O. Arrow

time limit per test: 0.5 seconds
memory limit per test: 64 megabytes
input: standard input
output: standard output

Petya has recently started working as a programmer in the IT city company that develops computer games.

Besides game mechanics implementation to create a game it is necessary to create tool programs that can be used by game designers to create game levels. Petya's first assignment is to create a tool that allows to paint different arrows on the screen.

A user of this tool will choose a point on the screen, specify a vector (the arrow direction) and vary several parameters to get the required graphical effect. In the first version of the program Petya decided to limit parameters of the arrow by the following: a point with coordinates (px, py), a nonzero vector with coordinates (vx, vy), positive scalars $a, b, c, d, a > c$.

The produced arrow should have the following properties. The arrow consists of a triangle and a rectangle. The triangle is isosceles with base of length a and altitude of length b perpendicular to the base. The rectangle sides lengths are c and d. Point (px, py) is situated in the middle of the triangle base and in the middle of side of rectangle that has length c. Area of intersection of the triangle and the rectangle is zero. The direction from (px, py) point to the triangle vertex opposite to base containing the point coincides with direction of (vx, vy) vector.

Enumerate the arrow points coordinates in counter-clockwise order starting from the tip.

Input

The only line of the input contains eight integers px, py, vx, vy ($-1000 \le px, py, vx, vy \le 1000, vx^2 + vy^2 > 0$), a, b, c, d ($1 \le a, b, c, d \le 1000, a > c$).

Output

Output coordinates of the arrow points in counter-clockwise order. Each line should contain two coordinates, first x, then y. Relative or absolute error should not be greater than 10^{-9}.

Examples

input

```
8 8 0 2 8 3 4 5
```

output

```
8.000000000000 11.000000000000
4.000000000000 8.000000000000
6.000000000000 8.000000000000
6.000000000000 3.000000000000
10.000000000000 3.000000000000
10.000000000000 8.000000000000
12.000000000000 8.000000000000
```