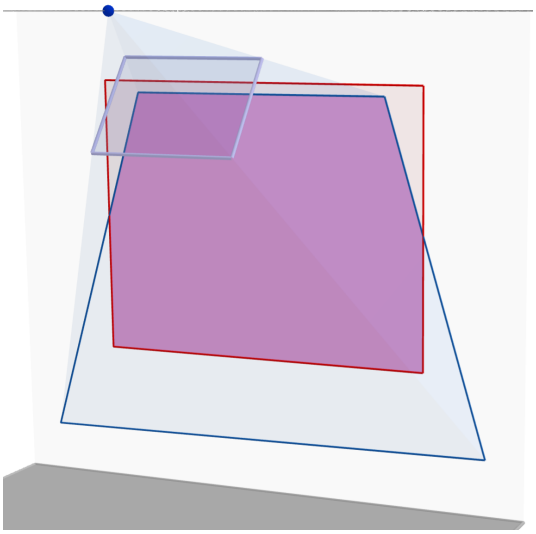
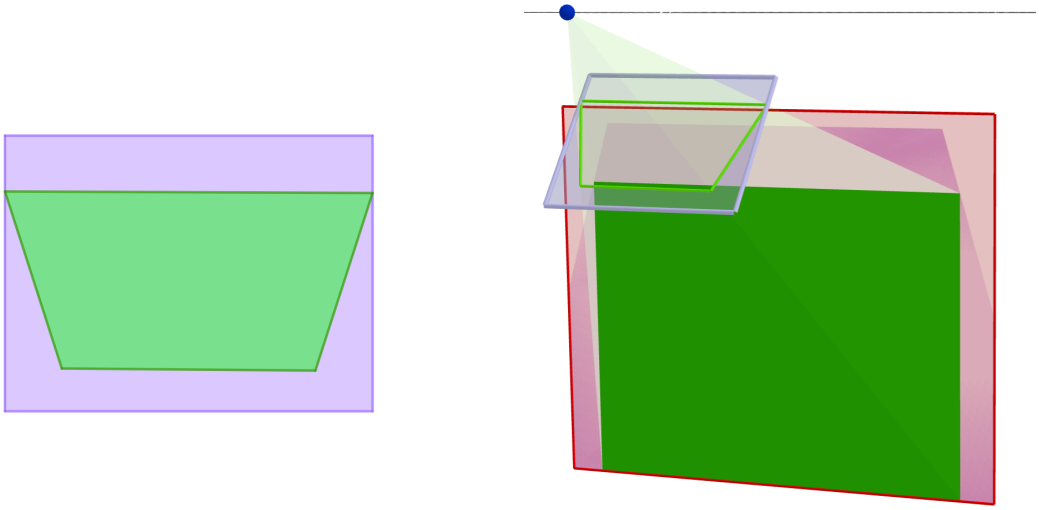
Keystone Correction

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 megabytes

As a huge fan of anime, Haru brought a projector recently to enjoy the home theater with a big screen on the wall. However, the projector is, well, cost-effective: it costs very little and functions like…"just works." It provides limited adjustment functionalities and is screwed to a fixed stand; hence, misalignment often occurs. Fortunately, there is a manual keystone correction feature that can align the corners of the image to the given points, ensuring that the output is as close to the original aspect ratio as possible.



An example of a misaligned projector.



The keystone correction and the final result. This is Haru's solution for the first sample case.

The projector is described as a point light source, and four corners of the lens rectangle of width w and height h in the lens plane. The lens rectangle defines the aspect ratio of the desired picture, denoted as $w : h$ – for example, the aspect ratio after simplification is usually $4 : 3$ or $16 : 9$ in practice. The light beam is bounded by the lens rectangle and finally projects to the screen, which is a rectangle on the wall plane.

Haru's challenge is to obtain four anchor points in the lens rectangle for the keystone correction, so that the corrected projection is a rectangle that keeps the aspect ratio of the lens rectangle with the bottom

(with the smallest maximum z coordinate) edge parallel to the xOy plane and that falls entirely inside the screen.

Tired of manually adjusting silhouettes on the wall, Haru decided to write a program to implement a keystone correction algorithm to calculate the maximum area of a good projection rectangle.

Input

The input consists of multiple test cases. The first line contains an integer T ($1 \le T \le 1000$), the number of test cases. For each test case, there are eight lines:

- In the first four lines, the i-th line contains integers x_i, y_i, z_i ($|x_i|, |z_i| \le 100$, $1 \le y_i \le 100$, $\max\{x_1, x_4\} < \min\{x_2, x_3\}$, $\max\{z_1, z_2\} < \min\{z_3, z_4\}$) representing a corner (x_i, y_i, z_i) of the lens rectangle.
- In the next four lines, the i-th line contains integers u_i, v_i, w_i ($|u_i|, |w_i| \le 1000$, $(\max_{j=1}^{4} y_j) < v_i \le 1000$, $u_1 = u_4 < u_2 = u_3$, $v_1 = v_4 < v_2 = v_3$, $w_1 = w_2 < w_3 = w_4$) representing a corner (u_i, v_i, w_i) of the screen rectangle.

Each four points form a rectangle and are listed in the order of the bottom-left, bottom-right, top-right, and top-left corner (see the above constraints for the x and z coordinates). The light source is at $(0, 0, 0)$. Let the distance between the bottom-left and bottom-right corner of the lens be w, and the distance between the bottom-left and top-left corner of the lens be h. Then, $1 \le \frac{w}{h} \le 4$. There are some extra constraints to ensure that the projection is done *roughly* along the front direction (the positive y axis):

- The wall plane is perpendicular to xOy, and the bottom edge of the screen is parallel to xOy.
- The lens rectangle and the point $(0, 0, 0)$ are both on the same side of the wall and are at least 1 unit away from the wall. The distance from the image plane to $(0, 0, 0)$ is at least 1 unit.
- The angle between the bottom edge of the screen and the x axis will not exceed $\frac{\pi}{18} = 10^\circ$. The angle between the bottom edge of the lens and the x axis will not exceed $\frac{\pi}{18} = 10^\circ$.
- The beam projects through the lens to the wall plane, lighting up an area of at most 10^6 square units. The intersection of the screen and the uncorrected projection is at least 1 square unit.

Output

Output the answer as a decimal real number on a line for each test case. The answer will be considered correct if it has an absolute or relative error not exceeding 10^{-6}.

Example

standard input	standard output
2 -20 23 -36 20 23 -36 20 41 -12 -20 41 -12 -50 80 -100 50 79 -100 50 79 -20 -50 80 -20 -1 2 4 1 2 4 1 3 5 -1 3 5 -20 6 0 20 6 0 20 6 20 -20 6 20	5450.0656584647 5.6568542495