POLYGON ORIENTATION

ANTHONY NGUYEN

Polygon Orientation

We define a polygon as a polygon without any intersecting edges or collinear vertices. If given the order of the vertices of the polygon, can we determine the orientation (whether clockwise or counter-clockwise)?

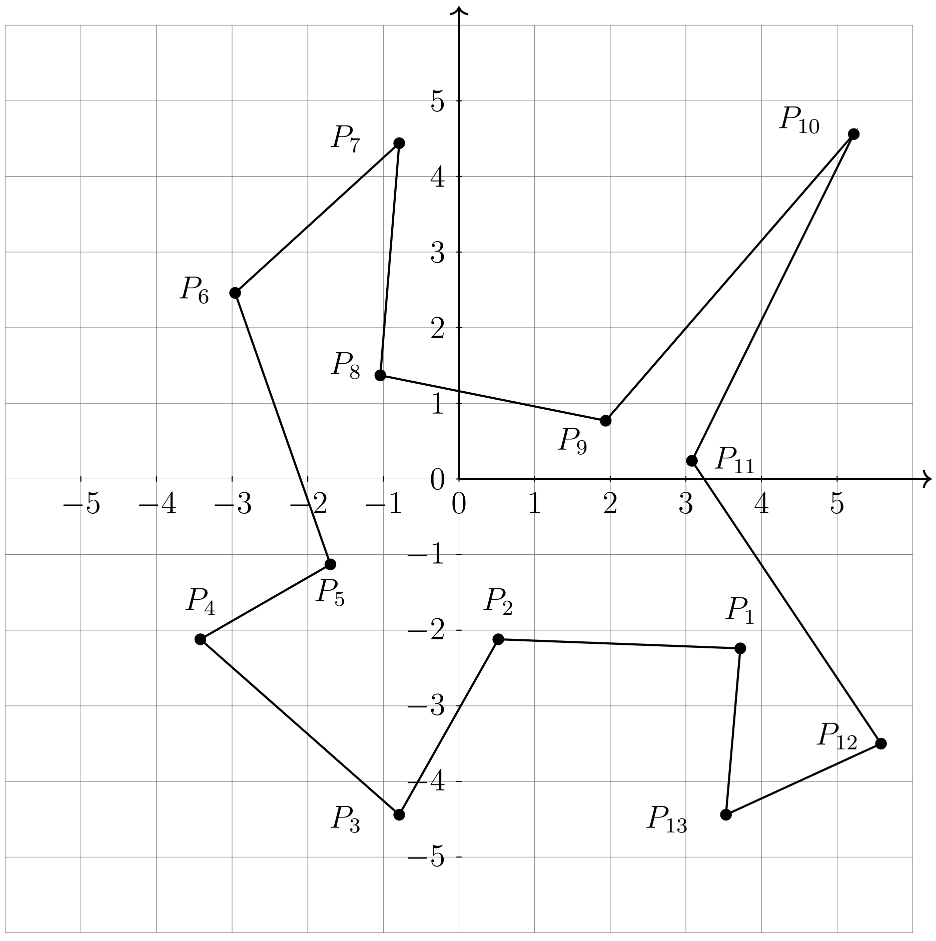


Figure 1. Polygon, with vertices ordered from $P_1 ... P_{13}$

From observation, it can be seen that this polygon is in a clockwise orientation, but how do we mathematically come to this conclusive (beyond an "eye test").

I believe that using the orientation of the bottom vertex (and its adjacent edges), we can determine the orientation of the whole polygon. For this particular polygon, these vertices are P_3 and P_{13}.

