5.2.31

J.NAVYASRI- EE25BTECH11028

september 2025

Question:

solve the following system of linear equations .

$$2x + 3y = 8$$

$$4x + 6y = 7$$

Solution:

Consider the system of linear equations:

$$2x + 3y = 8 \tag{1}$$

$$4x + 6y = 7 \tag{2}$$

$$\underbrace{\begin{pmatrix} 2 & 3 \\ 4 & 6 \end{pmatrix}}_{(A)} \underbrace{\begin{pmatrix} x \\ y \end{pmatrix}}_{(X)} = \underbrace{\begin{pmatrix} 8 \\ 7 \end{pmatrix}}_{(B)} \tag{3}$$

Solution:

$$\det(A) = \begin{vmatrix} 2 & 3 \\ 4 & 6 \end{vmatrix} = (2)(6) - (3)(4) = 12 - 12 = 0 \tag{4}$$

Since the determinant is zero, the system is either inconsistent or has infinitely many solutions.
Compare ratios of coefficients and constants:

$$\frac{2}{4} = \frac{3}{6} = \frac{1}{2} \quad \text{but} \quad \frac{8}{7} \neq \frac{1}{2} \tag{5}$$

The system is **inconsistent**. Therefore,

$$\text{No solution exists.} \tag{6}$$

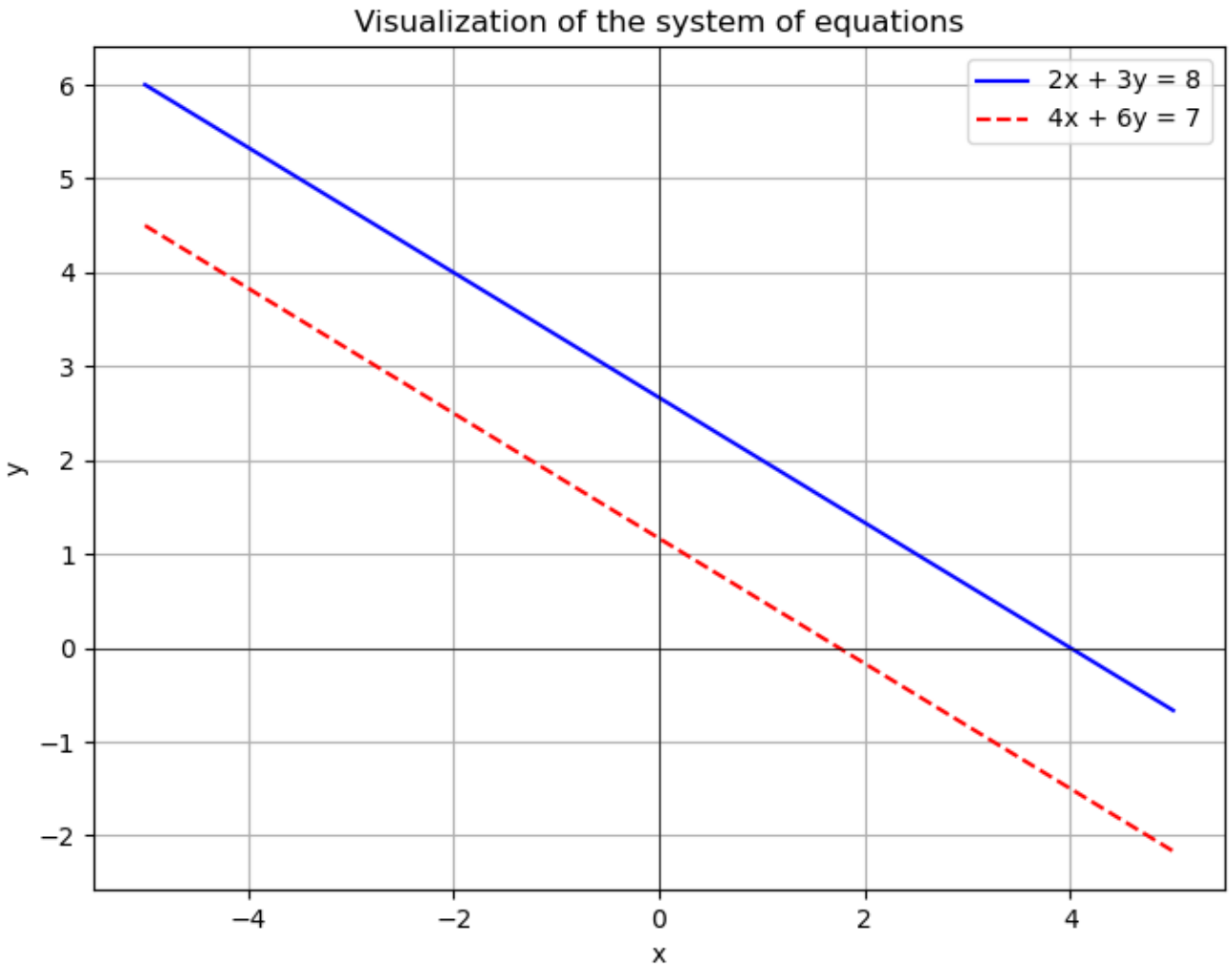
Python Code

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

# Define x values
x = np.linspace(-5, 5, 400)

# Line 1: 2x + 3y = 8 => y = (8 - 2x)/3
y1 = (8 - 2*x)/3

# Line 2: 4x + 6y = 7 => y = (7 - 4*x)/6
y2 = (7 - 4*x)/6

# Plot the lines
plt.figure(figsize=(8,6))
plt.plot(x, y1, label='2x + 3y = 8', color='blue')
```

Python Code

```
plt.plot(x, y2, label='4x + 6y = 7', color='red', linestyle='--')

# Labels and grid
plt.xlabel('x')
plt.ylabel('y')
plt.title('Visualization of the system of equations')
plt.grid(True)
plt.axhline(0, color='black', linewidth=0.5)
plt.axvline(0, color='black', linewidth=0.5)
plt.legend()
plt.savefig("fig10.png")
plt.show()
```

Plot-Using Python



C Code

```c
    #include <stdio.h>

int main() {
    // Coefficient matrix A
    double A[2][2] = {{2, 3}, {4, 6}};
    // Right-hand side vector B
    double B[2] = {8, 7};

    // Compute determinant
