2.6.18

Dhanush Kumar A - AI25BTECH11010

September 9, 2025

Question

Find the area of the region bounded by the triangle whose vertices are $(-1, 0)$, $(1, 3)$ and $(3, 2)$.

Variables Used

Variable	**Formula**
A	$A = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$
B	$B = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$
C	$C = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$

Table: Variables Used

Solution

$$\text{Area of triangle ABC} = \frac{1}{2}\left|(\mathbf{A}-\mathbf{B})\times(\mathbf{A}-\mathbf{C})\right| \tag{1}$$

$$\mathbf{A}-\mathbf{B} = \begin{pmatrix}-1\\0\end{pmatrix} - \begin{pmatrix}1\\3\end{pmatrix} = \begin{pmatrix}-2\\-3\end{pmatrix} \tag{2}$$

$$\mathbf{A}-\mathbf{C} = \begin{pmatrix}-1\\0\end{pmatrix} - \begin{pmatrix}3\\2\end{pmatrix} = \begin{pmatrix}-4\\-2\end{pmatrix} \tag{3}$$

$$(\mathbf{A}-\mathbf{B})\times(\mathbf{A}-\mathbf{C}) = (-2)(-2)-(-3)(-4) = 4-12 = -8 \tag{4}$$

$$\text{Area} = \frac{1}{2}|-8| = 4 \tag{5}$$

Thus, the area of the triangle is 4.

Python code - Calculating the area of triangle

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

# Define the vertices of the triangle
A = np.array([-1, 0])
B = np.array([1, 3])
C = np.array([3, 2])

# Calculate area using cross product formula
area = 0.5 * np.abs(np.cross(B - A, C - A))
print(f"Area of the triangle: {area}")
```

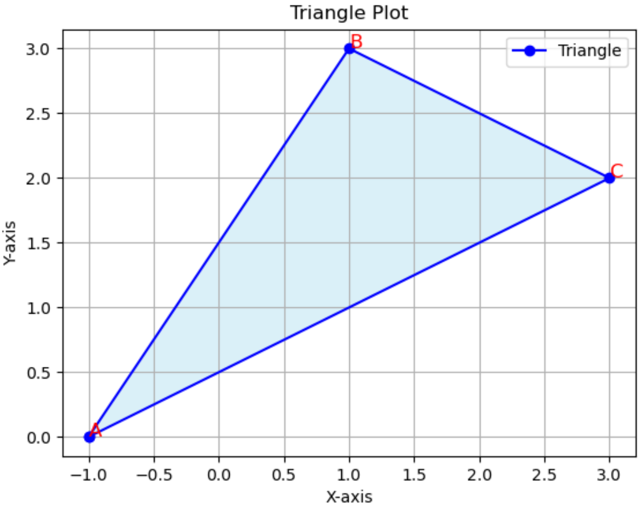
Python code - Plotting the triangle

```python
# Prepare triangle points for plotting
triangle = np.array([A, B, C, A]) # repeat first point to close the triangle

# Plot the triangle
plt.plot(triangle[:, 0], triangle[:, 1], 'b-o', label='Triangle')
plt.fill(triangle[:, 0], triangle[:, 1], 'skyblue', alpha=0.3)
plt.text(A[0], A[1], 'A', fontsize=12, color='red')
plt.text(B[0], B[1], 'B', fontsize=12, color='red')
plt.text(C[0], C[1], 'C', fontsize=12, color='red')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Triangle Plot')
plt.grid(True)
plt.legend()

# Save the figure
plt.savefig('../figs/triangle_plot.png')
plt.show()
```

Plot-Using Python



C code - To calculate the area of triangle and Save points

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

// Structure to store a 2D point
typedef struct {
    double x;
    double y;
} Point;

// Function to calculate the area of a triangle using 2D determinant formula
double triangle_area(Point A, Point B, Point C) {
    return 0.5 * fabs(A.x*(B.y - C.y) + B.x*(C.y - A.y) + C.x*(A.y - B.y));
}
```

