Topcoder SRM 641, D1, 250-Pointer
"TrianglesContainOrigin"

James S. Plank
EECS Department
University of Tennessee

CS494 Class
January 21, 2016
The problem

- You are given the $(x,y)$ values of points on a two-dimensional grid:

Example 1:
- $(-1,-1)$
- $(-1,1)$
- $(1,2)$
- $(2,-1)$
The problem

- Of all the triangles with these points as endpoints, how many have the origin inside?

- Example 1: There are 4 triangles, two of which include the origin.
Prototype and Constraints

- **Class name**: TrianglesContainOrigin
- **Method**: count()
- **Parameters**:
  
<table>
<thead>
<tr>
<th>x</th>
<th>vector &lt;int&gt;</th>
<th>X coordinates</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>vector &lt;int&gt;</td>
<td>Y coordinates</td>
</tr>
</tbody>
</table>

- **Return Value**: long long
- **Constraints**:
  - $x$.size() == $y$.size() ≤ 2500.
  - $x$ and $y$ values between -10,000 and 10,000.
  - No three values co-linear
  - No two values co-linear with the origin.
Brain-dead enumeration of triangles

• Let \( n = x\text{.size}() \).

• Then the number of triangles is:

\[
\binom{n}{3} = O(n^3)
\]

• When \( n = 2500 \), this is 2,590,630,000. Too slow.

• Our solution can be \( O(n^2) \), but not much slower.
The Key Insight

- Draw a line from the origin to each point, and then calculate the angles of adjacent lines:
The Key Insight

- Consider a triangle – it will include the origin if and only if each of the angles of lines from the origin is less than 180:
The Strategy

- Enumerate all pairs of points whose angle to the origin is less than 180 degrees:
- For a given pair, there is a minimum and maximum angle that the third point can have.

Any point between here and here works.

Any point between here and here is > 180 degrees to point 1.

Any point between here and here is > 180 degrees to point 2.
The Algorithm

- For each point, calculate the point's angle $\alpha$ from the origin.

- Insert the points into a map keyed by angle.
- Also insert the points keyed by angle+360.
- Number the points in the map by ascending angle.

- For each point $x < 360$ and each point $y$ whose angle to $x$ is less than 180 degrees, use `upper_bound()` to find:
  - The smallest point whose angle is $> 180$ to $x$.
  - The smallest point whose angle is $> 180$ to $y$.
  - The difference in vals is the number of points that can complete the triangle!

- Sum the triangles and divide by 3 for the answer!
Let's look at example 3
Let's look at example 3

Consider points 0 and 4.
Let's look at example 3

Consider points 0 and 4.

Upper-bound will find points 9 and 13.

Therefore, there are four points that can complete the triangle with points 0 and 4.
Running Time:

- There are $O(n^2)$ pairs of points.
- Upper bound is $O(\log(n))$.
- So the program is $O(n^2\log(n))$. 

MacBook Pro
2.4 GHz
-O3 optimization
How did the Topcoders Do?

- 580 competitors
- 285 (48%) submitted a solution.
- 216 (76%) of the submissions were correct.
- That's 38% - I suspected this one would be hard!