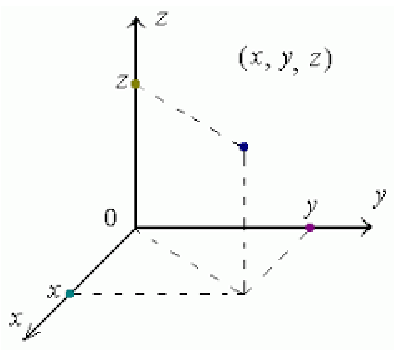
Problem 3: Best Sensor

Author: Graham Pateman & Jacob Starratt, Dartmouth, Nova Scotia, Canada

Problem Background

Accurately determining the location of a ship or aircraft with modern sensors is a very data intensive task. Different sensors “see” entities in different ways and at different times. The entire process involves gathering data from the sensors (called “reports”) and aligning that data in order to process reports that were all made at the same time. Once the data is “time aligned” in this manner, the location of the object must be determined. There are many ways to do this.

The Cartesian coordinate system uses a set of numerical coordinates to specify a unique point in space. For one dimension, we can use a line, or axis, called X. The origin of this line is at 0; negative values indicate positions to the left of the origin, and positive values indicate positions to the right of it. Most problems using coordinates in this packet use a two-dimensional system; this adds a second axis, Y, perpendicular to the X axis. The combination of two values, (X,Y), establishes a unique position along both axes.



In reality, however, we exist in three dimensions; the X axis covers positioning from left to right; the Y axis covers positioning to forward and back; and the Z axis, perpendicular to both, covers positioning up and down. These coordinates are written as (X,Y,Z).

All sensors will report what they see using a 3-dimensional Cartesian coordinate, with the ship or aircraft the sensor is mounted to at the origin, (0,0,0). Each sensor will report a slightly different location, and so each object being tracked has a “best sensor.” The best sensor is the sensor that reports the location closest to the average location amongst all of the sensors.

Problem Description

You’re working with Lockheed Martin’s Skunk Works to upgrade the sensors on the Lockheed U-2 “Dragon Lady” reconnaissance aircraft. As part of this effort, you’re adding a logging system to track what the best sensor is during each scan. These logs can be used to track the performance of each sensor and identify those that should be replaced due to inaccuracy.

Identifying the best sensor is an important task, and requires two steps. First, the average location must be calculated from amongst the various sensor reports. Each of the three coordinates - X, Y, and Z - must be separately averaged to calculate an average point. Once this is done, the distance between the average point and each of the sensor reports must be calculated. Whichever sensor has a report closest to the average is the “best sensor.”

The distance between two points in a three-dimensional coordinate system can be calculated using this formula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

Sample Input

The first line of your program’s input, **received from the standard input channel**, will contain a positive integer representing the number of test cases. Each test case will include:

- A line containing a positive integer, **S**, indicating the number of sensors
- **S** lines, each containing the following information, separated by spaces:
  - A string representing the sensor’s name, containing uppercase letters, underscores (_), and numbers
  - An integer, representing the X coordinate reported by the sensor
  - An integer, representing the Y coordinate reported by the sensor
  - An integer, representing the Z coordinate reported by the sensor

```
2
3
SENSOR_1 4 5 6
SENSOR_2 3 4 5
SENSOR_3 5 6 7
4
SENSOR_4 -10 10 -5
SENSOR_5 -11 9 -4
SENSOR_6 -12 11 -5
SENSOR_7 -9 8 -6
```

Sample Output

For each test case, your program must print a line containing the following information, separated by spaces:

- The name of the best sensor
- A decimal value representing the X coordinate of the average coordinate, rounded to three decimal places and including any trailing zeroes
- A decimal value representing the Y coordinate of the average coordinate, rounded to three decimal places and including any trailing zeroes

- A decimal value representing the Z coordinate of the average coordinate, rounded to three decimal places and including any trailing zeroes

```
SENSOR_1 4.000 5.000 6.000
SENSOR_4 -10.500 9.500 -5.000
```