One Two-One Chips

Input file:	`standard input`
Output file:	`standard output`
Time limit:	7 seconds
Memory limit:	256 megabytes

In the Algorithmia club, a new game invented by Marckess has become popular. Each person chooses a point in space, and chips will appear in the space. The closest person to a chip will receive one, while the farthest person from the others will lose one. The second closest person will receive two chips, and the second farthest will lose two chips.

However, Tony, Fabian, and Dani's team has obtained relevant information about the appearance of the chips, although it was acquired through somewhat underhanded means.

Despite this, they have only been able to approximate the appearance of each chip independently within a cube with a uniform distribution.

Since they do not know how to estimate the expected number of chips obtained per point, they have asked for your help to determine which point to choose.

Compute the expected value of chips per point.

Input

The first line of the input will contain two integers, n and m ($3 \leq n \leq 100$ and $1 \leq m \leq 100$), where n represents the number of points and m represents the number of cubes.

The following n lines will each contain three integers x_i, y_i and z_i ($x_i, y_i, z_i \leq |10^3|$), representing the coordinates of a point.

The subsequent m lines will each contain four integers x_i, y_i, z_i and d_i ($x_i, y_i, z_i \leq |10^3|$ and $1 \leq d_i \leq 2 \times 10^3$), representing a point at the lower left corner of the cube, along with a magnitude d, which denotes the length of the cube's side. The cube extends from (x, y, z) to $(x + d, y + d, z + d)$.

No two points will have identical coordinates, and all cubes are contained within the limits ($|10^3|$). For n > 10 all points were randomly generated with a uniform distribution.

Output

Print n numbers, each representing the expected value of chips for a given point. Your answer will be considered correct if its absolute or relative error does not exceed 10^{-4}.

Examples

standard input	standard output
3 1 0 1 1 1 1 1 2 1 1 0 0 0 2	-0.25 0.5 -0.25
9 1 -4 -4 -4 4 -4 -4 4 4 -4 -4 4 -4 4 4 4 -4 4 4 -4 -4 4 4 -4 4 0 0 0 -3 -3 -3 2	1.5 0 -0.666666667 0 -1 -0.666666667 0 -0.666666667 1.5
9 1 -4 -4 -4 4 -4 -4 4 4 -4 -4 4 -4 4 4 4 -4 4 4 -4 -4 4 4 -4 4 0 0 0 -3 -3 -3 6	-0.145833333 -0.145833333 -0.145833333 -0.145833333 -0.145833333 -0.145833333 -0.145833333 -0.145833333 1.16666667

Note

The sum of expected values is zero.