Bupt 13 级新手专题训练②高精度和几个小数学题

这次的题目其实主要就是想要大家熟悉一下 Java 的高精度编写带来的快感，短而简洁省时省力，附带的水题做着玩吧

A - Integer Inquiry

Java 秒过

```
import java.util.*;
import java.math.*;
import java.io.*;

public class Main {

	public static void main(String[] args) {

		Scanner in = new Scanner(new
BufferedInputStream(System.in));
		BigInteger sum = BigInteger.ZERO;
		BigInteger x;
		while(true) {
			x = in.nextBigInteger();
			if ( x.equals(BigInteger.ZERO) )
				break;
			sum = sum.add(x);
		}
		System.out.println(sum);
	}
}
```

B – Product

Java 秒过

```
import java.util.*;
import java.math.*;
import java.io.*;

public class Main {

	public static void main(String[] args) {
		BigInteger a;
	BigInteger b;
	Scanner in = new Scanner(new
BufferedInputStream(System.in));
		while( in.hasNext() )
	{
	    a = in.nextBigInteger();
	    b = in.nextBigInteger();
	    System.out.println(a.multiply(b));
	}
	}
}
```

C – Overflow

Double 转化，相当妙的思想

```
#include<stdio.h>
#include<stdlib.h>
#define INF  2147483647

int main()
{
    char num1[1000],num2[1000],op;
    double a,b;
    while(scanf("%s %c %s",num1,&op,num2)!=EOF){
        printf("%s %c %s\n",num1,op,num2);
        a=atof(num1);
        b=atof(num2);
        if(a>INF)printf("first number too big\n");
        if(b>INF)printf("second number too big\n");
        if(op=='+'&&a+b>INF)printf("result too big\n");
        if(op=='*'&&a*b>INF)printf("result too big\n");
    }
    return 0;
}
```

只能说这题用 Java 相当考验 Java 的使用细节

```
import java.util.*;
import java.math.*;
import java.io.*;

public class Main {

       public static void main(String[] args) {
              BigInteger a, b, ans = BigInteger.ZERO;
        String x, op, y;
        Scanner in = new Scanner(new
BufferedInputStream(System.in));

        while(in.hasNext()){
```

```
            ans = BigInteger.ZERO;
        x = in.next();
        op = in.next();
        y = in.next();

        a = new BigInteger(x);
        b = new BigInteger(y);

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

        if( op.equals( "+" ) ){ans = a.add(b);}

        if( op.equals( "*" )){ans = a.multiply(b);}


if(a.compareTo(BigInteger.valueOf(Integer.MAX_VALUE))>0){
System.out.println("first number too big");}

if(b.compareTo(BigInteger.valueOf(Integer.MAX_VALUE))>0){
System.out.println("second number too big");}

if(ans.compareTo(BigInteger.valueOf(Integer.MAX_VALUE))>0
){System.out.println("result too big");}
        }
        }
}
```

D – Exponentiation

Java 秒过

不过注意如果 0.几的话，去除首位的 0

```
import java.util.*;
import java.math.*;
import java.io.*;

public class Main {

	public static void main(String[] args) {
		Scanner in = new Scanner(new
BufferedInputStream(System.in));
		BigDecimal r;
		int n;
		while (in.hasNext())
		{
			r = in.nextBigDecimal();
			n = in.nextInt();
			r = r.pow(n);
			String ans =

r.stripTrailingZeros().toPlainString();		//忽略后面的结

尾 0，格式化输出

			if ( ans.startsWith( "0." ) )
				ans = ans.substring(1);
			System.out.println(ans);
		}
	}
}
```

E - If We Were a Child Again

Java 秒过

```
import java.util.*;
import java.math.*;
import java.io.*;

public class Main {

	public static void main(String[] args) {
		BigInteger a, b;
	String x, op, y;
	Scanner in = new Scanner(new
BufferedInputStream(System.in));

	while(in.hasNext()){
	    x = in.next();
	    op = in.next();
	    y = in.next();

	    a = new BigInteger(x);
	    b = new BigInteger(y);

	    if( op.equals("%") )
	       System.out.println(a.mod(b) );
	    else if( op.equals("/") )
	       System.out.println( a.divide(b) );
	}
	}
}
```

F - The Other Two Trees

水题，已知正方形对顶点，求另外两个对顶点。

```
#include<cstdio>
#include<cmath>

int main(){
	double x1, y1, x2, y2, a, b;
	while( scanf( "%lf%lf%lf%lf", &x1, &y1, &x2,
&y2 ) !=EOF )
    {
		if( x1 == x2 && y1 == y2 )
			printf("Impossible.\n");
		else
        {
            a = x2 - x1;
            b = y2 - y1;
            printf( "%.10lf %.10lf %.10lf %.10lf\n",  ( a + b )
/ 2.0 + x1, ( b - a ) / 2.0 + y1,( a - b ) / 2.0 + x1, ( b
+ a ) / 2.0 + y1 );
        }
    }
	return 0;
}
```

