1.7.3

BALU-ai25btech11017

August 29, 2025

Question

Show that the points $A(-2\hat{i} + 3\hat{j} + 5\hat{k})$, $B(\hat{i} + 2\hat{j} + 3\hat{k})$, and $C(7\hat{i} - \hat{k})$ are collinear.

Theoretical Solution

Given positional vectors,

$$\mathbf{A} = \begin{pmatrix} -2 \\ 3 \\ 5 \end{pmatrix} \quad \mathbf{B} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \quad \mathbf{C} = \begin{pmatrix} 7 \\ 0 \\ -1 \end{pmatrix} \tag{1}$$

To show that these are points are collinear,we show that echolon matrix $\mathbf{S}$ Rank=1

$$\mathbf{S} = \begin{pmatrix} \mathbf{B} - \mathbf{A} & \mathbf{C} - \mathbf{A} \end{pmatrix}^T \tag{2}$$

$$\mathbf{S} = \begin{pmatrix} 3 & -1 & -2 \\ 9 & -3 & -6 \end{pmatrix} \tag{3}$$

Theoretical Solution

$$BydoingR_2 = R_2 - 3R_1 weget \tag{4}$$

$$\mathbf{S} = \begin{pmatrix} 3 & -1 & -2 \\ 0 & 0 & 0 \end{pmatrix} \tag{5}$$

So the Rank of matrix $\mathbf{S}$ is 1

$\therefore$ The points are collinear.

C Code - Resultant velocity

```c
#include <stdio.h>

int main() {
    // Points A, B, C
    int Ax = -2, Ay = 3, Az = 5;
    int Bx = 1, By = 2, Bz = 3;
    int Cx = 7, Cy = 0, Cz = -1;

    // Vectors AB = B - A, AC = C - A
    int ABx = Bx - Ax;
    int ABy = By - Ay;
    int ABz = Bz - Az;

    int ACx = Cx - Ax;
    int ACy = Cy - Ay;
    int ACz = Cz - Az;

    printf("Vector AB = (%d, %d, %d)\n", ABx, ABy, ABz);
```

C Code - Resultant velocity

```
    printf("Vector AC = (%d, %d, %d)\n", ACx, ACy, ACz);
    // Check if AC is a scalar multiple of AB
    // (Cross product must be zero for collinearity)
    int cross_x = ABy * ACz - ABz * ACy;
    int cross_y = ABz * ACx - ABx * ACz;
    int cross_z = ABx * ACy - ABy * ACx;

    printf("Cross product AB x AC = (%d, %d, %d)\n", cross_x,
        cross_y, cross_z);

    if (cross_x == 0 && cross_y == 0 && cross_z == 0) {
        printf(" Points A, B, and C are collinear.\n");
    } else {
        printf(" Points A, B, and C are NOT collinear.\n");
    }

    return 0;
}
```

Python Code

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# Points
A = np.array([-2, 3, 5])
B = np.array([1, 2, 3])
C = np.array([7, 0, -1])

# Direction vector AB
AB = B - A

# Generate line through A in direction AB
t = np.linspace(-1, 3, 100) # parameter
line = A[:, None] + AB[:, None] * t
```

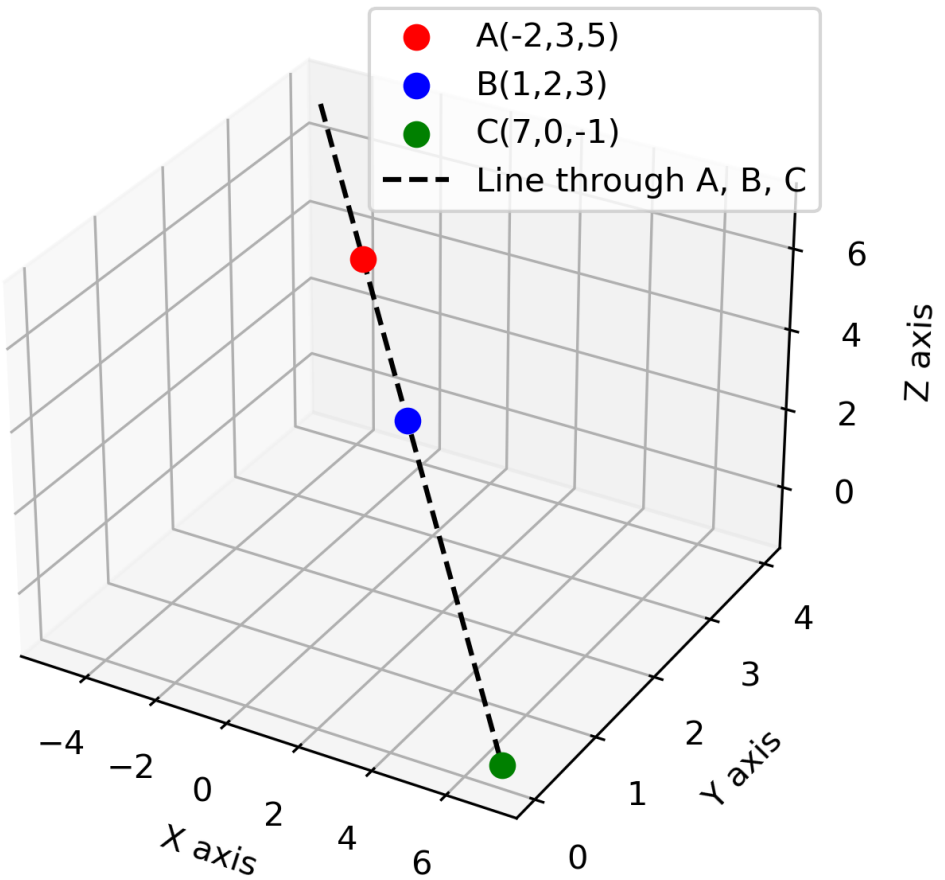
Python Code

```
# Plot
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

# Plot points
ax.scatter(*A, color='red', label='A(-2,3,5)', s=50)
ax.scatter(*B, color='blue', label='B(1,2,3)', s=50)
ax.scatter(*C, color='green', label='C(7,0,-1)', s=50)

# Plot line through A, B, C
ax.plot(line[0], line[1], line[2], color='black', linestyle='--',
    label='Line through A, B, C')
```

Python Code

```
# Labels
ax.set_xlabel('X axis')
ax.set_ylabel('Y axis')
ax.set_zlabel('Z axis')
ax.legend()

# Save figure
plt.savefig("/home/balu/matgeo/figs/fig.png", dpi=300)
plt.show()
```

Plot


A(-2,3,5)
B(1,2,3)
C(7,0,-1)
Line through A, B, C
X axis
Y axis
Z axis
−4
−2
0
2
4
6
0
1
2
3
4
0
2
4
6