# Automated Intersection point detection in 2D Line Graphs

Problem Statement

The analysis of graphical data is a common task in various fields, including finance, economics, and data science. A fundamental operation in graph analysis is the identification and counting of intersection points between lines. Manually identifying these points can be time-consuming and prone to error, especially with complex graphs. You task is to develop an algorithm and automate the process of accurately detecting and counting intersection points in 2D line graph images.

Sample image where there are 2 intersection points -
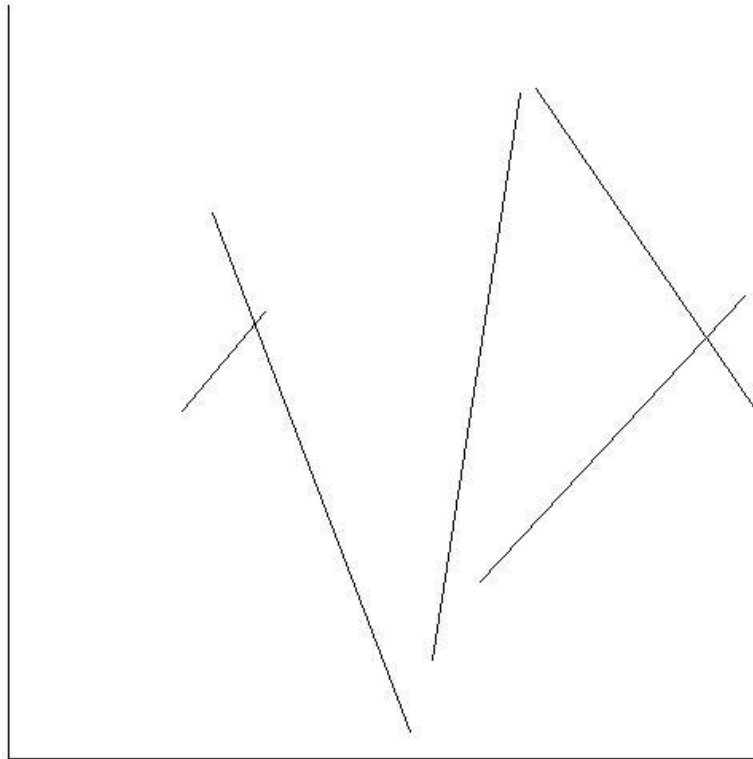
Input Format

Guidelines

1. First line in the standard input represents the image to be used eg. image1.jpg

Sample Input

image3.jpg

Reading the Input

You can read the input as standard input.

To load the image, you can load it as following:

Python

```
//read from standard input filename = input() // get file if os.path.isfile(filename):
# Open the file
with open(filename, 'r') as file:
...
```

Java

```
import java.io.File;
...
// read from standard input Scanner scanner = new Scanner(System.in); String query =
scanner.nextLine(); // Get the image file
File file = new File("image1.jpg");
```

Constraints

Image constraints

1. A JPEG image file of size 512x512 pixels representing a 2D line graph. The image will contain:

2. An x-axis and a y-axis is provided. Intersection with x-axis and y-axis is *not* counted as intersection.

3. A maximum of 10 lines, each 1 pixel thick.

4. All lines and intersection points are visible within the image boundaries.

Output Format

Guidelines

- Please output only the number of intersection points in standard output

Sample Output

```
2
```

Sample Input 0

```
image1.jpg
```

Sample Output 0

```
0
```

Sample Input 1

```
image4.jpg
```

Sample Output 1

```
2
```