (int j = str.length()-i-1;j>0;j--){

System.out.print(" ");

}

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

System.out.print(str.charAt(i));

}

System.out.println();

}

}

}

}

import java.util.Scanner;

//C 矩阵数字

public class Rectangle {

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()) {

int n = cin.nextInt();

for (int i = 1; i <= n; i++) {

System.out.print(1);

for (int j = 2; j <= i; j++) {

System.out.print(String.format("%4s",j));

}

for (int j = n-i; j >0; j--) {

System.out.print(String.format("%4s","1"));

}

System.out.println();

}

}

}

}

import java.util.Scanner;

//D SB用法 以及 int与String 的转换 字符串逆转

public class allInAll {

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()) {

int n =cin.nextInt();

int key = 10;

boolean flag = true;

while (key!=0){

String str = n+"";

int a=0;

int b=str.length()-1;

flag =true;

while (a<b){

if (str.charAt(a++)!=str.charAt(b--)){

flag=false;

break;

}

}

if (flag){

break;

}

String str2= new StringBuffer(str).reverse().toString();

int m=Integer.parseInt(str2);

n+=m;

key--;

}

if (flag){

System.out.println(n);

}else {

System.out.println("not");

}

}

}

}

import java.util.Scanner;

//E 转换 格式化输出

public class charNumber {

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()) {

String str = cin.nextLine();

int[] num = new int[10];

for (int i=0;i<str.length();i++){

if (Character.isDigit(str.charAt(i))){

num[Integer.parseInt(String.valueOf(str.charAt(i)))]++;

}

}

System.out.print(num[0]);

for (int i = 1; i < 10; i++) {

System.out.print(String.format("%3s",num[i]));

}

System.out.println();

}

}

}

import java.util.Scanner;

import java.util.StringTokenizer;

//多吃鱼少吃肉

public class Main {

public static void main(String arg[])

{

Scanner input = new Scanner(System.in);

while(input.hasNext())

{

int n =input.nextInt();

while(n!=0)

{

n--;

int sex=input.nextInt();

int tall=input.nextInt();

int weight=input.nextInt();

if(sex==1)

{

if(tall>130)

{

System.out.print("ni li hai!");

}

else if(tall<130)

{

System.out.print("duo chi yu!");

}

else

{

System.out.print("wan mei!");

}

if(weight>27)

{

System.out.print(" shao chi rou!\n");

}

else if(weight<27)

{

System.out.print(" duo chi rou!\n");

}

else

{

System.out.print(" wan mei!\n");

}

}

else

{

if(tall>129)

{

System.out.print("ni li hai!");

}

else if(tall<129)

{

System.out.print("duo chi yu!");

}

else

{

System.out.print("wan mei!");

}

if(weight>25)

{

System.out.print(" shao chi rou!\n");

}

else if(weight<25)

{

System.out.print(" duo chi rou!\n");

}

else

{

System.out.print(" wan mei!\n");

}

}

}

}

}

}

import java.util.\*;

//6翻了

public class Main {

public static void main(String arg[]){

String a;

char s[]=new char[1005];

Scanner sc=new Scanner(System.in);

while(sc.hasNext()){

a=sc.nextLine();

s=a.toCharArray();

int l=a.length();

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

if(s[i]==&apos;6&apos;){

int j=0;

while(i+j<l){

if(s[i+j]==&apos;6&apos;) j++;

else break;

}

if(j>3&&j<=9){

s[i]=&apos;9&apos;;

for(int k=i+1;k<i+j;k++) s[k]=&apos;#&apos;;

}

else if(j>9){

s[i]=&apos;2&apos;;

s[i+1]=&apos;7&apos;;

for(int k=i+2;k<i+j;k++) s[k]=&apos;#&apos;;

}

i+=j-1;

}

}

for(int i=0;i<l;i++) if(s[i]!=&apos;#&apos;) System.out.print(s[i]);

System.out.println();

}

}

}

//心理阴影面积

import java.util.Scanner;

public class Main {

public static void main(String args[])

{

Scanner input = new Scanner(System.in);

while(input.hasNext())

