2.4.22

INDHIRESH S - EE25BTECH11027

August, 2025

Question

Find the equation of a plane which bisects perpendicularly the line joining the points $A(2, 3, 4)$ and $B(4, 5, 8)$ at right angles.

Equation

Let,

$$\mathbf{A} = \begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix} \quad \textit{and} \quad \mathbf{B} = \begin{pmatrix} 4 \\ 5 \\ 8 \end{pmatrix} \tag{1}$$

Theoretical Solution

Given that the plane is a perpendicular bisector to the line joining points A and B. Since it is a perpendicular bisector to the line joining points A and B , the midpoint of the line joining points A and B lies on the plane.
Let the midpoint of points A and B be C. Then

$$norm(\mathbf{C}-\mathbf{A}) = norm(\mathbf{C}-\mathbf{B}) \tag{2}$$

$$\sqrt{(\mathbf{C}-\mathbf{A})^T(\mathbf{C}-\mathbf{A})} = \sqrt{(\mathbf{C}-\mathbf{B})^T(\mathbf{C}-\mathbf{B})} \tag{3}$$

$$(\mathbf{C}-\mathbf{A})^T(\mathbf{C}-\mathbf{A}) = (\mathbf{C}-\mathbf{B})^T(\mathbf{C}-\mathbf{B}) \tag{4}$$

Theoretical Solution

Let,

$$C = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{5}$$

$$\left(\begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix}\right)^T \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix}\right) = \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} 4 \\ 5 \\ 8 \end{pmatrix}^T\right) \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} 4 \\ 5 \\ 8 \end{pmatrix}\right) \tag{6}$$

$$\begin{pmatrix} x-2 \\ y-3 \\ z-4 \end{pmatrix}^T \begin{pmatrix} x-2 \\ y-3 \\ z-4 \end{pmatrix} = \begin{pmatrix} x-4 \\ y-5 \\ z-8 \end{pmatrix}^T \begin{pmatrix} x-4 \\ y-5 \\ z-8 \end{pmatrix} \tag{7}$$

Theoretical Solution

$$(x-2)^2+(y-3)^2+(z-4)^2=(x-4)^2+(y-5)^2+(z-8)^2 \tag{8}$$

$$x^2+4-4x+y^2+9-6y+z^2+16-8z=x^2+16-8x+y^2+25-10y \tag{9}$$

$$4x+4y+8z=76 \tag{10}$$

$$x+y+2z=19 \tag{11}$$

Theoretical Solution

Now the equation of plane is :

$$x + y + 2z = 19 \tag{12}$$

In matrix form:

$$\begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}^T \mathbf{R} = 19 \tag{13}$$

Where $\mathbf{R}$ is the equation of the plane

C Code - Midpoint formula

```c
#include<stdio.h>
void midpoint(float* out, float* A, float* B) {
    out[0] = (A[0] + B[0]) / 2.0f; // X-coordinate
    out[1] = (A[1] + B[1]) / 2.0f; // Y-coordinate
    out[2] = (A[2] + B[2]) / 2.0f; // Z-coordinate
}
```

Python Code

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

# Load C library
libmid = ctypes.CDLL('./plane.so')

# Define arrays (float32)
A = np.array([2.0, 3.0, 4.0], dtype=np.float32)
B = np.array([4.0, 5.0, 8.0], dtype=np.float32)
M = np.zeros(3, dtype=np.float32)

# Set argtypes/restype for C function
libmid.midpoint.argtypes = [ctypes.POINTER(ctypes.c_float),
                            ctypes.POINTER(ctypes.c_float),
                            ctypes.POINTER(ctypes.c_float)]
libmid.midpoint.restype = None
```

Python Code

```
# Call C function to compute midpoint
libmid.midpoint(M.ctypes.data_as(ctypes.POINTER(ctypes.c_float)),
                A.ctypes.data_as(ctypes.POINTER(ctypes.c_float)),
                B.ctypes.data_as(ctypes.POINTER(ctypes.c_float)))

print(Midpoint:, M)

