

12th South African Regional ACM Collegiate Programming Contest

Sponsored by IBM

16 October 2010

Problem B - Blue Balloon Cloud chamber

Problem Description

You are a particle physicist, and you have built yourself a cloud chamber to monitor ionizing radiation generated by a radioactive source. The cloud chamber resembles a fish tank, and is filled with supersaturated water vapour. When an alpha or beta particle passes through the chamber it ionises the mixture, causing tiny clouds to form along the path of the particle. You have set up a camera to take photographs of these vapour trails so that you can count the number of trails formed in an experiment.

After the experiment, you digitize the photographs to generate a set of coordinates where tiny clouds of mist have been observed. These tiny clouds lie approximately in a straight line, but various factors add a little bit of noise to the detection of their positions. Your input data (the coordinates of the tiny clouds) resembles Figure 1. Unfortunately, you receive this information in random order, so you have to associate each coordinate with the appropriate line before you can count the number of lines, and hence the number of particle trails observed.

Your task is to analyse the coordinate information to determine the minimum number of particle trails that agree with the observed points in a given experiment.

Input

Your input will consist of an arbitrary number of records, with each record representing a complete problem. Each record starts with the number of samples n , followed by n pairs of (x, y) coordinates, e.g.,

```
2 0.1 0.7 0.5 0.9
```

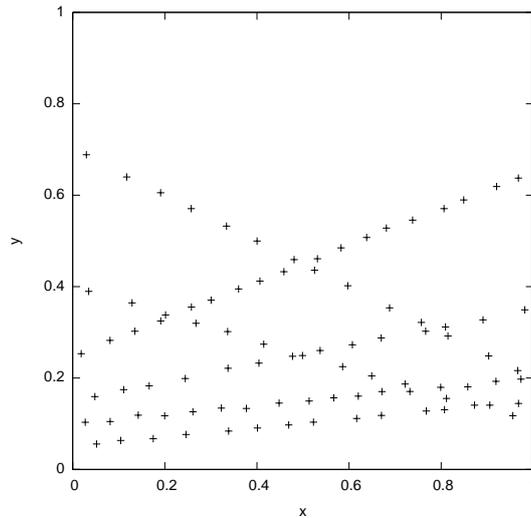


Figure 1: Sample particle trail image with 6 particles (90 tiny clouds were observed)

represents two sets of (x, y) coordinates, namely $(0.1, 0.7)$ and $(0.5, 0.9)$. Your program should not be sensitive to the type of white space used between coordinate pairs in the input (spaces and/or newlines may be used). The final input record is followed by a line containing just the value “-1”, i.e., $n = -1$.

There are some constraints to simplify the problem:

1. All x and y coordinates are in the range $[0, 1]$.
2. Each particle trail will be represented by at least 7 coordinates, but these may be shared by other particle trails.
3. Any given input record will contain between 1 and 6 particle trails (inclusive). The value of n (the number of points) can not necessarily be used to deduce the number of particle trails. The range of n values that your program must support is $n \in 7..90$.
4. All particle trails can be described using a straight line of the form

$$y = mx + c.$$

The observed y -coordinates are corrupted by zero-mean Gaussian noise. The standard deviation of the Gaussian noise is 0.0025, but the distribution is truncated to the range $[-0.0075, 0.0075]$. This implies that your y -coordinates are effectively

$$y = mx + c + \delta,$$

where δ is randomly selected from the range $[-0.0075, 0.0075]$ for each cloud centre coordinate in your input data set.

5. Each line contains at least two points of which the y -coordinate has not been corrupted, so that these two points fall exactly on the ideal line $y = mx + c$.
6. The x -coordinates of all points belonging to a single particle trail are selected randomly from the range $[0, 1]$, but are guaranteed to fill the range $[0, 1]$ approximately evenly, as illustrated in Figure 1.

7. The input data may contain particle trails that are parallel or near-parallel. In such cases, you may assume that the lines are at least 0.02 units apart (i.e., their c values differ by at least 0.02). If the trails are near-parallel, you may assume that they will be further than 0.02 apart for at least 75% of the x -range. Note that lines may intersect.

Output

For each input record, your program must produce a line of output of the form

`detected p trails`

where p is the number of particle trails that your program has detected.

Sample Input

```
22 0.649450 0.290338 0.764918 0.275861 0.945091 0.289006 0.293607 0.159421 0.039155
0.203915 0.492198 0.243136 0.044274 0.227341 0.392089 0.239219 0.932939 0.290596
0.713961 0.242792 0.210581 0.141786 0.806211 0.259329 0.208577 0.241857 0.794285
0.296602 0.090540 0.116380 0.481133 0.269113 0.937954 0.312788 0.328992 0.248345
0.566546 0.213309 0.646714 0.261703 0.464371 0.192177 0.233014 0.223301
22 0.058963 0.111542 0.695777 0.239418 0.559235 0.259827 0.608457 0.222840 0.227600
0.222575 0.431530 0.237943 0.473586 0.194717 0.549408 0.211299 0.365677 0.235803
0.374104 0.172854 0.957333 0.291869 0.973621 0.295743 0.141686 0.128337 0.626350
0.265812 0.282122 0.152690 0.060561 0.206056 0.196960 0.141115 0.869109 0.277804
0.161950 0.217790 0.798431 0.256207 0.851678 0.286084 0.773092 0.277309
-1
```

Sample Output

```
detected 3 trails
```

```
detected 2 trails
```

Time Limit

60 seconds