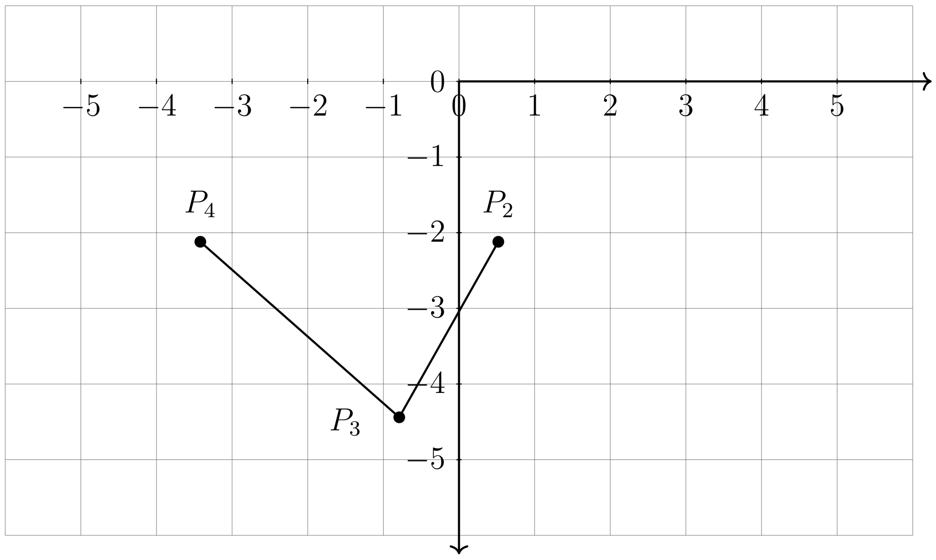


FIGURE 2. P_3, with its adjacent edges to vertices P_2 and P_4

In regards to vertex P_3, all other vertices either must have an equal or higher y-value because P_3 is the bottom-most vertex by definition. For this polygon (or any n-sided polygon, in general), there must be an edge between the last vertex to the first vertex (to complete and close the polygon). As a result, the rest of the vertices will complete the cycle above P_3. This "finishing of cycle" from P_3 to P_3 will be in the same orientation as the current cycle of $P_2 - -P_4$ because the orientation of how the cycle approaches P_3 cannot change.

Using our current example, the orientation of $P_2 - -P_4$ is clockwise. As a result, the polygon is oriented clockwise and will finish in a clockwise direction. If it were to finish in a counter-clockwise direction, the cycle would have to "loop around" to reach P_3. Because our polygon cannot have intersecting edges and the cycle cannot go below the y-axis of P_3 (due to P_3's bottom-most property), a counter-clockwise orientation is not possible. Thus, the polygon's orientation must be clockwise.

One particular example of note is the "Crescent Moon" polygon, shown below.

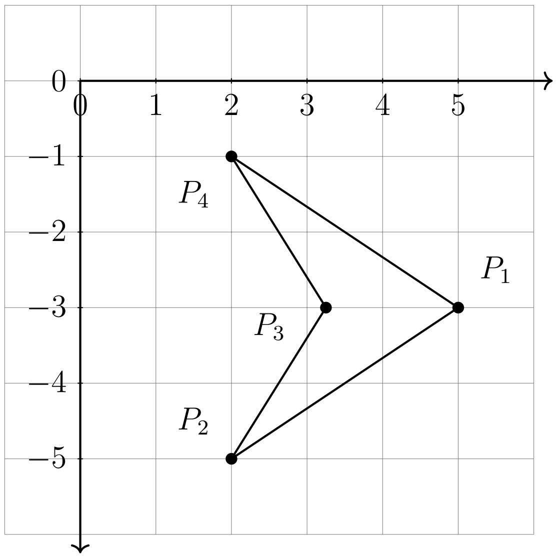


FIGURE 3. Clockwise "Crescent Moon" Polygon

Although $P_2 - -P_3$ appears to a counter-clockwise vector, the clockwise orientation of the cycle of $P_1 - -P_3$ matches the clockwise orientation of this entire polygon.

This hypothesis also works in determining counter-clockwise polygons.

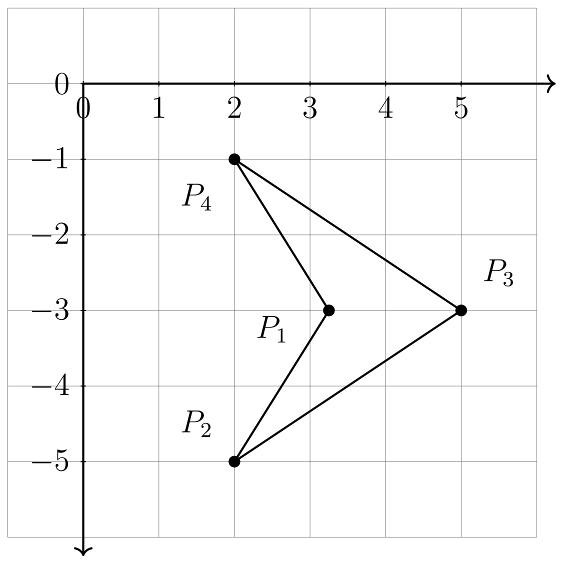


FIGURE 4. Counter-Clockwise "Crescent Moon" Polygon

Now, we know that the orientation of the bottom-most vertex matches the orientation of the entire polygon. From here, however, how can we mathematically determine the orientation of the bottom-most vertex?