# Prepare plane x + y + z = 10
xx, yy = np.meshgrid(np.linspace(0, 6, 20), np.linspace(0, 8, 20)
    )
zz = 10 - xx - yy

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

# Plane
ax.plot_surface(xx, yy, zz, alpha=0.3, color='cyan')
```

Python Code

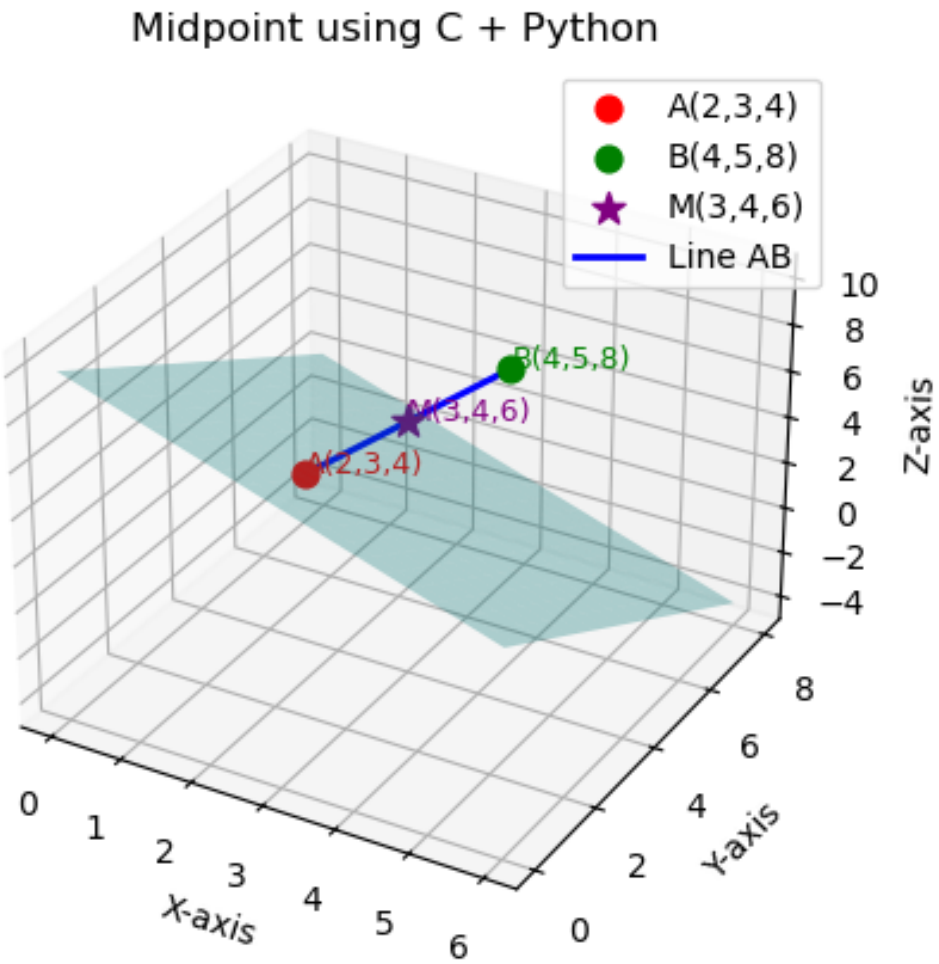
```
# Points and line
ax.scatter(*A, color='red', s=60, label='A(2,3,4)')
ax.scatter(*B, color='green', s=60, label='B(4,5,8)')
ax.scatter(*M, color='purple', s=100, marker='*', label='M(3,4,6)
    ')
ax.plot([A[0], B[0]], # x coordinates
        [A[1], B[1]], # y coordinates
        [A[2], B[2]], # z coordinates
        color='blue', linewidth=2, label='Line AB')
ax.text(*A, 'A(2,3,4)', fontsize=9, color='red')
ax.text(*B, 'B(4,5,8)', fontsize=9, color='green')
ax.text(*M, 'M(3,4,6)', fontsize=9, color='purple')

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

Python Code

```
plt.title('Midpoint using C + Python')
plt.savefig(/media/indhiresh-s/New Volume/Matrix/ee1030-2025/
    ee25btech11027/MATGEO/2.4.22/figs/figure1.png)
plt.show()
```

Plot


Midpoint using C + Python
A(2,3,4)
B(4,5,8)
M(3,4,6)
Line AB
B(4,5,8)
M(3,4,6)
A(2,3,4)
X-axis
Y-axis
Z-axis
0
1
2
3
4
5
6
0
2
4
6
8
10
8
6
4
2
0
−2
−4