5.8.17

EE25BTECH11019 – Darji Vivek M.

Question

Question:
A fraction becomes $\frac{9}{11}$ if 2 is added to both the numerator and the denominator. If 3 is added to both the numerator and the denominator, it becomes $\frac{5}{6}$. Find the fraction.

Solution

Matrix Method:

Let the numerator be n and the denominator be d.

From the given conditions,

$$\frac{n+2}{d+2} = \frac{9}{11} \implies 11(n+2) = 9(d+2) \implies 11n - 9d = -4, \tag{1}$$

$$\frac{n+3}{d+3} = \frac{5}{6} \implies 6(n+3) = 5(d+3) \implies 6n - 5d = -3. \tag{2}$$

In matrix form:

$$\begin{pmatrix} 11 & -9 \\ 6 & -5 \end{pmatrix} \begin{pmatrix} n \\ d \end{pmatrix} = \begin{pmatrix} -4 \\ -3 \end{pmatrix}. \tag{3}$$

Solution

Augmented matrix and row-reduction:

$$\left(\begin{array}{cc|c} 11 & -9 & -4 \\ 6 & -5 & -3 \end{array}\right) \xrightarrow{R_1 \leftarrow \frac{1}{11}R_1,\ R_2 \leftarrow R_2 - 6R_1} \left(\begin{array}{cc|c} 1 & -\frac{9}{11} & -\frac{4}{11} \\ 0 & -\frac{1}{11} & -\frac{9}{11} \end{array}\right) \xrightarrow{R_2 \leftarrow -11R_2,\ R_1 \leftarrow R_1 + \frac{9}{11}R_2} \left(\begin{array}{cc|c} 1 & 0 & 7 \\ 0 & 1 & 9 \end{array}\right)$$

Therefore:

$$n = 7, \qquad d = 9. \tag{4}$$

Hence, the required fraction is

$$\boxed{\frac{7}{9}}.$$

Verification: $\frac{7+2}{9+2} = \frac{9}{11}, \quad \frac{7+3}{9+3} = \frac{10}{12} = \frac{5}{6}.$

C code

```
#include <stdio.h>

// Function for first line: 11x - 9y = -4        y = (11
    x + 4)/9
float line1(float x) {
    return (11*x + 4)/9;
}

// Function for second line: 6x - 5y = -3        y = (6x
    + 3)/5
float line2(float x) {
    return (6*x + 3)/5;
}

// Function to find intersection point of two lines
void findIntersection(float *x, float *y) {
    float a1 = 11, b1 = -9, c1 = -4;
    float a2 = 6,  b2 = -5, c2 = -3;
```

C code

```
    float det = a1*b2 - a2*b1;

    if(det == 0) {
        printf("Lines are parallel.\n");
        return;
    }

    *x = (b1*c2 - b2*c1)/det;
    *y = (c1*a2 - c2*a1)/det;
}
```

Python

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

# Load the shared library
lib = ctypes.CDLL("./12.so")

# Define function argument and return types
lib.line1.argtypes = [ctypes.c_float]
lib.line1.restype = ctypes.c_float

lib.line2.argtypes = [ctypes.c_float]
lib.line2.restype = ctypes.c_float

lib.findIntersection.argtypes = [ctypes.POINTER(ctypes
    .c_float), ctypes.POINTER(ctypes.c_float)]

# Prepare intersection variables
x_inter = ctypes.c_float()
```

Python

```
# Call C function to find intersection
lib.findIntersection(ctypes.byref(x_inter), ctypes.
    byref(y_inter))

print("Intersection point from C:", x_inter.value,
    y_inter.value)

# Prepare x values and call C functions for plotting
x_vals = np.linspace(-5, 5, 100)
y1_vals = np.array([lib.line1(ctypes.c_float(x)) for x
    in x_vals])
y2_vals = np.array([lib.line2(ctypes.c_float(x)) for x
    in x_vals])

# Plot lines
plt.plot(x_vals, y1_vals, label="11x - 9y = -4", color
    ='blue')
plt.plot(x_vals, y2_vals, label="6x - 5y = -3", color=
    'red')
```

Python

