5.8.11

Bhargav - EE25BTECH11013

September 14, 2025

Question

The coach of a cricket team buys 3 bats and 6 balls for ₹3900. Later, she buys another bat and 3 more balls of the same kind for ₹3300. Find the cost of a bat and ball.

Solution

Let the cost of the bat, ball be ₹x and ₹y respectively.

$$\begin{pmatrix} 3 & 6 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 3900 \tag{1}$$

$$\begin{pmatrix} 1 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 3300 \tag{2}$$

These can be combined to give the matrix equation

$$\begin{pmatrix} 3 & 6 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3900 \\ 3300 \end{pmatrix} \tag{3}$$

Solution

This gives the augmented matrix

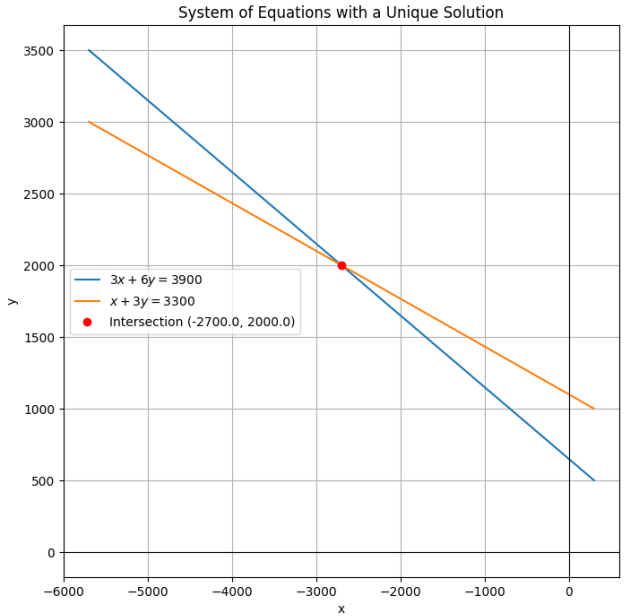
$$\left(\begin{array}{cc|c} 3 & 6 & 3900 \\ 1 & 3 & 3300 \end{array}\right) \xleftrightarrow{R_2 \leftarrow R_2 - \frac{1}{3}R_1} \left(\begin{array}{cc|c} 3 & 6 & 3900 \\ 0 & 1 & 2000 \end{array}\right) \tag{4}$$

This gives the following values of x and y:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} -2700 \\ 2000 \end{pmatrix} \tag{5}$$

Thus, the cost of one ball is ₹2000 and the cost of one bat is - ₹2700

Plot


System of Equations with a Unique Solution
3500
3000
2500
2000
1500
1000
500
0
y
−6000
−5000
−4000
−3000
−2000
−1000
0
x
3x + 6y = 3900
x + 3y = 3300
Intersection (-2700.0, 2000.0)

C Code

```c
#include <stdio.h>

int solve_2x2(double A[4], double b[2], double x[2]) {
    double det = A[0]*A[3] - A[1]*A[2];

    if (det == 0.0) {
        return -1;
    }

    x[0] = (b[0]*A[3] - b[1]*A[1]) / det;
    x[1] = (A[0]*b[1] - A[2]*b[0]) / det;

    return 0;
}
```

Python + C Code

```
import numpy as np
import matplotlib.pyplot as plt
import ctypes
lib = ctypes.CDLL(./code.so)
lib.solve_2x2.argtypes = [ctypes.POINTER(ctypes.c_double),
                          ctypes.POINTER(ctypes.c_double),
                          ctypes.POINTER(ctypes.c_double)]
lib.solve_2x2.restype = ctypes.c_int
A = np.array([[3, 6],
              [1, 3]], dtype=np.float64)
b = np.array([3900, 3300], dtype=np.float64)
x = np.zeros(2, dtype=np.float64)
status = lib.solve_2x2(A.ctypes.data_as(ctypes.POINTER(ctypes.
    c_double)),
                       b.ctypes.data_as(ctypes.POINTER(ctypes.
                           c_double)),
                       x.ctypes.data_as(ctypes.POINTER(ctypes.
                           c_double)))
```

Python + C Code

```
if status == 0:
    x_sol, y_sol = x
    print(fThe system has a unique solution:)
    print(fx = {x_sol})
    print(fy = {y_sol})
    x_vals = np.linspace(x_sol - 3000, x_sol + 3000, 400)
    y1 = (3900 - 3 * x_vals) / 6
    y2 = (3300 - x_vals) / 3
    plt.figure(figsize=(8, 8))
    plt.plot(x_vals, y1, label=r'$3x + 6y = 3900$')
    plt.plot(x_vals, y2, label=r'$x + 3y = 3300$')
    plt.plot(x_sol, y_sol, 'ro', label=f'Intersection ({x_sol:.2f
        }, {y_sol:.2f})')
    plt.axhline(0, color='black', linewidth=0.8)
    plt.axvline(0, color='black', linewidth=0.8)
    plt.xlabel(x)
    plt.ylabel(y)
```

Python + C Code

```
    plt.title(System of Equations with a Unique Solution)
    plt.legend()
    plt.grid(True)
    plt.savefig(/Users/bhargavkrish/Desktop/BackupMatrix/
        ee25btech11013/matgeo/5.2.16/figs/Figure_1.png)
    plt.show()
else:
    print(The system does not have a unique solution.)
```

Python Code

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

A = np.array([[3, 6],
              [1, 3]])
b = np.array([3900, 3300])

try:
    solution = np.linalg.solve(A, b)
    x_sol, y_sol = solution
    print(fThe system has a unique solution:)
    print(fx = {x_sol})
    print(fy = {y_sol})
    x_vals = np.linspace(x_sol - 3000, x_sol + 3000, 400)
    y1 = (3900 - 3 * x_vals) / 6
    y2 = (3300 - x_vals) / 3
    plt.figure(figsize=(8, 8))
    plt.plot(x_vals, y1, label=r'$3x + 6y = 3900$')
    plt.plot(x_vals, y2, label=r'$x + 3y = 3300$')
```

Python Code

```
    plt.plot(x_sol, y_sol, 'ro', label=f'Intersection ({x_sol}, {
        y_sol})')
    plt.axhline(0, color='black', linewidth=0.8)
    plt.axvline(0, color='black', linewidth=0.8)
    plt.xlabel(x)
    plt.ylabel(y)
    plt.title(System of Equations with a Unique Solution)
    plt.legend()
    plt.grid(True)
    plt.savefig(/Users/bhargavkrish/Desktop/BackupMatrix/
        ee25btech11013/matgeo/5.8.11/figs/Figure_1.png)
    plt.show()

except np.linalg.LinAlgError:
    print(The system does not have a unique solution.)
```