4.5.12

Equation of plane

EE25BTECH11010 - Arsh Dhoke

Question

Find the equation of the plane passing through (a, b, c) and parallel to the plane $\mathbf{r} \cdot (\mathbf{i} + \mathbf{j} + \mathbf{k}) = 2$

Input parameters

Description	Vector
Normal to plane($\mathbf{n}$)	$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$
Point on plane($\mathbf{P}$)	$\begin{pmatrix} a \\ b \\ c \end{pmatrix}$

Solution

Given plane:

$$\mathbf{n}^T\mathbf{x} = 2 \tag{1}$$

Required plane (parallel to given plane):

$$\mathbf{n}^T\mathbf{x} = d \tag{2}$$

Finding d

Substitute given point in required plane:

$$\begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} a \\ b \\ c \end{pmatrix} = d \tag{3}$$

Thus,

$$d = a + b + c \tag{4}$$

Final answer

$$\boxed{\mathbf{n}^T\mathbf{x} = a + b + c}$$

This can also be written in the form: $$\boxed{\mathbf{r} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = a + b + c}$$

Plot

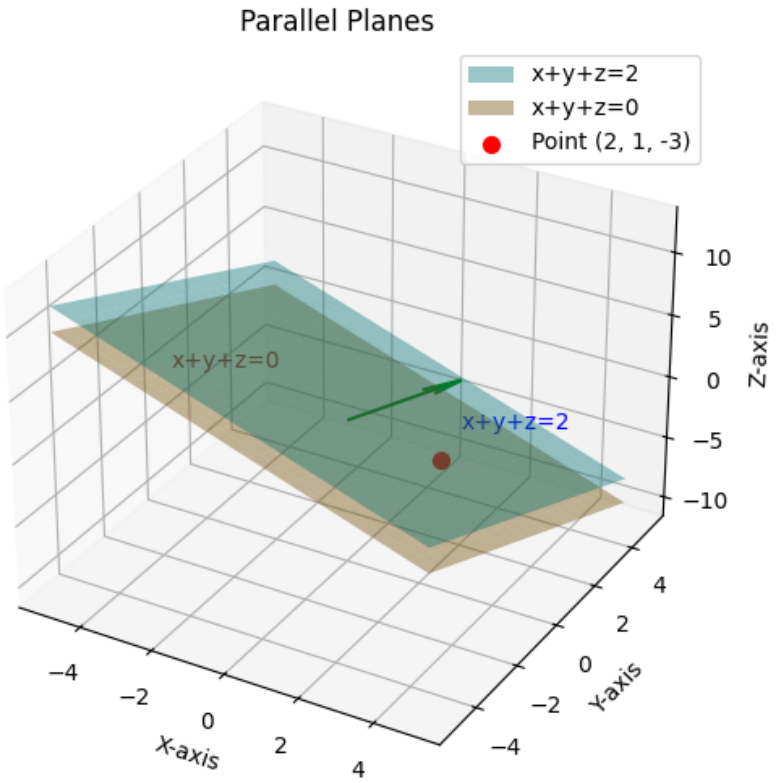

Figure: Graph

C Code

```c
#include <stdio.h>

double plane_equation(double a, double b, double c) {
    return a + b + c;
}
```

Python Code

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

def plane_equation(point):
    """Returns d for the plane x + y + z = d through point (a, b,
        c)."""
    a, b, c = point
    return a + b + c

def plot_planes(point):
    # Normal vector
    n = np.array([1, 1, 1])

    # Given plane: x + y + z = 2
    d1 = 2

    # Required plane: x + y + z = a+b+c
    d2 = plane_equation(point)
```

Python Code

```
# Meshgrid for plotting
x = np.linspace(-5, 5, 20)
y = np.linspace(-5, 5, 20)
X, Y = np.meshgrid(x, y)

# z for each plane
Z1 = (d1 - X - Y) / n[2]
Z2 = (d2 - X - Y) / n[2]

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

# Given plane
ax.plot_surface(X, Y, Z1, alpha=0.4, color='cyan', label="x+y
    +z=2")
ax.text(2, 2, (d1 - 2 - 2), "x+y+z=2", color='blue', fontsize
    =10)
```

Python Code

```
    # Required plane
    ax.plot_surface(X, Y, Z2, alpha=0.4, color='orange', label=f"
        x+y+z={d2}")
    ax.text(-3, -3, (d2 - (-3) - (-3)), f"x+y+z={d2}", color='
        brown', fontsize=10)

    # Mark the given point with legend
    ax.scatter(point[0], point[1], point[2], color='red', s=50,
        label=f"Point {point}")

    # Normal vector arrow
    origin = np.array([0, 0, 0])
    ax.quiver(*origin, *n, length=2, color="green")
```

Python Code

```
    # Axes labels
    ax.set_xlabel("X-axis")
    ax.set_ylabel("Y-axis")
    ax.set_zlabel("Z-axis")
    ax.set_title("Parallel Planes")

    # Show legend
    ax.legend()
    plt.savefig("/home/arsh-dhoke/ee1030-2025/ee25btech11010/
        matgeo/4.5.12/figs/q7.png")
    plt.show()

# Example usage
point = (2, 1, -3)
plot_planes(point)
```

Python+ C Code

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

# Load the shared library
lib = ctypes.CDLL('./code.so')

# Tell Python about the argument and return types
lib.plane_equation.argtypes = (ctypes.c_double, ctypes.c_double,
    ctypes.c_double)
lib.plane_equation.restype = ctypes.c_double

# Example point (a,b,c) that lies on the plane
a, b, c = 2.0, 1.0,-3.0

# Use C function to calculate d = a+b+c
d = lib.plane_equation(a, b, c)
print(f"Plane equation: x + y + z = {d}")
```

Python+ C Code

```
# Define a meshgrid for x and y
x_vals = np.linspace(-5, 5, 50)
y_vals = np.linspace(-5, 5, 50)
X, Y = np.meshgrid(x_vals, y_vals)

# Solve for Z using x + y + z = d
Z = d - X - Y

# Plot the plane
fig = plt.figure(figsize=(8,6))
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface(X, Y, Z, alpha=0.7, color="lightblue", edgecolor=
    'k')

# Plot the example point
ax.scatter(a, b, c, color='red', s=50, label=f'Point ({a},{b},{c
    })')
```

Python+ C Code

```
# Label axes
ax.set_xlabel("X")
ax.set_ylabel("Y")
ax.set_zlabel("Z")
ax.set_title(f"Plane: x + y + z = {d}")

plt.legend()
plt.savefig("/home/arsh-dhoke/ee1030-2025/ee25btech11010/matgeo
    /4.5.12/figs/q7.png")
plt.show()
```