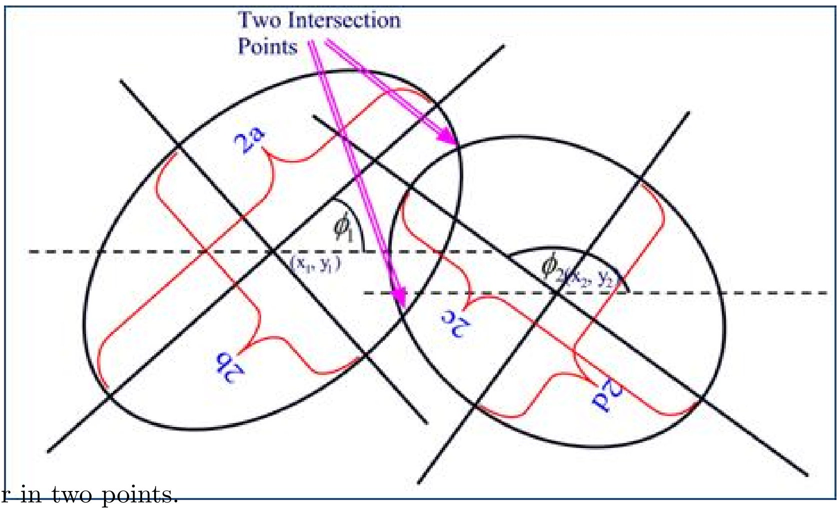
Finding the intersection of two arbitrary ellipses can be tricky. Actually even the intersection of two axis-parallel ellipses can be tricky, let alone ellipses that are in rotated position like the two shown on the right:



For two arbitrary ellipses, the following cases can occur:

(a) They do not intersect.

(b) They intersect each other in two points.

(c) They intersect each other in four points.

(d) They touch each other in one point.

(e) They touch each other in one point and intersect in another two points.

(f) They touch each other in two points.

In this problem you will not have to detect whether two ellipses touch each other as the given ellipses will not touch each other: They will either intersect each other in 2 or 4 different points or will not intersect. In other words you will have to deal with only situation (a), (b) and (c).

Input

There will be around 6000 cases for this problem.

Each case contains 10 integers x_1, y_1, a, b, ϕ_1, x_2, y_2, c, d, ϕ_2. Here

a) (x_1, y_1) = Coordinate of the center of first ellipse $(0 \leq x_1, y_1 \leq 1000)$.

b) (x_2, y_2) = Coordinate of the center of the second ellipse $(0 \leq x_2, y_2 \leq 1000)$.

c) a = length of semi-major axis of first ellipse $(10 \leq a \leq 500)$.

d) b = length of semi-minor axis of first ellipse $(10 \leq b \leq 500)$.

e) c = length of the semi-major axis of second ellipse $(10 \leq c \leq 500)$.

f) d = length of the semi-minor axis of second ellipse $(10 \leq d \leq 500)$.

g) ϕ_1 = angle (in degrees) the major axis of the first ellipse creates with the positive direction of x-axis $(0 \leq \phi_1 \leq 360)$.

h) ϕ_2 = angle (in degrees) the major axis of second ellipse creates with the positive direction of x-axis $(0 \leq \phi_2 \leq 360)$.

Input is terminated by a case where the value of a, b, c and d are negative. This case should not be processed.

Output

For each case, produce two, three or five lines of outputs. The first line contains the serial of output. If the two ellipses do not intersect then in the next line print the line '`Do_not_intersect.`' If the ellipses intersect in two points then print two floating-point numbers in two lines and if the ellipses intersect in four points then print four floating-point numbers in four lines. These floating-point numbers are actually summation of abscissa and ordinate of the intersection points. These numbers should be printed in ascending order and should have eight digits after the decimal point. Errors less than 10^{-5} will be ignored.

Sample Input

```
755 216 313 109
130 671 997 355
85 338
861 654 216 319
300 936 910 338
483 318
653 819 271 144
325 145 959 145
49 185
653 819 -1 -1
325 145 959 -1
-1 185
```

Sample Output

```
Case 1:
Do_not_intersect.
Case 2:
1173.83963384
1754.28733987
Case 3:
Do_not_intersect.
```