```
# Mark intersection point
plt.scatter(x_inter.value, y_inter.value, color='black
    ', label=f"({x_inter.value:.2f}, {y_inter.value:.2f
    })")

plt.xlabel("x")
plt.ylabel("y")
plt.title("Two Lines from C Functions (ctypes)")
plt.legend()
plt.grid(True)
plt.show()
```

Pyhton plot

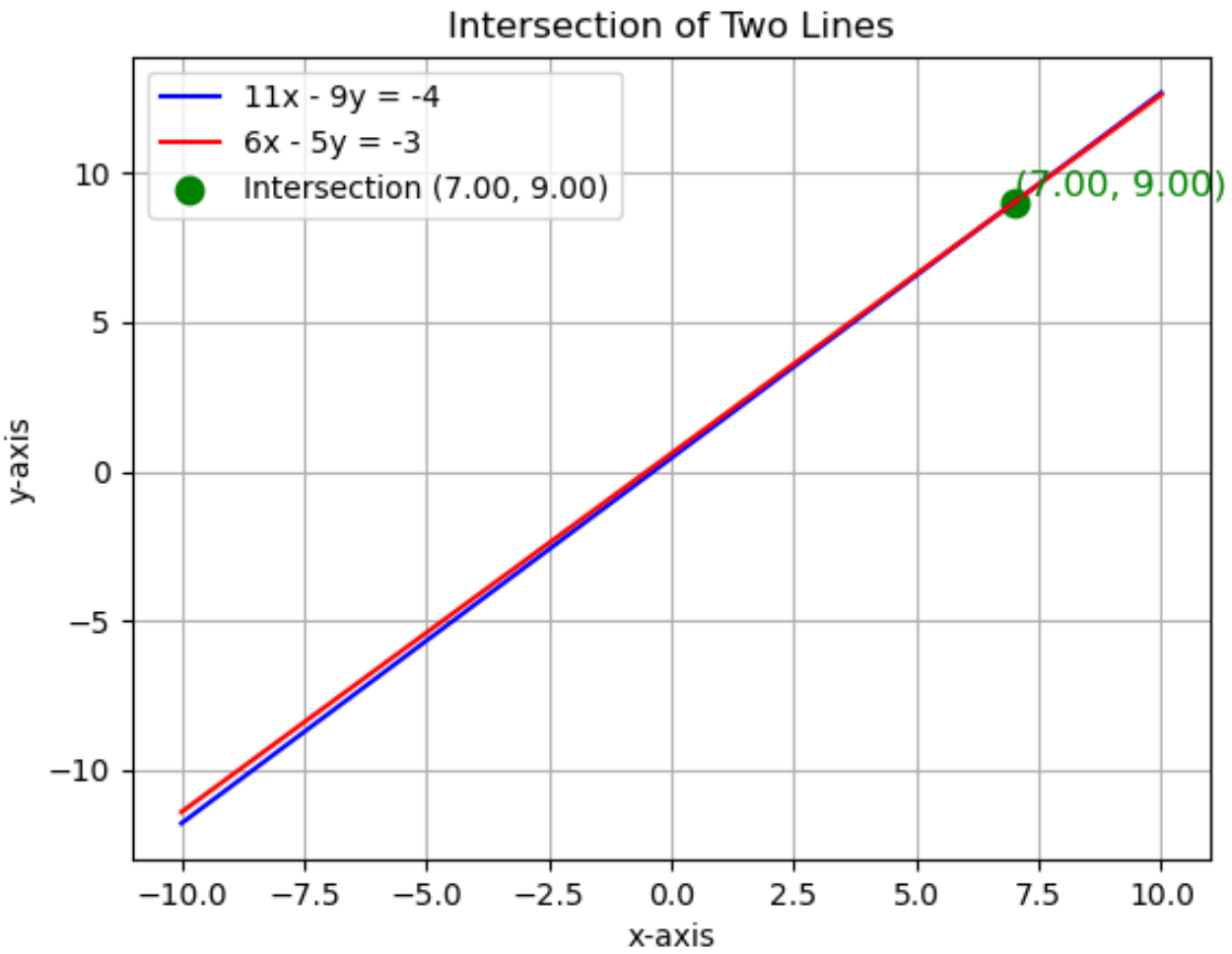

Figure: parallel lines