MatGeo Presentation - Problem 2.9.15

EE25BTECH11064 - Yojit Manral

Question

If the points **A** $(2, 0)$, **B** $(6, 1)$, and **C** (p, q) form a triangle of area 12 square units(positive only) and

$$2p + q = 10 \tag{0.1}$$

then find the values of p and q.

Solution

Points	**Name**
$\begin{pmatrix} 2 \\ 0 \end{pmatrix}$	Point **A**
$\begin{pmatrix} 6 \\ 1 \end{pmatrix}$	Point **B**
$\begin{pmatrix} p \\ q \end{pmatrix}$	Point **C**

Table: List of Points

→ Area of any △ABC can be given by

$$Area(\mathrm{ABC}) = \frac{1}{2}\left|\begin{pmatrix} A & B & C \\ 1 & 1 & 1 \end{pmatrix}\right|$$

Solution

→ The area of the given △ABC can be given by

$$Area(\mathrm{ABC}) = \frac{1}{2}\left|\begin{pmatrix} 2 & 6 & p \\ 0 & 1 & q \\ 1 & 1 & 1 \end{pmatrix}\right| \tag{0.2}$$

$$2 \times Area(\mathrm{ABC}) = 2 \times \left|\begin{pmatrix} 1 & q \\ 1 & 1 \end{pmatrix}\right| - 6 \times \left|\begin{pmatrix} 0 & q \\ 1 & 1 \end{pmatrix}\right| + p \times \left|\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}\right| \tag{0.3}$$

$$= 2(1-q) - 6(0-q) + p(0-1) \tag{0.4}$$

$$= 2 + 4q - p \tag{0.5}$$

$$Area(\mathrm{ABC}) = 12 \tag{0.6}$$

$$|4q - p + 2| = 24 \tag{0.7}$$

$$4q - p = \pm 24 - 2 \tag{0.8}$$

Solution

→ From (1) and (8), we get

$$\begin{pmatrix} 2 & 1 \\ -1 & 4 \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} 10 \\ \pm 24 - 2 \end{pmatrix} \tag{0.9}$$

$$\begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ -1 & 4 \end{pmatrix}^{-1} \begin{pmatrix} 10 \\ \pm 24 - 2 \end{pmatrix} \tag{0.10}$$

$$= \frac{1}{9} \begin{pmatrix} 4 & -1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} 10 \\ \pm 24 - 2 \end{pmatrix} \tag{0.11}$$

$$\begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} 2 \\ 6 \end{pmatrix} \text{ or } \begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} 22/3 \\ -14/3 \end{pmatrix} \tag{0.12}$$

Solution

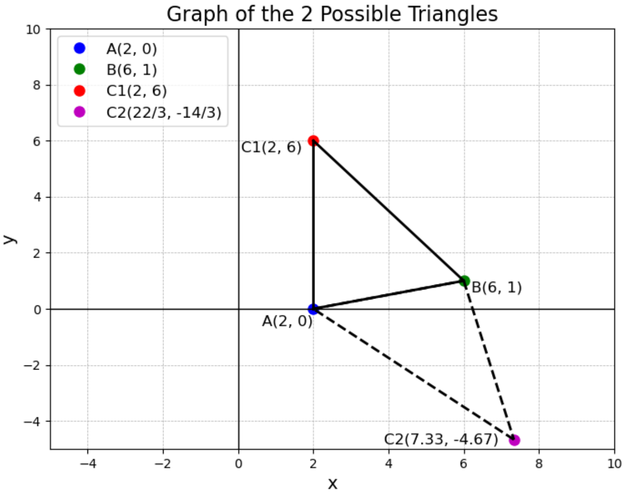

Figure: Plot of points and triangles

File: points.c

```c
#include <stdio.h>

int main() {
  FILE *fp;

  // -------------------
  // Question 2.9.15
  // -------------------


  fp = fopen("points.dat", "w");
  fprintf(fp, "%d,%d,%d\n", 2, 0, 0); // A
  fprintf(fp, "%d,%d,%d\n", 6, 1, 0); // B
  fprintf(fp, "%d,%d,%d\n", 2, 6, 0); // C1
  fprintf(fp, "%d,%d,%d\n", 22/3, -14/3, 0); // C2
  fclose(fp);
  return 0;
}
```

File: call_c.py

```
import subprocess

# Compile the C program
subprocess.run(["gcc", "points.c", "-o", "points"])

# Run the compiled C program
result = subprocess.run(["./points"], capture_output=True, text=True)

# Print the output from the C program
print(result.stdout)
```

File: plot.py

```
import matplotlib.pyplot as plt

# Points
A = (2, 0)
B = (6, 1)
C1 = (2, 6)
C2 = (22/3, -14/3)

# Create the plot
fig, ax = plt.subplots(figsize=(8, 6))

# Plot the points A, B, C1, and C2
ax.plot(A[0], A[1], ’bo’, label="A(2,␣0)", markersize=8)
ax.plot(B[0], B[1], ’go’, label="B(6,␣1)", markersize=8)
ax.plot(C1[0], C1[1], ’ro’, label="C1(2,␣6)", markersize=8)
ax.plot(C2[0], C2[1], ’mo’, label="C2(22/3,␣-14/3)", markersize=8)

# Connect the points A, B, and C1 to form the first triangle
ax.plot([A[0], B[0]], [A[1], B[1]], ’k-’, lw=2) # AB
ax.plot([B[0], C1[0]], [B[1], C1[1]], ’k-’, lw=2) # BC1
ax.plot([C1[0], A[0]], [C1[1], A[1]], ’k-’, lw=2) # C1A

# Connect the points A, B, and C2 to form the second triangle
ax.plot([A[0], B[0]], [A[1], B[1]], ’k--’, lw=2) # AB
ax.plot([B[0], C2[0]], [B[1], C2[1]], ’k--’, lw=2) # BC2
ax.plot([C2[0], A[0]], [C2[1], A[1]], ’k--’, lw=2) # C2A
```

File: plot.py

```
# Labels for the points
ax.text(A[0], A[1]-0.2, ’A(2,␣0)’, fontsize=12, ha=’right’, verticalalignment=’top’)
ax.text(B[0]+0.2, B[1], ’B(6,␣1)’, fontsize=12, ha=’left’, verticalalignment=’top’)
ax.text(C1[0]-0.3, C1[1], f’C1(2,␣6)’, fontsize=12, ha=’right’, verticalalignment=’top’)
ax.text(C2[0]-0.4, C2[1], f’C2({C2[0]:.2f},␣{C2[1]:.2f})’, fontsize=12, ha=’right’, verticalalignment=’
      center’)

# Set the axes limits
ax.set_xlim(-5, 10)
ax.set_ylim(-5, 10)

# Set labels and title
ax.set_xlabel(’x’, fontsize=14)
ax.set_ylabel(’y’, fontsize=14)
ax.set_title(’Graph␣of␣the␣2␣Possible␣Triangles’, fontsize=16)

# Show grid and customize
ax.grid(True, which=’both’, linestyle=’--’, linewidth=0.5)
ax.axhline(0, color=’black’,linewidth=1)
ax.axvline(0, color=’black’,linewidth=1)

# Show the legend
ax.legend(loc=’upper␣left’, fontsize=12)

# Show the plot
plt.show()
```