2.4.29

Nipun Dasari - EE25BTECH11042

September 16, 2025

Question

The points $\mathbf{A}\,(2,9)$, $\mathbf{B}\,(a,5)$ and $\mathbf{C}\,(5,5)$ are the vertices of a triangle **ABC** right angled at **B**. Find the values of a and hence the area of $\triangle\mathbf{ABC}$.

Theoretical Solution

Given the points A, B and C, also consider **c** to be vector opposite to side AB and **b**, **a** similarly

$$\mathbf{A} = \begin{pmatrix} 2 \\ 9 \end{pmatrix}, \mathbf{B} = \begin{pmatrix} a \\ 5 \end{pmatrix}, \mathbf{C} = \begin{pmatrix} 5 \\ 5 \end{pmatrix} \tag{1}$$

Since the sides c and a are perpendicular their inner product will be 0

Take the inner product of **c** and **a**

Vector **c**:

$$\mathbf{c} = \mathbf{A} - \mathbf{B} = \begin{pmatrix} 2-a \\ 9-5 \end{pmatrix} = \begin{pmatrix} 2-a \\ 4 \end{pmatrix} \tag{2}$$

Vector **a**:

$$\mathbf{a} = \mathbf{B} - \mathbf{C} = \begin{pmatrix} a-5 \\ 5-5 \end{pmatrix} = \begin{pmatrix} a-5 \\ 0 \end{pmatrix} \tag{3}$$

Orthogonality $\implies$ matrix product is zero :

C Code- Triangle Area function

```c
// triangle.c
#include <math.h>

float triangle_area(float ax, float ay, float bx, float by, float cx, float cy) {
      float area = fabs(ax*(by-cy) + bx*(cy-ay) + cx*(ay-by)) / 2.0;
      return area;
}
```

Python Code using shared output

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

# Load the shared C library
lib = ctypes.CDLL(./trianglearea.so)
lib.triangle_area.argtypes = [ctypes.c_float, ctypes.c_float,
ctypes.c_float, ctypes.c_float,
ctypes.c_float, ctypes.c_float]
lib.triangle_area.restype = ctypes.c_float

# Vertices
A = (2, 9)
B = (2, 5)
C = (5, 5)
```

Python Code using shared output

```python
# Call C function
area = lib.triangle_area(A[0], A[1], B[0], B[1], C[0], C[1])
print(Area of triangle ABC (from C):, area)

# Plot triangle
x = [A[0], B[0], C[0], A[0]]
y = [A[1], B[1], C[1], A[1]]

plt.plot(x, y, 'bo-')
plt.text(A[0], A[1], A(2,9), fontsize=10, ha=right)
plt.text(B[0], B[1], B(2,5), fontsize=10, ha=right)
plt.text(C[0], C[1], C(5,5), fontsize=10, ha=right)
plt.title(Right Triangle ABC)
plt.grid(True)
plt.axis(equal)
plt.show()
```

Python Code using shared output

```
# Plotting
plt.figure(figsize=(6, 6))
plt.scatter(A[0], A[1], color='red', label='A(5,1)
    ')
plt.scatter(B[0], B[1], color='blue', label='B
    (-1,5)')
plt.scatter(midpoint[0], midpoint[1], color='black
    ', label='Midpoint')

# Perpendicular bisector line
plt.plot(x_vals, y_vals, 'g--', label='3x = 2y (
    Perp. bisector)')

# Mark example points
for p in points_to_check:
plt.scatter(p[0], p[1], label=fPoint {p})
```

Python Code

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

# Vertices
A = np.array([2, 9])
B = np.array([2, 5])
C = np.array([5, 5])

# Check right angle at B using dot product
AB = A - B
BC = C - B
print(Dot product ABBC =, np.dot(AB, BC))
```

Python Code

```
if np.dot(AB, BC) == 0:
print(Right angle at B )


area = abs(np.linalg.det(np.array([
[A[0], A[1], 1],
[B[0], B[1], 1],
[C[0], C[1], 1]
])) / 2
print(Area of triangle ABC (Python):, area)

# Plot
x = [A[0], B[0], C[0], A[0]]
y = [A[1], B[1], C[1], A[1]]
```

Python Code

```
plt.plot(x, y, 'ro-')
plt.text(A[0], A[1], A(2,9), fontsize=10, ha=right)
plt.text(B[0], B[1], B(2,5), fontsize=10, ha=right)
plt.text(C[0], C[1], C(5,5), fontsize=10, ha=right)
plt.title(Right Triangle ABC (Pure Python))
plt.grid(True)
plt.axis(equal)
plt.show()
```

Plot by python using shared output from c

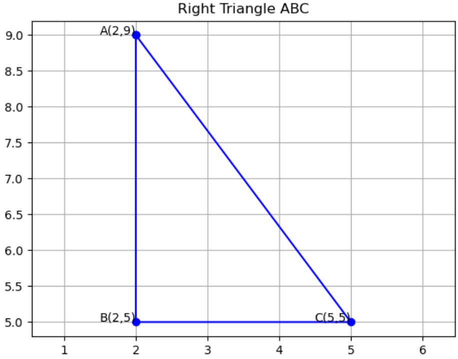

Figure: *

Plot by python only

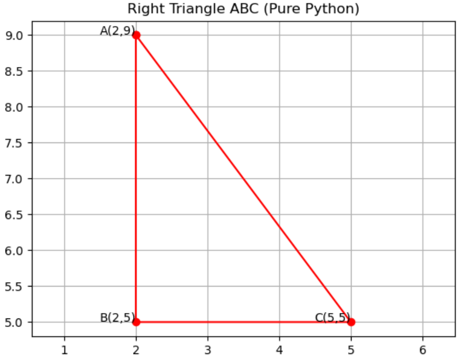

Figure: *