G – ClockHands

题目好长，没想到最后一句直接亮瞎我，什么上面的历史都很有趣，但和这题没什么关系。。就是求分针和时针的角度差(来自世界深深的恶意？)

```
#include<cstdio>
#include<cmath>
#include<cstdlib>
int main() {
        double H, M, ans;
        while( scanf( "%lf:%lf", &H, &M ) !=EOF )
    {
        if( !H && !M ) break;
                ans = fabs( H * 30.0 + M * 0.5 - M * 6.0 );
        printf( "%.3lf\n", ans >= 180 ? 360 - ans : ans );
        }
        return 0;
}
```

H - Inscribed Circles and Isosceles Triangles

等腰三角形，做一个内切圆，然后再以这个圆的顶部位置做切线，形成一个新的等腰三角形，再重复这个过程，直到圆半径小于一个值，然后累加这些半径。

```
#include<cstdio>
#include<cmath>
#define Pi 4.0 * atan( 1.0 )
#define dis 0.000001
int main(){
        int t;
        double B, H, Sum, R, L;
        scanf( "%d", &t );
        while( t-- )
    {
        Sum = 0;
        scanf( "%lf %lf", &B, &H );
        L = sqrt( B * B / 4 + H * H );
        R = B * H / ( B + 2 * L );
        while( R >= dis )
        {
            Sum += R;
            H -= 2*R;
            R = B * H / ( B + 2 * L );
        }
        printf( "%13.6lf\n", 2 * Pi * Sum );
        if( t ) printf( "\n" );
        }
        return 0;
}
```

I – Billiard

想不明白的时候超难，想明白了发现时间过了好久。。。就是无论球用什么角度出射，总水平距离和竖直距离都可以直接算出来，然后求出比值，用反三角函数求得角度，然后求总长度，再干些你想干的。

```
#include<cstdio>
#include<cmath>

const double Pi = 4.0 * atan(1.0);

int main()
{
    double a, b, s, m, n, v, vetic, horiz;
    while( scanf( "%lf %lf %lf %lf %lf", &a, &b, &s, &m, &n ) !=
EOF )
    {
        if( !a && !b && !s && !m && !n ) break;
        vetic = a * m;
        horiz = b * n;
        printf( "%.2lf %.2lf\n", 180.0 * atan( horiz / vetic )
/ Pi, (sqrt( vetic * vetic + horiz * horiz ) ) / s );
    }
    return 0;
}
```

J - Myacm Triangles

【暴力枚举】

有一堆点，找出里面最大的三角形来，这个三角形要求不包含其他任一点。

```
#include<cstdio>
#include<cstdlib>
#include<cstring>
struct point
{
    char a[2];
    int x;
    int y;
}p[20];

int main()
{
    int i, j, k, l, n, area, p1, p2, p3, s1, s2, s3, flag, Max;
    while( scanf( "%d", &n ) != EOF )
    {
        memset( p, 0, sizeof(p) );
        if( !n ) break;
        Max = 0;
        for( i = 0; i < n; i++ )
        {
            getchar();
            scanf( "%c%d%d", &p[i].a[0], &p[i].x, &p[i].y );
        }
        for( i = 0; i < n; i++ )
            for(j = i + 1; j < n; j++ )
                for( k = j + 1; k < n; k++ )
                {
                    flag = 1;
                    area = abs( ( p[k].y - p[i].y )*( p[j].x -
p[i].x )-( p[j].y - p[i].y )*( p[k].x - p[i].x ) );
```

```
                for( l = 0; l < n; l++ )
                {
                    if( l == i || l == j || l == k ) continue;
                    s1 = abs(( p[k].y - p[l].y )*( p[j].x -
p[l].x )-( p[j].y - p[l].y )*( p[k].x - p[l].x ));
                    s2 = abs(( p[l].y - p[i].y )*( p[j].x -
p[i].x )-( p[j].y - p[i].y )*( p[l].x - p[i].x ));
                    s3 = abs(( p[k].y - p[i].y )*( p[l].x -
p[i].x )-( p[l].y - p[i].y )*( p[k].x - p[i].x ));
                    if( s1 + s2 + s3 == area ) { flag = 0;
break; }
                }
                if( area >= Max && flag ) { Max = area; p1
= i; p2 = j; p3 = k; }
            }
            printf( "%c%c%c\n", p[p1].a[0], p[p2].a[0],
p[p3].a[0] );
    }
    return 0;
}
```