Matgeo Presentation - Problem 2.5.31

ee25btech11063 - Vejith

August 29, 2025

Question

If two vertices of an equilateral triangle are $(3, 0)$ and $(6, 0)$,find the third vertex

Description

Solution:

vector	**Name**
$\begin{pmatrix} 3 \\ 0 \end{pmatrix}$	Vector **A**
$\begin{pmatrix} 6 \\ 0 \end{pmatrix}$	Vector **B**
$\begin{pmatrix} x \\ y \end{pmatrix}$	Vector **C**

Table: Variables Used

Solution

$$\text{The vector joining from } \mathbf{A} \text{ to } \mathbf{B} \text{ is given by } \mathbf{B} - \mathbf{A} = \begin{pmatrix} 6 \\ 0 \end{pmatrix} - \begin{pmatrix} 3 \\ 0 \end{pmatrix} \tag{0.1}$$

$$\implies \mathbf{B} - \mathbf{A} = \begin{pmatrix} 3 \\ 0 \end{pmatrix}. \tag{0.2}$$

$$\tag{0.3}$$

An equilateral triangle can be obtained by rotating **B**-**A** by **A** about $+60^\circ$ or -60° .The rotation matrix p at angle θ is defined as

$$p(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \tag{0.4}$$

$$\tag{0.5}$$

Solution

$$p(60^\circ) = \begin{pmatrix} \frac{1}{2} & \frac{-\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & \frac{1}{2} \end{pmatrix} \qquad p(-60^\circ) = \begin{pmatrix} \frac{1}{2} & \frac{\sqrt{3}}{2} \\ \frac{-\sqrt{3}}{2} & \frac{1}{2} \end{pmatrix} \tag{0.6}$$

$$\tag{0.7}$$

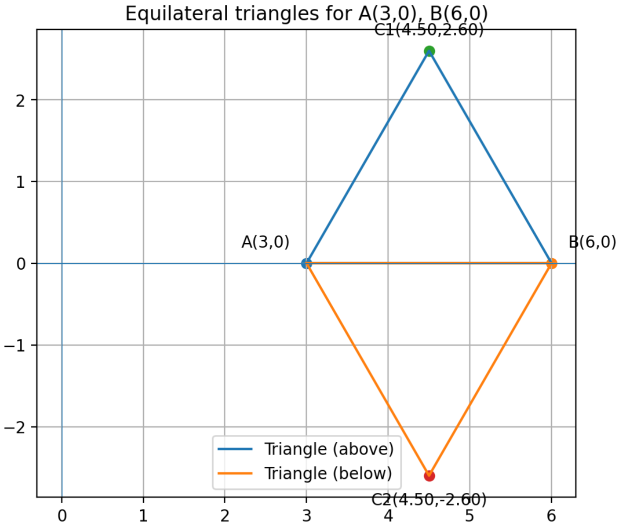
Apply $\mathrm{p}(60^\circ)$ or $\mathrm{p}(-60^\circ)$ to $\mathbf{B}-\mathbf{A}$ and add it to $\mathbf{A}$ to get $\mathbf{C}$

$$\mathbf{C} = \mathbf{A} + \mathrm{p}(60^\circ)(\mathbf{B}-\mathbf{A}) \qquad \text{or} \qquad \mathbf{C} = \mathbf{A} + \mathrm{p}(-60^\circ)(\mathbf{B}-\mathbf{A}) \tag{0.8}$$

$$\mathrm{p}(60^\circ)\begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{3}{2} \\ \frac{3\sqrt{3}}{2} \end{pmatrix} \qquad \text{or} \qquad \mathrm{p}(-60^\circ)\begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{3}{2} \\ \frac{-3\sqrt{3}}{2} \end{pmatrix} \tag{0.9}$$

$$\mathbf{C} = \begin{pmatrix} \frac{3}{2} \\ \frac{3\sqrt{3}}{2} \end{pmatrix} \qquad \text{or} \qquad \mathbf{C} = \begin{pmatrix} \frac{3}{2} \\ \frac{-3\sqrt{3}}{2} \end{pmatrix} \tag{0.10}$$

