Plane and a Line

Filename: INTERSECT

A line in three dimensions can be defined by two distinct points. A plane in three dimensions can be defined by three non-collinear points. The intersection of a line and a plane can fall into one of three cases:

1) no intersection
2) a point of intersection
3) the line itself lies on the plane and is the intersection

Your program, given a plane P and a line L, should distinguish between these three intersection cases. If the intersection is a single point, your program should determine this point.

The Input:

The first line of the input file will contain a single integer n, specifying the number of input cases. The next 2n lines of the input file will specify each of the n cases of input. Each case will be specified by 2 lines of input. The first line will contain 6 integers x_1, y_1, z_1, x_2, y_2, z_2, where (x_1, y_1, z_1) and (x_2, y_2, z_2) are two distinct points on the line L. The second line will contain 9 integers x_3, y_3, z_3, x_4, y_4, z_4, x_5, y_5, z_5, where (x_3, y_3, z_3), (x_4, y_4, z_4), and (x_5, y_5, z_5) are three distinct non-collinear points on the plane P.

The Output

The k^{th} data set should have the heading:

```
Data Set #k:
```

The next line should follow one of the following three formats based upon the intersection of the given line and plane:

```
There is no intersection.
The intersection is the point (x, y, z).
The line lies on the plane.
```

If there is a single point of intersection, the coordinates of that point should be rounded to one place after the decimal point.

The output for each data set should be separated by a blank line.

Sample Input:

```
3
1 1 1 8 -7 1
3 9 0 -1 3 0 4 5 0
1 3 4 -1 2 1
2 1 1 1 2 -1 3 1 0
3 0 3 6 0 0
2 1 1 1 2 -1 3 1 0
```

Sample Output:

```
Data Set #1:
There is no intersection.

Data Set #2:
The intersection is the point (-1.0, 2.0, 1.0).

Data Set #3:
The line lies on the plane.
```