{

int a=input.nextInt();

int b=input.nextInt();

int ans = a\*100-b\*100;

System.out.println(ans/2);

}

}

}

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

while(scan.hasNext()){

String s = scan.nextLine();

int count[] = new int[30];

// 逆序

StringBuilder s2 = new StringBuilder();

for(int i=0; i < s.length(); i++){

if(i+1 == s.length() ||s.charAt(i+1) == ' '){

for(int j = i; j>=0 && s.charAt(j) != ' ' ; j--){

s2.append(s.charAt(j));

}

if(i+1 != s.length())

s2.append(' ');

}

}

System.out.println(s2);

// 输出最多的字母

String s1 = s.toLowerCase();

for (int i = 0; i < s1.length(); i++) {

if(s.charAt(i) != ' ')

count[s1.charAt(i) - 'a']++;

}

int pos = 0;

int max = 0;

for(int i=0; i<30; i++){

if(count[i] > max){

pos = i;

max = count[i];

}

}

char ch = (char) (pos + 'a');

System.out.println(ch);

}

}

}

import java.util.\*;

public class Main {

public static void main(String arg[]) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()){

int n=cin.nextInt();

int all=0;

while (n!=0){

int sum=1;

int ans=n;

while (ans!=0){

sum\*=ans;

ans--;

}

all+=sum;

n--;

}

System.out.println(all);

}

}

}

import java.util.\*;

public class Main {

public static void main(String[] args)

{

Scanner scanf = new Scanner(System.in);

while(scanf.hasNext()){

int n = scanf.nextInt();

int a[] = new int[n];

int x = 0;

int y = 0;

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

{

a[i] = scanf.nextInt();

}

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

{

if(a[k] % 2 == 0)

{

x += 1;

}

if(a[k] % 2 != 0)

{

y += 1;

}

}

System.out.println(y + " " + x);

}

}

}

import java.lang.reflect.Array;

import java.util.Arrays;

import java.util.Scanner;

public class Main {

public static String fun(int a){

String str = null;

if (a==0){

str="ling";

}else if (a==1){

str="yi";

}else if (a==2){

str="er";

}else if (a==3){

str="san";

}else if (a==4){

str="si";

}else if (a==5){

str="wu";

}else if (a==6){

str="liu";

}else if (a==7){

str="qi";

}else if (a==8){

str="ba";

}else if (a==9){

str="jiu";

}

return str;

}

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()){

int a = cin.nextInt();

String s = String.valueOf(a);

int b = s.length();

String[] str = new String[b+1];

int sum=0;

int f = a;

if (a<0){

System.out.print("fu ");

a=-a;

sum=1;

}

if (a==0){

System.out.println("ling");

continue;

}

while (a>0){

int temp = a%10;

str[sum++] = fun(temp);

a/=10;

}

if (f>0){

for (int i = b-1; i > 0; i--) {

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

}

System.out.println(str[0]);

}else {

for (int i = b-1; i > 1; i--) {

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

}

System.out.println(str[1]);

}

}

}

}

？ + ？ =?

import java.util.Scanner;

import java.util.regex.Pattern;

public class Main {

public static boolean isNumeric(String string){

Pattern pattern = Pattern.compile("[0-9]\*");

return pattern.matcher(string).matches();

}

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while (cin.hasNext()){

String str1 = cin.next();

String str2 =cin.nextLine();

str2=str2.trim(); //除去前面的空格

int flag =0;

if (!isNumeric(str1)){

str1="?";

flag++;

}

if (!isNumeric(str2)){

str2="?";

flag++;

}

if (flag==0){

System.out.println(Integer.parseInt(str1)+" + "+Integer.parseInt(str2)+" = "+(Integer.parseInt(str1)+Integer.parseInt(str2)));

}else {

System.out.println(str1+" + "+str2+" = ?");

}

}

}

}

//找出唯一数

class Solution {

public int singleNumber(int[] nums) {

int ans=0;

for(int i=0;i<nums.length;i++){

ans^=nums[i]; //异或运算

}

return ans;

