4.4.13

BALU-ai25btech11017

September 9, 2025

Question

A line passes through the point with position vector

$$2\hat{i} - \hat{j} + 4\hat{k} \tag{1}$$

and is in the direction of the vector

$$\hat{i} + \hat{j} - 2\hat{k}. \tag{2}$$

Find the equation of the line.

Theoretical Solution

Let us solve the given equation theoretically and then verify the solution computationally
According to the question,
Given

$$\mathbf{P} = \begin{pmatrix} 2 \\ -1 \\ 4 \end{pmatrix} \ \mathbf{D} = \begin{pmatrix} 1 \\ 1 \\ -2 \end{pmatrix} \tag{3}$$

The equation of line is

$$\mathbf{r} = \begin{pmatrix} 2 \\ -1 \\ 4 \end{pmatrix} + \lambda \begin{pmatrix} 1 \\ 1 \\ -2 \end{pmatrix} \tag{4}$$

C Code

```c
#include <stdio.h>

int main() {
    // Point on line
    int x0 = 2, y0 = -1, z0 = 4;
    // Direction vector
    int a = 1, b = 1, c = -2;

    printf("Equation of the line passing through (2, -1, 4)\n");
    printf("and parallel to vector (1, 1, -2):\n\n");

    // Vector form
    printf("Vector form:\n");
    printf("r = (2, -1, 4) + t(1, 1, -2)\n\n");

    // Parametric form
    printf("Parametric form:\n");
    printf("x = %d + t\n", x0);
```

C Code

```c
     printf("y = %d + t\n", y0);
    printf("z = %d - 2t\n\n");

    // Symmetric form
    printf("Symmetric form:\n");
    printf("(x - %d)/%d = (y - %d)/%d = (z - %d)/%d\n",
           x0, a, y0, b, z0, c);

    return 0;
}
```

Python Code

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

# Point and direction vector
P = np.array([2, -1, 4])
d = np.array([1, 1, -2])

# Parameter range for t
t = np.linspace(-5, 5, 100)

# Line points
x = P[0] + d[0]*t
y = P[1] + d[1]*t
z = P[2] + d[2]*t

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

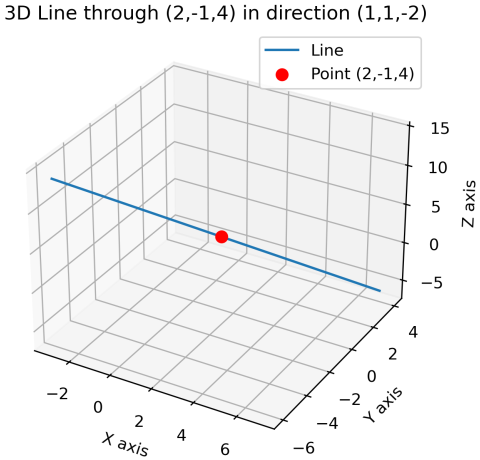
Python Code

```python
ax.plot(x, y, z, label="Line")
ax.scatter(P[0], P[1], P[2], color='red', s=50, label="Point (2,-1,4)")

# Labels
ax.set_xlabel('X axis')
ax.set_ylabel('Y axis')
ax.set_zlabel('Z axis')
ax.set_title('3D Line through (2,-1,4) in direction (1,1,-2)')
ax.legend()

# Save as picture
plt.savefig("line_3d.png", dpi=300)
plt.show()
```

Plot



3D Line through (2,-1,4) in direction (1,1,-2)
Line
● Point (2,-1,4)
X axis
−2
0
2
4
6
Y axis
−6
−4
−2
0
2
4
Z axis
−5
0
5
10
15