To do so, we use the sign of the magnitude of the z-axis of the cross-product of the two vectors emerging from this particular vertex, from the polygon in Figure 1.

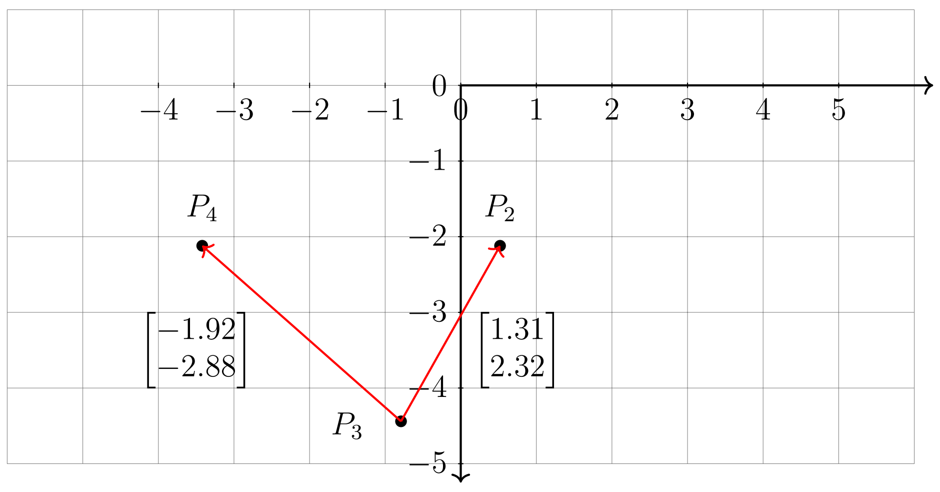

−4
−3
−2
−1
0
1
2
3
4
5
0
−1
−2
−3
−4
−5
P_4
P_2
P_3
$\begin{bmatrix}-1.92\\-2.88\end{bmatrix}$
$\begin{bmatrix}1.31\\2.32\end{bmatrix}$

FIGURE 5. P_3 and its emerging vectors

Then, we determine the magnitude of the z-axis of their cross-product, in the order of the vector toward the previous vertex to the vector toward the next vertex.

$$\overrightarrow{P_3P_2} \times \overrightarrow{P_3P_4} = \begin{vmatrix} X & Y & Z \\ 1.31 & 2.32 & 0 \\ -1.92 & -2.88 & 0 \end{vmatrix}$$

$$= (0,\ 0,\ 0.6816)$$

$$\rightarrow 0.6816\,\widehat{z}$$

For the case of P_3, the z-axis is observed to be positive.

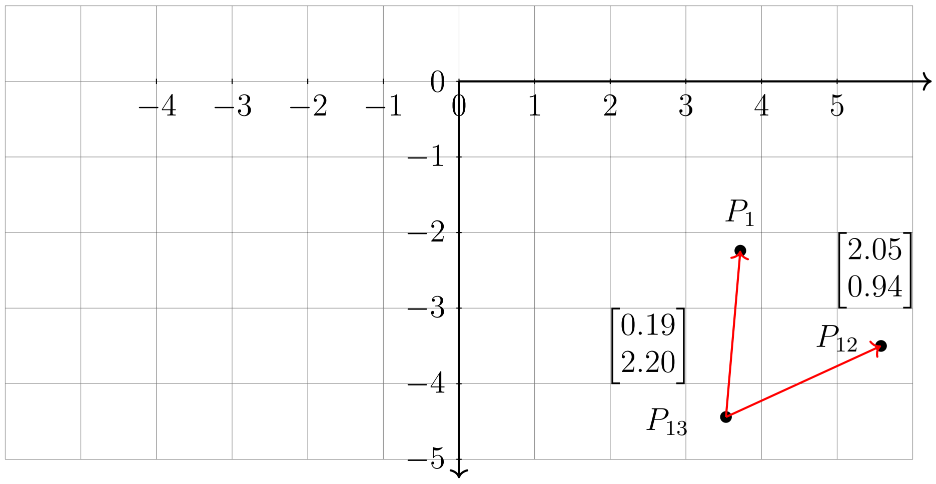
$$\overrightarrow{P_3P_2} \times \overrightarrow{P_3P_4} = \begin{vmatrix} X & Y & Z \\ -1.92 & -2.88 & 0 \\ 1.31 & 2.32 & 0 \end{vmatrix}$$

