1.5.3

Bhoomika V - EE25BTECH11015

August 29,2025

Question

In what ratio does the X axis divide line segment joining the points **A**(3,6) and **B**(-12,-3)?

Theoretical Solution

Let **A**(3,6) , **B**(-12,-3) and the point on X axis be **X**(t,0)
Using the collinearity (*rank*) test, form the matrix with difference vectors:

$$(\mathbf{B}-\mathbf{A} \quad \mathbf{X}-\mathbf{A}) = \begin{pmatrix} -12-3 & t-3 \\ -3-6 & 0-6 \end{pmatrix}$$
$$= \begin{pmatrix} -15 & t-2 \\ -9 & -6 \end{pmatrix}.$$

Table

Table: Vectors

Equation

The three points are collinear $\iff$ this matrix has rank 1
(*its rows are linearly dependent*)**.**

Theoretical Solution

Using Gauss–Jordan elimination,

$$R_2 \leftarrow 5R_2 - 3R_1 \implies \begin{pmatrix} -45 & 3t-9 \\ 0 & -3t-21 \end{pmatrix}.$$

For rank 1, the second row must be zero:

$$-3t - 21 = 0 \implies t = -7$$

Section Formulae

let **X** divide **A** and **B** in the ratio k:1 then

$$k = \frac{(A - X)^T (X - B)}{\|X - B\|^2} \tag{1}$$

solving equation(1) we get k = 2

C Code - A function to find the value of t

```c
#include <stdio.h>

// Function to compute ratio in which X-axis divides AB
// Returns ratio m:n as a floating point (m/n)
float find_ratio(float Ax, float Ay, float Bx, float By) {
    // Equation of line AB: y = slope * x + c
    float slope = (By - Ay) / (Bx - Ax);
    float intercept = Ay - slope * Ax;

    // Intersection with X-axis (y = 0)
    float x_intersect = -intercept / slope;
    float y_intersect = 0;
```

C Code - A function to find the value of t

```
    // Compute distances AX and XB (only x-difference since y=0
        for X)
    float AX = (Ax - x_intersect);
    if (AX < 0) AX = -AX;

    float XB = (Bx - x_intersect);
    if (XB < 0) XB = -XB;

    // Print ratio
    printf("The X-axis divides AB in the ratio %.0f:%.0f\n", AX,
        XB);

    // Return the floating ratio (AX/XB)
    return AX / XB;
}
```

Python Code

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

# Points
A = np.array([3.0, 6.0])
B = np.array([-12.0, -3.0])
X = np.array([-7.0, 0.0]) # Intersection with X-axis

# Calculate ratio AX : XB
AX = np.linalg.norm(A - X)
XB = np.linalg.norm(B - X)
ratio = AX / XB
print(f"The X-axis divides AB in the ratio {int(AX)}:{int(XB)} (i
    .e., {ratio:.2f}:1)")
```

Python Code

```
# Equation of line AB
slope = (B[1] - A[1]) / (B[0] - A[0])
intercept = A[1] - slope * A[0]

x_min = min(A[0], B[0], X[0]) - 2
x_max = max(A[0], B[0], X[0]) + 2
x_line = np.linspace(x_min, x_max, 100)
y_line = slope * x_line + intercept

# Plot line AB
plt.plot(x_line, y_line, label='Line AB', color='black',
    linestyle='--')

# Plot points
all_points = np.vstack((A, B, X)).T
plt.scatter(all_points[0, :], all_points[1, :], color=['red', '
    blue', 'green'], zorder=5)
```

Python Code

```
# Labels
point_labels = [f'A({A[0]},{A[1]})', f'B({B[0]},{B[1]})', f'X({X
    [0]},{X[1]})']
for i, txt in enumerate(point_labels):
    plt.annotate(txt,
                 (all_points[0, i], all_points[1, i]),
                 textcoords="offset points",
                 xytext=(10, 5),
                 ha='center')
```

Python Code

```
slope = (B[1] - A[1]) / (B[0] - A[0])
intercept = A[1] - slope * A[0]

x_min = min(A[0], B[0], C[0]) - 2
x_max = max(A[0], B[0], C[0]) + 2
x_line = np.linspace(x_min, x_max, 100)
y_line = slope * x_line + intercept

plt.plot(x_line, y_line, label=f'Line through A, B, and C (m={
    m_value:.2f})', color='blue')

all_points = np.vstack((A, B, C)).T
plt.scatter(all_points[0, :], all_points[1, :], color='red',
    zorder=5)
```

Python Code

```
# Axes and formatting
plt.axhline(0, color='gray', linewidth=1)
plt.axvline(0, color='gray', linewidth=1)
plt.xlabel('$x$')
plt.ylabel('$y$')
plt.title('Division of Line Segment AB by X-axis')
plt.legend(loc='best')
plt.grid(True)
plt.axis('equal')

# Save figure
plt.savefig('division_AB.png', dpi=300)

# Show plot
plt.show()
```

Plot