2.7.27

Bhoomika V - EE25BTECH11015

September 14,2025

Question

Find the equation of the line which passes through the point $(-2, 4, -5)$ and is parallel to the line

$$\frac{x+3}{3} = \frac{y-4}{5} = \frac{z+8}{6}.$$

The vertices of a triangle

Let the equation of line passing through the given point be

$$\mathbf{x} = \begin{pmatrix} -2 \\ 4 \\ -5 \end{pmatrix} + \mu\mathbf{d}$$

where $\mathbf{d}$ is the direction vector of the line.

The direction vector of the line

$$\mathbf{x} = \begin{pmatrix} -3 \\ 4 \\ -8 \end{pmatrix} + \lambda \begin{pmatrix} 3 \\ 5 \\ 6 \end{pmatrix}$$

is

$$\mathbf{d} = \begin{pmatrix} 3 \\ 5 \\ 6 \end{pmatrix}. \tag{1}$$

Thus, the required equation of the line is

$$\mathbf{x} = \begin{pmatrix} -3 \\ 4 \\ -8 \end{pmatrix} + \mu \begin{pmatrix} 3 \\ 5 \\ 6 \end{pmatrix}.$$

C Code - A function to find if triangle is right angled

```c
#include <stdio.h>

// Function to compute a point on the line
// Params: (px, py, pz) = point on line
// (dx, dy, dz) = direction vector
// t = parameter
// Output: coords[3] = coordinates of point
void line_point(float px, float py, float pz,
                float dx, float dy, float dz,
                float t, float coords[3]) {
    coords[0] = px + dx * t; // x = x0 + t*dx
    coords[1] = py + dy * t; // y = y0 + t*dy
    coords[2] = pz + dz * t; // z = z0 + t*dz
}
```

Python Code

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

# --- Load the compiled C library ---
c_lib = ctypes.CDLL('./line.so')

# Define argument & return types
c_lib.line_point.argtypes = [ctypes.c_float, ctypes.c_float,
    ctypes.c_float,
                             ctypes.c_float, ctypes.c_float, ctypes.
                                c_float,
                             ctypes.c_float,
                             ctypes.POINTER(ctypes.c_float)]

# --- Given point & direction vector ---
P = (-2.0, 4.0, -5.0) # point
d = (3.0, 5.0, 6.0) # direction vector
```

Python Code

```python
# --- Generate points on the line using C function ---
t_values = np.linspace(-5, 5, 50)
line_points = []

for t in t_values:
    coords = (ctypes.c_float * 3)() # array of 3 floats
    c_lib.line_point(P[0], P[1], P[2],
                     d[0], d[1], d[2],
                     ctypes.c_float(t), coords)
    line_points.append([coords[0], coords[1], coords[2]])

line_points = np.array(line_points)

# --- Plot the line ---
fig = plt.figure(figsize=(8,6))
ax = fig.add_subplot(111, projection='3d')
```

Python Code

```python
ax.plot(line_points[:,0], line_points[:,1], line_points[:,2],
        color="red", label="Required Line")
ax.scatter(P[0], P[1], P[2], color="green", s=50)
ax.text(P[0], P[1], P[2], "P(-2,4,-5)", color="green")

# Labels
ax.set_xlabel("X-axis")
ax.set_ylabel("Y-axis")
ax.set_zlabel("Z-axis")
ax.set_title("Line through P(-2,4,-5) with direction (3,5,6)")
ax.legend()
plt.show()
```

Plot

Line through P(-2,4,-5) parallel to given line