$$= (0,\ 0,\ -0.6816)$$

$$\rightarrow -0.6816\,\widehat{z}$$

If we were to reverse the order of the vertices adjacent to P_3 to create a counter-clockwise orientation, the z-axis is observed to be negative.

Since P_3 and P_{13} have the same y-axis value, we can examine another case for clockwise orientation.



P_1
P_{12}
P_{13}
$\begin{bmatrix}2.05\\0.94\end{bmatrix}$
$\begin{bmatrix}0.19\\2.20\end{bmatrix}$

FIGURE 6. P_{13} and its emerging vectors

$$\overrightarrow{P_{13}P_{12}} \times \overrightarrow{P_{13}P_1} = \begin{vmatrix} X & Y & Z \\ 2.05 & 0.94 & 0 \\ 0.19 & 2.20 & 0 \end{vmatrix}$$

$$= (0,\ 0,\ 4.3314)$$

$$\rightarrow 4.3314\,\widehat{z}$$

Using this cross-product, a positive value for the z-axis implies a clockwise orientation. Conversely, a negative value for the z-axis implies a counter-clockwise orientation.

One particular issue, however, would be the case where the cross-product was equal to 0. This happens when the three chosen points (from the set of vertices with the lowest y-value) are collinear.

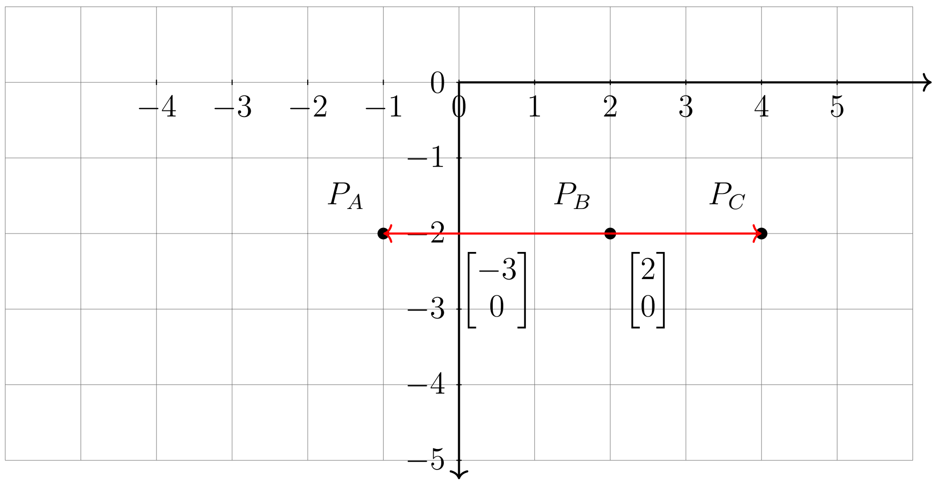

P_A
P_B
P_C
$\begin{bmatrix}-3\\0\end{bmatrix}$
$\begin{bmatrix}2\\0\end{bmatrix}$

FIGURE 7. P_B and its emerging collinear vectors

$$\overrightarrow{P_BP_A} \times \overrightarrow{P_BP_C} = \begin{vmatrix} X & Y & Z \\ -3 & 0 & 0 \\ 2 & 0 & 0 \end{vmatrix}$$

$$= (0,\ 0,\ 0)$$

$$\rightarrow 0\,\widehat{z}$$

To solve this, we perform a sorting algorithm on this set of bottom-most points, then choosing the vertex with the highest x-value. As a result, one of its adjacent vertices would have a y-value greater than the lowest value in the polygon. Thus, the cross-product of its adjacent vectors would no longer be 0.

