1.10.25

Vaishnavi - EE25BTECH11059

August 31, 2025

Question

If a line makes angles 90°, 60°, and 30° with the positive directions of the X, Y, and Z axes respectively, find its direction cosines.

Definition

Definition of Direction Cosines: Direction cosines are the cosine values of the angles a vector makes with the x, y, and z axes; they are the components of the unit vector along x,y,z axes

Solution

Angle (α)	$\cos(\alpha)$	Value	Axis
90°	$\cos(90^\circ) = 0$	$l = 0$	x-axis
60°	$\cos(60^\circ) = \frac{1}{2}$	$m = \frac{1}{2}$	y-axis
30°	$\cos(30^\circ) = \frac{\sqrt{3}}{2}$	$n = \frac{\sqrt{3}}{2}$	z-axis

Table: Variables Used

Solution

Let the direction cosines be l,m,n
l,m,n, which are the cosines of the angles that the line makes with the X, Y, and Z axes respectively.

$$l = \cos(90^\circ) = 0 \tag{1}$$

$$m = \cos(60^\circ) = \frac{1}{2} \tag{2}$$

$$n = \cos(30^\circ) = \frac{\sqrt{3}}{2} \tag{3}$$

A key property is that the sum of the squares of the direction cosines always equals one: $l^2 + m^2 + n^2 = 1$

solution

$$\text{unit vector in direction of} \mathbf{x} = \begin{pmatrix} 0 \\ \frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{pmatrix} \tag{4}$$

Graph

Refer to Figure

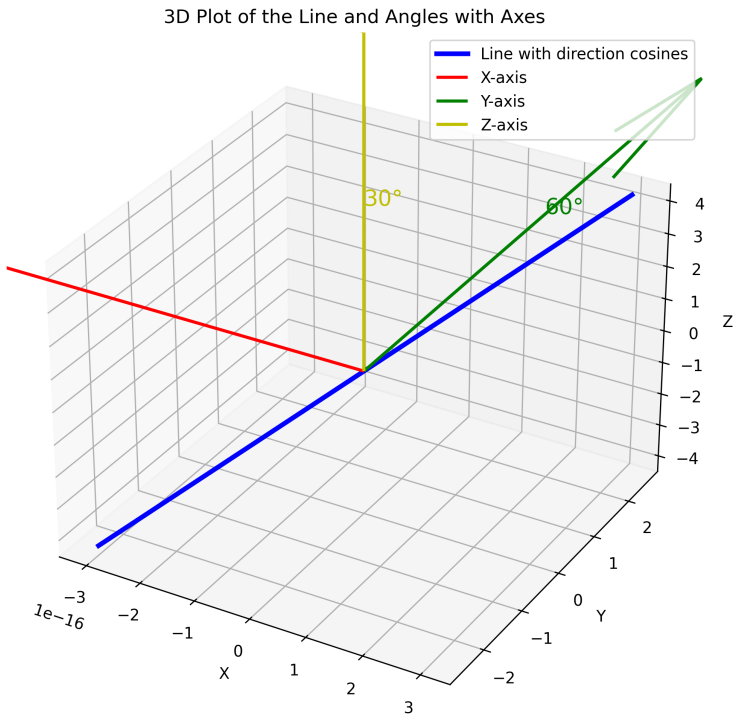

Figure:

Python Code

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

# Create a figure and 3D axis
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

# Define the line with direction cosines
x = np.linspace(-3, 3, 100)
y = np.linspace(-3, 3, 100)
z = np.linspace(-3, 3, 100)

# Direction cosines for the line (example: cosines of
    30   , 60   )
cos_30 = np.cos(np.radians(30))
cos_60 = np.cos(np.radians(60))
```

Python Code

```
# Parametric equations for the line with direction
    cosines
x_line = x * cos_30
y_line = y * cos_60
z_line = z
# Plot the line
ax.plot(x_line, y_line, z_line, label='Line with
    direction cosines', color='b')

# Plot the axes (X, Y, Z)
ax.plot([0, 3], [0, 0], [0, 0], color='r', label='X-
    axis')
ax.plot([0, 0], [0, 3], [0, 0], color='g', label='Y-
    axis')
ax.plot([0, 0], [0, 0], [0, 3], color='y', label='Z-
    axis')
```

Python Code

```
 # Set labels
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')

# Set the title
ax.set_title('3D Plot of the Line and Angles with Axes
    ')

# Display the legend
ax.legend()
# Save the figure
fig.savefig( collinear_3d_plot.png )

# Show the plot
plt.show()
```

C Code

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

#define DEG_TO_RAD(deg) ((deg) * (M_PI / 180.0))

int main() {
    // Define the angles in degrees
    double alpha = 90.0;  // Angle with X-axis
    double beta = 60.0;   // Angle with Y-axis
    double gamma = 30.0;  // Angle with Z-axis

    // Convert degrees to radians
    double alpha_rad = DEG_TO_RAD(alpha);
    double beta_rad = DEG_TO_RAD(beta);
    double gamma_rad = DEG_TO_RAD(gamma);
```

C Code

```
    // Calculate the direction cosines
    double l = cos(alpha_rad);  // cos(90 degrees)
    double m = cos(beta_rad);   // cos(60 degrees)
    double n = cos(gamma_rad);  // cos(30 degrees)

    // Print the direction cosines
    printf( Direction Cosines of the vector:\n );
    printf( l = cos(90 degrees) = %.2f\n , l);
    printf( m = cos(60 degrees) = %.2f\n , m);
    printf( n = cos(30 degrees) = %.2f\n , n);

    // Display the vector (l, m, n)
    printf( Direction cosines of vector x = (%.2f, %.2
        f, %.2f)\n , l, m, n);

    return 0;
}
```

Python and C Code

```
import subprocess

# Compile the C program
subprocess.run([ gcc ,  points.c ,  -o ,  points ])

# Run the compiled C program
result = subprocess.run([ ./points ], capture_output=
    True, text=True)

# Print the output from the C program (direction
    cosines)
print(result.stdout)
```