Plot


Equilateral triangles for A(3,0), B(6,0)
C1(4.50,2.60)
A(3,0)
B(6,0)
C2(4.50,-2.60)
Triangle (above)
Triangle (below)
2
1
0
−1
−2
0
1
2
3
4
5
6

C Code: code.c

```c
#include <stdio.h>
#include <math.h>

int main() {
    FILE *fp;
    fp = fopen("triangle.dat", "w");
    if (fp == NULL) {
        printf("Error opening file!\n");
        return 1;
    }

    // Given vertices
    double x1 = 3, y1 = 0;
    double x2 = 6, y2 = 0;

    // Midpoint of AB
    double xm = (x1 + x2) / 2.0;
    double ym = (y1 + y2) / 2.0;

    // Length of AB
    double side = sqrt((x2 - x1)*(x2 - x1) + (y2 - y1)*(y2 - y1));

    // Height of equilateral triangle
    double h = (sqrt(3) / 2.0) * side;

    // Two possible third vertices
    double x3a = xm;
    double y3a = ym + h;

    double x3b = xm;
    double y3b = ym - h;

    // Writing results to file
```

C Code: code.c

```
    fprintf(fp, "First␣vertex:␣(%.2f,␣%.2f)\n", x1, y1);
    fprintf(fp, "Second␣vertex:␣(%.2f,␣%.2f)\n", x2, y2);
    fprintf(fp, "Third␣vertex␣(above␣x-axis):␣(%.2f,␣%.2f)\n", x3a, y3a);
    fprintf(fp, "Third␣vertex␣(below␣x-axis):␣(%.2f,␣%.2f)\n", x3b, y3b);

    fclose(fp);

    printf("Results␣written␣to␣triangle.dat␣successfully.\n");
    return 0;
}
```

Python: plot.py

```
   import numpy as np
import matplotlib.pyplot as plt

# Given vertices
A = np.array([3.0, 0.0])
B = np.array([6.0, 0.0])

# Midpoint and side/height
M = (A + B) / 2
AB = B - A
side = np.linalg.norm(AB)
h = (np.sqrt(3) / 2) * side

# Unit vector perpendicular to AB
perp = np.array([-AB[1], AB[0]]) / np.linalg.norm(AB)

# Two possible third vertices
C1 = M + h * perp # above x-axis
C2 = M - h * perp # below x-axis

fig, ax = plt.subplots(figsize=(6, 6))

# Plot triangles (above and below)
ax.plot([A[0], B[0], C1[0], A[0]], [A[1], B[1], C1[1], A[1]], label="Triangle (above)")
ax.plot([A[0], B[0], C2[0], A[0]], [A[1], B[1], C2[1], A[1]], label="Triangle (below)")

# Mark points
for p in (A, B, C1, C2):
    ax.scatter(p[0], p[1])

# Label points (use annotate to offset labels)
ax.annotate("A(3,0)", xy=A, xytext=(-10, 8), textcoords="offset points", ha="right", va="bottom")
ax.annotate("B(6,0)", xy=B, xytext=(10, 8), textcoords="offset points", ha="left", va="bottom")
```

Python: plot.py

```
ax.annotate(f"C1({C1[0]:.2f},{C1[1]:.2f})", xy=C1, xytext=(0, 8), textcoords="offset points", ha="center",
      va="bottom")
ax.annotate(f"C2({C2[0]:.2f},{C2[1]:.2f})", xy=C2, xytext=(0, -10), textcoords="offset points", ha="center"
      , va="top")

# Axes/formatting
ax.axhline(0, linewidth=0.5)
ax.axvline(0, linewidth=0.5)
ax.set_aspect('equal', adjustable='box')
ax.grid(True)
ax.legend()
ax.set_title("Equilateral triangles for A(3,0), B(6,0)")

# Save and show
plt.savefig("triangle.png", dpi=200, bbox_inches="tight")
plt.show()
```