One hypothesis that we would like to observe is the case where we can get our result by running this "cross-product" algorithm on all vertices. Here, each vertex would put a "vote" for either type of orientation (CW vs CCW). Then, whatever side had the greater amount of "votes" would be the determined orientation.

This would not work, as observed with the following polygon.

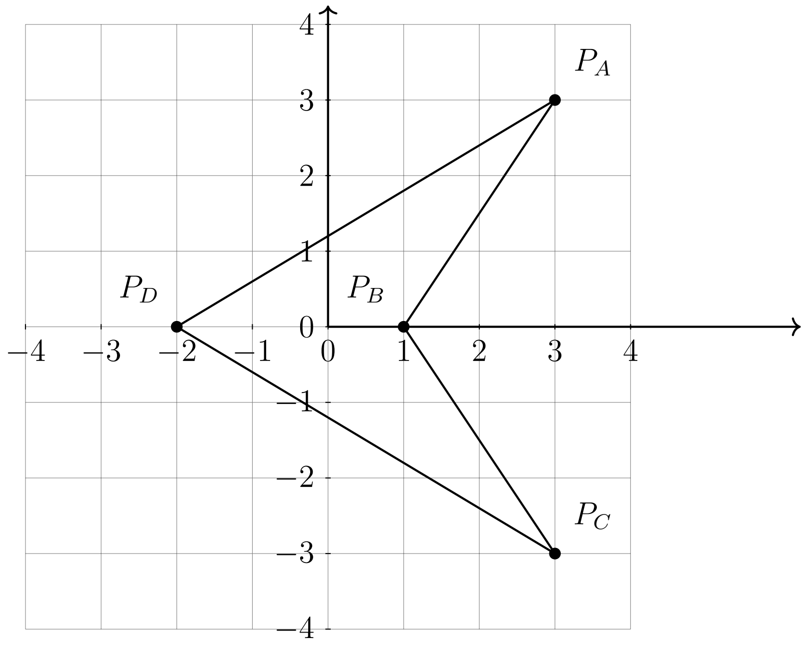


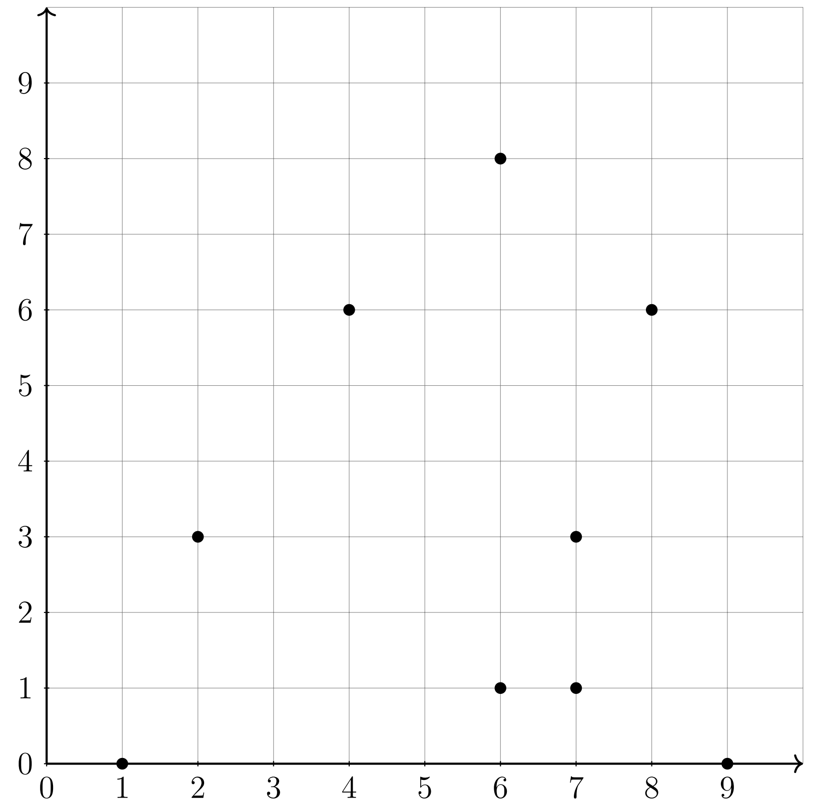
FIGURE 8. P_9

If we apply the cross-product algorithm to each vertex, we can observe 2 votes for Clockwise and 2 votes for Counterclockwise for this clockwise polygon. As a result, this method of voting would not be practical.

