**import** java.util.\*;

**import** java.math.BigDecimal;

**public** **class** BD {

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

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

**int** t=cin.nextInt();

**int** cas=1;

List<BigDecimal> x = **new** ArrayList<BigDecimal>();

**while**(t>0){

x.clear();

t--;

**int** ans=0;

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

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

String str = cin.next();

String[] da = str.split(":");

BigDecimal a=**new** BigDecimal(da[0]);

BigDecimal b=**new** BigDecimal(da[1]);

BigDecimal c=a.add(b);

BigDecimal d=a.divide(c,1000,BigDecimal.***ROUND\_HALF\_EVEN***);

x.add(d);

}

Collections.*sort*(x);

BigDecimal temp=**new** BigDecimal(0);

BigDecimal tt=**new** BigDecimal(1);

**int** ii=0;

**while**(ii<x.size()&&temp.add(x.get(ii)).compareTo(tt)==-1){

temp=temp.add(x.get(ii));

ii++;

ans++;

}

System.***out***.println("Case #"+cas+": "+ans);

cas++;

}

}

}

/\*

Collections.sort(list,new Comparator<User>(){

public int compare(User arg0, User arg1) {

return arg0.getOrder().compareTo(arg1.getOrder());

}});\*/

import java.util.\*;

import java.math.\*;

public class Main {

public static BigInteger qpow(BigInteger a,BigInteger b,BigInteger m) {

  BigInteger ans = BigInteger.ONE;

  a = a.mod(m);

  while(!(b.equals(BigInteger.ZERO))) {

  if((b.mod(BigInteger.valueOf(2))).equals(BigInteger.ONE)) {

  ans = (ans.multiply(a)).mod(m);

  b = b.subtract(BigInteger.ONE);

  }

  b = b.divide(BigInteger.valueOf(2));

  a = (a.multiply(a)).mod(m);

}

return ans;

}

public static void main(String[] args) {

Scanner cin = new Scanner(System.in);

while(cin.hasNextBigInteger()) {

BigInteger n = cin.nextBigInteger();

BigInteger m = cin.nextBigInteger();

BigInteger a = n.multiply(m);

System.out.println(a);

}

}

}

/\*

  //d为int型，a,b,c都为大数

c=a.add(b);             //  相加

c=a.subtract(b);       //    相减

c=a.multiply(b);          // 相乘

c=a.divide(b);        // 相除取整

c=a.gcd(b);          //  最大公约数

c=a.remainder(b);   //  取余

c=a.mod(b);         // a mod b

c=a.abs();           // a的绝对值

c=a.negate();        // a的相反数

c=a.pow(d);           // a的b次幂        d为int型

c=a.max(b);           //  取a，b中较大的

c=a.min(b);                //  取a，b中较小的

d=a.compareTo(b);      //  比较a与b的大小   d=-1小于 d=0等于 d=1大于  d为int型

a.equals(b);            //  判断a与b是否相等    相等返回true  不相等返回false

\*/

/\*

 //進制轉換

String st = Integer.toString(num, base); //把int型num当10进制的数转成base进制数存入st中    (base <= 35).

int num = Integer.parseInt(st, base); //把st当做base进制，转成10进制的int

(parseInt有两个参数,第一个为要转的字符串,第二个为说明是什么进制).

BigInter m = new BigInteger(st, base); // st是字符串，base是st的进制.

BigInteger a;

int b;

Stringc;

a=cin.nextBigInteger(b);   //以b进制读入一个大数赋值给a

c=a.toString(b);          // 将大数a以b进制的方式赋给字符串c

a=newBigInteger(c, b);  //把c 当做“b进制“转为十进制大数赋值给a

\*/

import java.math.BigDecimal;

public class Main {

    public static void main(String[] args) {

        // 构造函数

        BigDecimal a = new BigDecimal(10);

        BigDecimal b = new BigDecimal(10.5);

        BigDecimal c = new BigDecimal(123456789101222L);

        BigDecimal d = new BigDecimal("-123456.31");

        //四则运算

        System.out.println(a.add(b).add(c).add(d));

        System.out.println(a.subtract(b).subtract(a));

        System.out.println(a.multiply(b).multiply(c));

        System.out.println(c.divide(a).divide(new BigDecimal(20)));

        System.out.println(d.divide(a));

        System.out.println(d.doubleValue());

        System.out.println(a.compareTo(a));// 0

        System.out.println(a.compareTo(b));// -1

        System.out.println(b.compareTo(a));// 1

        // System.out.println(aa.equals(aa));// true？

    }

}