

```

/* Problem 4--Elliptical Orbit
   You don't need as much calculus as you might think.  An ellipse is a
   "squashed circle," so find the corresponding areas of an unsquashed
   circle, and then squash the angles. */

import java.io.*;
import java.util.*;

public class prob4 {

    private static Scanner in;
    private static PrintWriter out;
    private static int cs;

    public static void main (String[] args) throws Exception {

        in = new Scanner (new File ("prob4.in"));
        out = new PrintWriter ("prob4.out");
        cs = 1;
        double a, b;
        while ((a = in.nextDouble()) > 0) { //Read in a and b
            b = in.nextDouble();
            int div = in.nextInt();
            double f = Math.sqrt (a*a-b*b); //Compute f
            out.print ("Case "+(cs++)+" : The angles are\r\n");
            for (int i = 1; i < div; i++) computeportion (a,b,f,((double)i)/div);
            out.print ("\r\n"); //Find each area
        }
        in.close ();
        out.close ();
    }

    //computeportion computes the angle that finds the appropriate area
    public static void computeportion (double a, double b, double f,
                                       double frac) {

        b /= a; f /= a; //Scale the shape so that a is 1.
        double Asc = Math.PI * frac; //Unsquashed area to find
        double x1 = 2*Math.PI*frac; //Estimated angle.
        double x0 = 0;
        do {
            x0 = x1; //Use Newton's method to find angle
            double func = x0/2 - 0.5*f*Math.sin(x0) - Asc;
            double funcp = 0.5 - 0.5*f*Math.cos(x0);
            double funcpp = 0.5*f*Math.sin(x0);
            x1 = x0 - func*funcp / (funcp*funcp - func*funcpp);
        } while (Math.abs (x1-x0) > 1e-15);
        double angle = Math.atan2 (b*Math.sin(x0),Math.cos(x0)-f)*180/Math.PI;
        angle = angle < 0 ? 360+angle : angle; //Squash angle
        int ai = (int)(angle*3600+0.5+1e-8); //Round to nearest second
        int deg = ai/3600; //Convert to degrees, minutes, and seconds
        int min = (ai-deg*3600)/60;
        int sec = ai-deg*3600-min*60;
        out.print (deg+" deg "+min+" min "+sec+" sec\r\n");
    }
}

```