Simulation and Testing

In order to test our algorithm, we needed a way to mass-produce and test simulated polygons under a given orientation. By doing so, we could examine the algorithm's accuracy.

To create a singular polygon, our first step was to generate n sets of *(x,y)* coordinates, before throwing out any cases with duplicate vertices. This will serve as our list of vertices for our n-sided polygon.



Coordinates
(1, 0)
(2, 3)
(4, 6)
(6, 1)
(6, 8)
(7, 1)
(7, 3)
(8, 6)
(9, 0)

Figure 9. Plot of vertices within randomly-generated 9-sided polygon

Now, we have our list of vertices, but we do not know the ordering of these vertices to ensure that we have a proper polygon.

In order to account for this, we must first calculate the center, before shifting the center to *(0,0)* and offsetting each vertex by the center shift distance.

$$\frac{1}{9}\sum_{i=1}^{9}(x_i,y_i) =$$

$$\frac{(1,0)+(2,3)+(4,6)+(6,1)+(6,8)+(7,1)+(7,3)+(8,6)+(9,0)}{9} =$$

$$(5.6, 3.1)$$

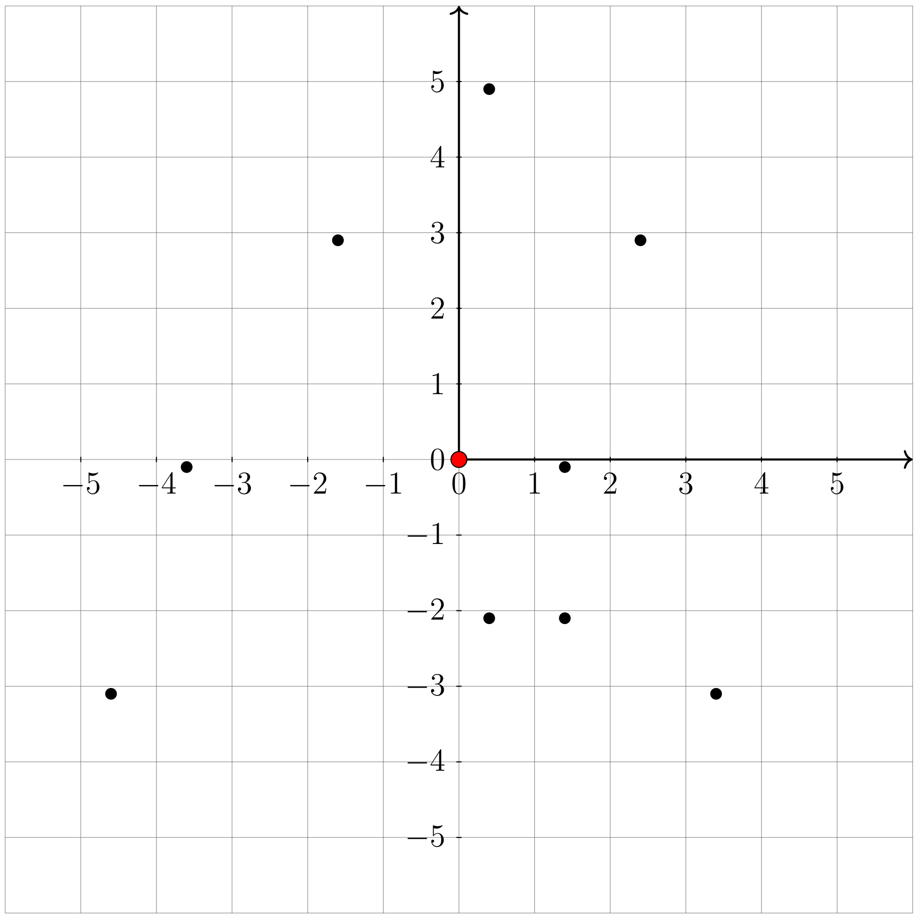
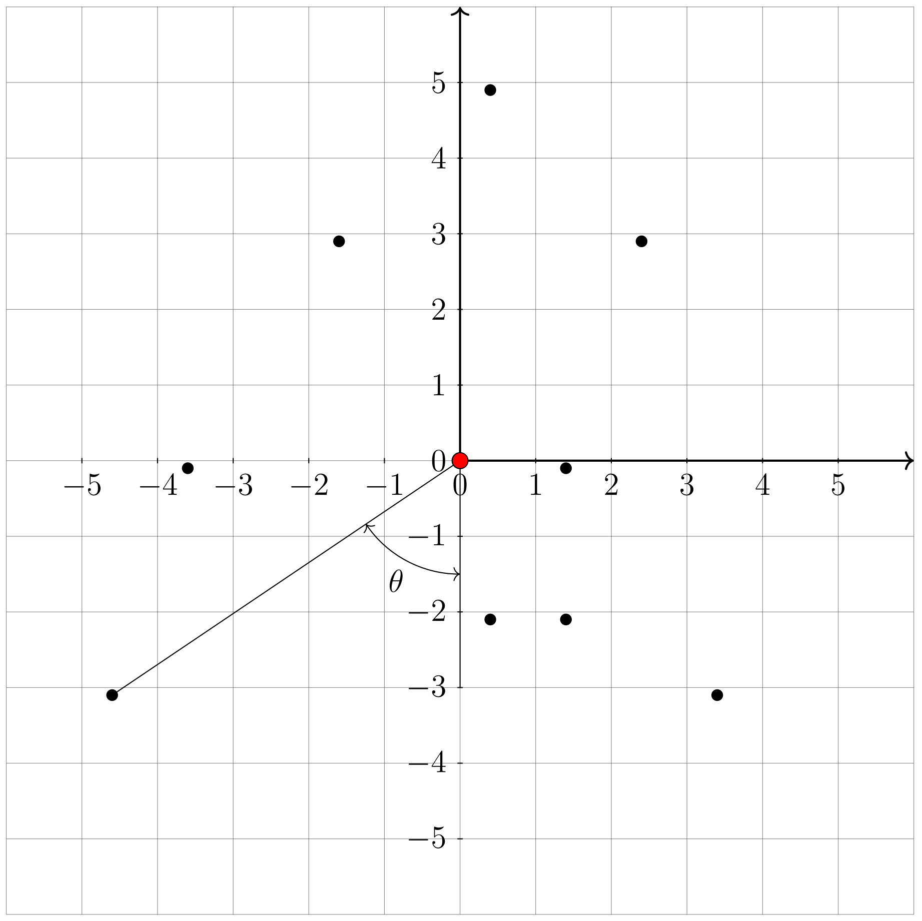