}

}

• str.trim(); //去掉首尾空格

• str.replace(" ",""); //去除所有空格，包括首尾、中间

• str.replaceAll(" ", ""); //去掉所有空格，包括首尾、中间

• str.replaceAll(" +",""); //去掉所有空格，包括首尾、中间

• str.replaceAll("\\s\*", ""); //可以替换大部分空白字符， 不限于空格 ；

/JAVA将字符串拆分放入数租.(截取字符串：截取第i个位置到第i+1个位置的字符串)

String a = "abc";

String[] arr = new String[a.length()];

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

{ arr[i] = a.substring(i, (i+1));}

//数组输出：System.out.println(Arrays.toString(arr));

//如何将字串 String 转换成整数 int

1). int i = Integer.parseInt([String]);

2). int i = Integer.valueOf(my\_str).intValue();

//int 转换成字串 String

1.) String s = String.valueOf(i);

2.) String s = Integer.toString(i);

3.) String s = "" + i;

//String型转换为float||float型转换为String型

s.floatValue();||f.toString();

十进制转成十六进制：

Integer.toHexString(int i)

十进制转成八进制

Integer.toOctalString(int i)

十进制转成二进制

Integer.toBinaryString(int i)

十六进制转成十进制

Integer.valueOf("FFFF",16).toString()

八进制转成十进制

Integer.valueOf("876",8).toString()

二进制转十进制

Integer.valueOf("0101",2).toString()

进制转换如何写（二，八，十六）不用算法

Integer.toBinaryString

Integer.toOctalString

Integer.toHexString

字串出现的次数

public class SubStringCount {

　　public static void main(String[] args) {

　　　　String str1 = "abcjavanbajavasrcjavammmjava";

　　　　String str2 = "java";

　　　　int count = 0;

　　　　int index=str1.indexOf(str2);

　　　　while(index!=-1){

　　　　　　index = str2.length() + index;

　　　　　　count++;

　　　　　　index = str1.indexOf(str2, index);

　　　　}

　　　　System.out.println(count);

　　}

}

public class SearchStringEmp {

public static void main(String[] args) {

String strOrig = "Hello World!Hello World!";

int intIndex = strOrig.indexOf("Hello");

if(intIndex == - 1){

System.out.println("没有找到字符串 Hello");

}else{

System.out.println("Hello 字符串位置:" + intIndex);

}

}

}

质数查找

import java.util.Scanner;

public class MyFun2 {

public static void main(String[] args) {

// TODO Auto-generated method stub

Scanner in = new Scanner(System.in);

int x;

System.out.println("你要求取那个值的范围素数:");

x = in.nextInt();//统计x范围有多少素数；

boolean isPrime = true;//布尔类型;判断一个数是不是素数的标志位

int primeNUM = 0;

for(int i=2;i<=x;i++){

for(int j=2;j<i;j++){

if(i%j==0){

isPrime = false;

break;

}

}

if(isPrime){

System.out.print(i+"\t");

primeNUM++;

if(primeNUM%5==0)

System.out.println();

}

isPrime = true;

}

System.out.println();

System.out.println("素数"+ x +"范围内,一共有："+ primeNUM +"个素数");

}

}

1，charAt()

通过String类的charAt()的方法来获取字符串中的每一个字符，然后将其拼接为一个新的字符串

public static String CharAtreverse(String s) {

int length = s.length();

String reverse = "";

for (int i=0; i<length; i++)

reverse = s.charAt(i) + reverse;

return reverse;

}

2，toCharArray()

通过String的toCharArray()方法可以获得字符串中的每一个字符并转换为字符数组，然后用一个空的字符串从后向前一个个的拼接成新的字符串。

public static String reverseCharArray(String s) {

char[] array = s.toCharArray();

String reverse = "";

for (int i = array.length - 1; i >= 0; i--) {

reverse += array[i];

}

return reverse;

}

3，reverse()

通过StringBuiler或StringBuffer的reverse()的方法，此方法最快

public static String reverseStringBuilder(String s) {

StringBuilder sb = new StringBuilder(s);

String afterReverse = sb.reverse().toString()；

return afterReverse;