C code - To calculate the area of triangle and Save points

```c

// Function to save points and area to a file
void save_points_and_area(const char *filename, Point A, Point B,
     Point C, double area) {
    FILE *fp = fopen(filename, "w");
    if (fp == NULL) {
        printf("Error opening file!\n");
        exit(1);
    }
    fprintf(fp, "Triangle Vertices:\n");
    fprintf(fp, "A: %.2lf %.2lf\n", A.x, A.y);
    fprintf(fp, "B: %.2lf %.2lf\n", B.x, B.y);
    fprintf(fp, "C: %.2lf %.2lf\n", C.x, C.y);
    fprintf(fp, "Area of the triangle: %.2lf\n", area);
    fclose(fp);
}
```

C code - To calculate the area of triangle and Save points

```c

int main() {
    // Triangle vertices
    Point A = {-1, 0};
    Point B = {1, 3};
    Point C = {3, 2};

    // Calculate area
    double area = triangle_area(A, B, C);

    // Print points and area
    printf("Triangle Vertices:\n");
    printf("A: (%.2lf, %.2lf)\n", A.x, A.y);
    printf("B: (%.2lf, %.2lf)\n", B.x, B.y);
    printf("C: (%.2lf, %.2lf)\n", C.x, C.y);
    printf("Area of the triangle: %.2lf\n", area);
```

C code - To calculate the area of triangle and Save points

```c

    // Save points and area to file
    save_points_and_area("points.dat", A, B, C, area);
    printf("Triangle points and area saved in points.dat\n");

    return 0;
    }
```

Python code -Ploting the points using c function

```python
import os
import matplotlib.pyplot as plt

# Run the C program
# On Windows use: os.system("triangle.exe")
os.system("./triangle") # Linux/Mac

# Read points and area from points.dat
points = {}
area=0
```

Python code -Ploting the points using c function

```python


with open("points.dat", "r") as file:
    for line in file:
        line = line.strip()
        if line.startswith("A:"):
            x, y = map(float, line[2:].split())
            points['A'] = (x, y)
        elif line.startswith("B:"):
            x, y = map(float, line[2:].split())
            points['B'] = (x, y)
        elif line.startswith("C:"):
            x, y = map(float, line[2:].split())
            points['C'] = (x, y)
        elif line.startswith("Area"):
     print(f"Read Area from C program: {area}")
print("Triangle Points:", points)
```

Python code -Ploting the points using c function

```python

# Prepare triangle points for plotting
triangle_coords = [points['A'], points['B'], points['C'], points['A']] # close the triangle
x_vals, y_vals = zip(*triangle_coords)

# Create figs folder if it doesn't exist
os.makedirs('figs', exist_ok=True)

# Plot
plt.plot(x_vals, y_vals, 'b-o', label='Triangle')
plt.fill(x_vals, y_vals, 'skyblue', alpha=0.3)

# Label points
for label, coord in points.items():
    plt.text(coord[0], coord[1], label, fontsize=12, color='red')
```

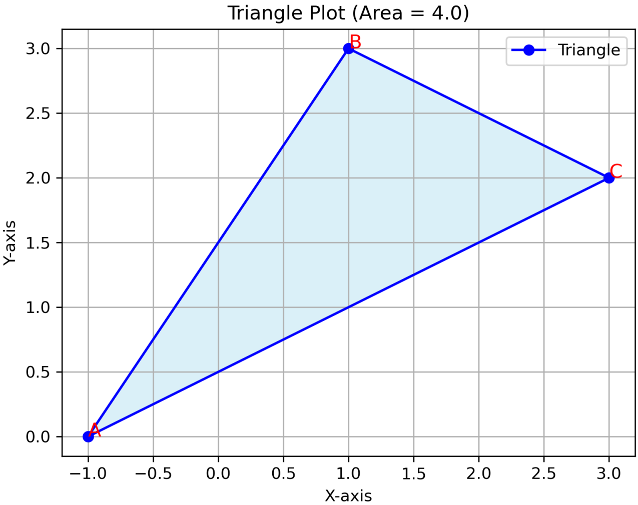
Python code -Ploting the points using c function

```

plt.title(f'Triangle Plot (Area = {area})')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.grid(True)
plt.legend()

# Save the plot in figs folder
plt.savefig('../figs/triangle_plot1.png', dpi=300)
print("Triangle plot saved in figs/triangle_plot.png")
plt.show()
```

Plot-Using Python and C


Triangle Plot (Area = 4.0)
Triangle
A
B
C
Y-axis
X-axis