Figure 10. Plot of vertices within randomly-generated 9-sided polygon, after shifting

To determine the order of these vertices, we connect each vertex with a line to *(0,0)*. Then, we sort the vertices by the angle created by the vertex, *(0,0)*, and the negative y-axis.



Order	Coordinates
1	(1, 0)
2	(2, 3)
3	(4, 6)
4	(6, 8)
5	(8, 6)
6	(7, 3)
7	(9, 0)
8	(7, 1)
9	(6, 1)

FIGURE 11. The corresponding vertices of the original polygon, in CW order

With this method, our polygon will automatically be in clockwise orientation. If we wanted to create a polygon with counter-clockwise orientation, we would reverse the ordering for vertices $2 - n$ (in this case, $n = 9$).

Order	Coordinates
1	(1, 0)
2	(2, 3)
3	(4, 6)
4	(6, 8)
5	(8, 6)
6	(7, 3)
7	(9, 0)
8	(7, 1)
9	(6, 1)

Order	Coordinates
1	(1, 0)
2	(6, 1)
3	(7, 1)
4	(9, 0)
5	(7, 3)
6	(8, 6)
7	(6, 8)
8	(4, 6)
9	(2, 3)

Figure 12. Vertices for 9-sided polygon with CW vs CCW orientation

Once we have our polygon's vertices, we will randomly select an orientation and export a file with the polygon's ordered vertices (based on the table above).