}

字符串替换：

1.replace（）

该方法的作用是替换字符串中所有指定的字符，然后生成一个新的字符串。经过该方法调用以后，原来的字符串不发生改变。例如：

String s = “abcat”；

String s1 = s.replace（‘a’，‘1’）；

该代码的作用是将字符串s中所有的字符a替换成字符1，生成的新字符串s1的值是“1bc1t”，而字符串s的内容不发生改变。

2. replaceAll（）

如果需要将字符串中某个指定的字符串替换为其它字符串，则可以使用replaceAll（），例如：

String s = “abatbac”；

String s1 = s.replaceAll（“ba”，“12”）；

该代码的作用是将字符串s中所有的字符串“ab”替换为“12”，生成新的字符串“a12t12c”，而字符串s的内容也不发生改变。

3.replaceFirst（）

如果只需要替换第一个出现的指定字符串时，可以使用replaceFirst（），例如：

String s = “abatbac”；

String s1 = s. replaceFirst （“ba”，“12”）；

该代码的作用是只将字符串s中第一次出现的字符串“ab”替换为字符串“12”，则字符串s1的值是“a12tbac”，字符串s的内容也不发生改变。

正则

只能输入整数，正数负数零都可以："^[+-]?[0-9]+$"或\d\*

整数或者小数：^[0-9]+\.{0,1}[0-9]{0,2}$

只能输入数字："^[0-9]\*$"。

只能输入n位的数字："^\d{n}$"。

只能输入至少n位的数字："^\d{n,}$"。

只能输入m~n位的数字：。"^\d{m,n}$"

只能输入零和非零开头的数字："^(0|[1-9][0-9]\*)$"。

只能输入有两位小数的正实数："^[0-9]+(.[0-9]{2})?$"。

只能输入有1~3位小数的正实数："^[0-9]+(.[0-9]{1,3})?$"。

只能输入非零的正整数："^\+?[1-9][0-9]\*$"。

只能输入非零的负整数："^\-[1-9][]0-9"\*$。

只能输入长度为3的字符："^.{3}$"。

只能输入由26个英文字母组成的字符串："^[A-Za-z]+$"。

只能输入由26个大写英文字母组成的字符串："^[A-Z]+$"。

只能输入由26个小写英文字母组成的字符串："^[a-z]+$"。

只能输入由数字和26个英文字母组成的字符串："^[A-Za-z0-9]+$"。

只能输入由数字、26个英文字母或者下划线组成的字符串："^\w+$"。

public class三角形面积 {

public static void main(String[] args) {

Scanner scan =new Scanner(System.in);

System.out.println("输入三角形的三边");

int a = scan.nextByte();

int b = scan.nextByte();

int c = scan.nextByte();

float s =(a+b+c)/2f;

float S = (float) Math.sqrt(s\*(s-a)\*(s-b)\*(s-c));

if (a+b>c && b+c>a && a+c>b){

System.out.println(S);

}

else{

System.out.println("不成立三角形");

}

}

}

import java.util.Scanner;

public class CommonDiviser

{

private static Scanner input = new Scanner(System.in);

public static void main(String[] args)

{

int n = input.nextInt();

int m = input.nextInt();

int zdgys = result(n, m);

System.out.println("最大公约数是:" + zdgys);

System.out.println("最大公倍数是:" + n \* m / zdgys);

input.close();

}

public static int result(int n, int m)

{

// 余数

int k = 0;

// 使n>m

if (n < m)

{

int t = 0;

t = n;

n = m;

m = t;

}

// 直到余数为0

while (m != 0)

{

k = n % m;

n = m;

m = k;

}

System.out.println(n);

return n;

}

}

Eclipse用法

Ctrl+1 快速修复

Ctrl+Shift+F 格式化当前代码

Debug 全局 继续 F8

导包 ctrl+Shift+/

eclipse中补全代码快捷键,默认Alt+/

找到左侧菜单栏

1.首先，点击导航栏中的window选项。

2.在该选项中，点击show view选项。

3.在show view选项中，如果发现有project Exlorer选项就直接点击该选项，就会自动出现左侧的项目栏了。如...

4.在show view的右拉菜单中点击最下方的other选项。

5.找到general选项，并在该选项中就可以找到projectexplorer选项