    double det = A[0][0]*A[1][1] - A[0][1]*A[1][0];

    if(det != 0) {
        // If determinant is non-zero, solve using Cramer's rule
        double x = (B[0]*A[1][1] - B[1]*A[0][1]) / det;
        double y = (A[0][0]*B[1] - A[1][0]*B[0]) / det;
```

C Code

```
        printf("Unique solution:\n");
        printf("x = %.2lf\n", x);
        printf("y = %.2lf\n", y);
    } else {
        // Determinant is zero, check for consistency
        if((A[0][0]*B[1] - A[1][0]*B[0] != 0) || (A[0][1]*B[1] -
            A[1][1]*B[0] != 0)) {
            printf("The system is inconsistent. No solution exists
                .\n");
        } else {
            printf("The system has infinitely many solutions.\n");
        }
    }

    return 0;
}
```

Python and C Code

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

# Load the C library
lib = ctypes.CDLL("./linear.so")

# Define array size
n = 400
x = np.linspace(-10, 10, n)
```

Python and C Code

```
# Create empty arrays for y1, y2
y1 = np.zeros(n, dtype=np.double)
y2 = np.zeros(n, dtype=np.double)

# Convert numpy arrays to ctypes pointers
lib.compute_lines.argtypes = [np.ctypeslib.ndpointer(dtype=np.
    double, ndim=1, flags="C_CONTIGUOUS"),
                              np.ctypeslib.ndpointer(dtype=np.double
                                  , ndim=1, flags="C_CONTIGUOUS"),
                              np.ctypeslib.ndpointer(dtype=np.double
                                  , ndim=1, flags="C_CONTIGUOUS"),
                              ctypes.c_int]

lib.compute_lines(x, y1, y2, n)
```

Python and C Code

```
# Plot the results
plt.plot(x, y1, label="2x + 3y - 8 = 0")
plt.plot(x, y2, label="4x + 6y - 7 = 0")
plt.xlabel("x")
plt.ylabel("y")
plt.title("Plot using C computations via ctypes")
plt.grid(True)
plt.legend()
plt.show()
```

Plot-Using by C and Python