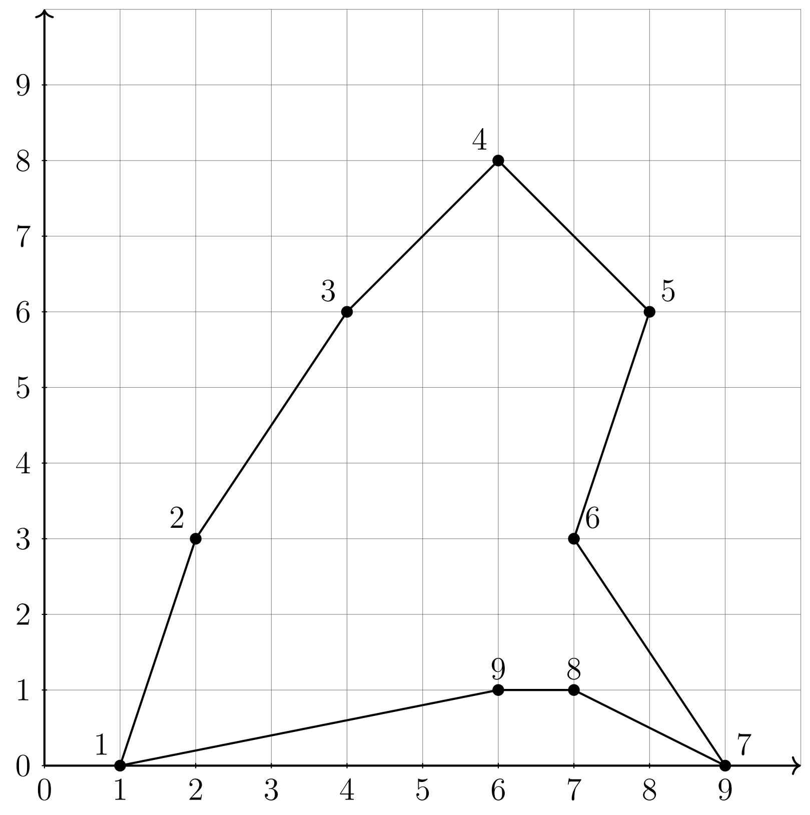
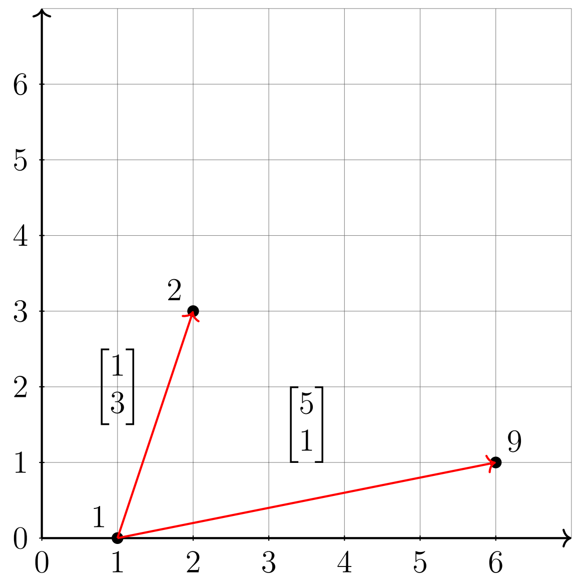


Figure 13. 9-sided ordered polygon, in CW orientation

Now that we have our polygon, we must use it to test our algorithm.



6
5
4
3
2
1
0
0 1 2 3 4 5 6
2
$\begin{bmatrix}1\\3\end{bmatrix}$
$\begin{bmatrix}5\\1\end{bmatrix}$
9
1

Figure 14. Vertex 1 and its emerging vectors

$$\overrightarrow{V_1V_9} \times \overrightarrow{P_1P_2} = \begin{vmatrix} X & Y & Z \\ 5 & 1 & 0 \\ 1 & 3 & 0 \end{vmatrix}$$

$$= (0,\ 0,\ 14)$$

$$\rightarrow 14\ \widehat{z}$$

Because our Z-value from the vector product is positive, this confirms that our polygon is clockwise.

GitHub Repository: https://github.com/anthony-nguyen-04/PolygonOrientation