5.2.43

Naman Kumar-EE25BTECH11041

26 September,2025

Question)

Solve the linear equation:

$$6x + 3y = 6xy \tag{1}$$

$$2x + 4y = 5xy \tag{2}$$

Solution

Dividing both equations with xy

$$\frac{6}{y} + \frac{3}{x} = 6 \tag{3}$$

$$\frac{2}{y} + \frac{4}{x} = 5 \tag{4}$$

Let

$$\frac{1}{x} = a, \frac{1}{y} = b \tag{5}$$

Solution

So, new equations

$$3a + 6b = 6 \tag{6}$$

$$4a + 2b = 5 \tag{7}$$

$$\begin{pmatrix} 3 & 6 \\ 4 & 2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} \begin{pmatrix} 6 \\ 5 \end{pmatrix} \tag{8}$$

$$\mathbf{Ax} = \mathbf{c} \tag{9}$$

Gaussian elimination on $\mathbf{A}$

$$\left(\begin{array}{cc|c} 3 & 6 & 6 \\ 4 & 2 & 5 \end{array}\right) \xrightarrow{R_2 - \frac{4R_1}{3}} \left(\begin{array}{cc|c} 3 & 6 & 6 \\ 0 & -6 & -3 \end{array}\right) \tag{10}$$

$$\xrightarrow{\frac{R_2}{-6}} \left(\begin{array}{cc|c} 3 & 6 & 6 \\ 0 & 1 & \frac{1}{2} \end{array}\right) \tag{11}$$

Solution

Therefore,by putting values in (8)

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix} \tag{12}$$

For x,y

$$\begin{pmatrix} \frac{1}{x} \\ \frac{1}{y} \end{pmatrix} = \begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix} \tag{13}$$

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix} \tag{14}$$

Figure

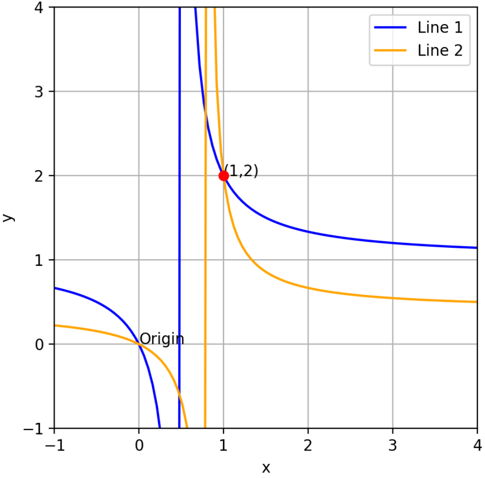

Figure:

Direct Python

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

plt.figure(figsize=(5,5), dpi=200)
plt.xlim(-1,4)
plt.ylim(-1,4)
A=np.array([[3,6],[4,2]])
c=np.array([6,5])

an=np.linalg.inv(A)
ans=np.dot(an,c)

Ans=np.linalg.solve(A,c)
```

Direct Python

```
print("x=",Ans[0],"y=",Ans[1])
a=1/ans[0]
b=1/ans[1]

x = np.array([a,b]).reshape(-1,1)

x1= np.linspace(-1,4,100)
l1= (6*x1)/(6*x1-3)
l2= (2*x1)/(5*x1-4)

plt.plot(x1,l1, color='blue', label="Line 1")
```

Direct Python

```
plt.plot(x1,l2, color='orange', label="Line 2")
plt.scatter(1,2, c='r', zorder=5)
plt.text(1,2,"(1,2)")
plt.text(0,0,"Origin",)

plt.xlabel("x")
plt.ylabel("y")
plt.grid()
plt.legend()
plt.savefig("figure.png", dpi=200)
plt.show()
```

C code

```c
#include <stdio.h>

typedef struct {
    double x;
    double y;
} Point;

typedef struct {
    Point sols[2];
    int count;
} SolutionSet;

// Solve 6x+3y=6xy, 2x+4y=5xy
SolutionSet solve_equations() {
    SolutionSet S;
    S.count = 0;
```

C code

```
    // Solution 1: (0,0)
    S.sols[S.count].x = 0;
    S.sols[S.count].y = 0;
    S.count++;

    // Solution 2: (1,2)
    S.sols[S.count].x = 1;
    S.sols[S.count].y = 2;
    S.count++;

    return S;
}
```

C code

```c
#ifdef TEST_C
int main(){
    SolutionSet S = solve_equations();
    for(int i=0; i<S.count; i++){
        printf("Solution %d: (%.2f, %.2f)\n", i+1, S.sols[i].x, S
            .sols[i].y);
    }
    return 0;
}
#endif
```

Python code with shared object

```
# main.py
import ctypes
from ctypes import Structure, c_double, c_int
import matplotlib.pyplot as plt

class Point(Structure):
    _fields_ = [("x", c_double), ("y", c_double)]

class SolutionSet(Structure):
    _fields_ = [("sols", Point * 2), ("count", c_int)]

# Load C lib
lib = ctypes.CDLL("./libsolver.so")
lib.solve_equations.restype = SolutionSet
```

Python code with shared object

```python
# Call function
solutions = lib.solve_equations()
print(f"Found {solutions.count} solutions:")
for i in range(solutions.count):
    x, y = solutions.sols[i].x, solutions.sols[i].y
    print(f"Solution {i+1}: ({x}, {y})")

# Plot equations and solutions
import numpy as np
x_vals = np.linspace(-1, 3, 400)
y1 = (6*x_vals)/(6*x_vals - 3) # from eqn (1)
y2 = (2*x_vals)/(5*x_vals - 4) # from eqn (2)
```

Python code with shared object

```
plt.figure(figsize=(6,6))
plt.plot(x_vals, y1, label="6x+3y=6xy")
plt.plot(x_vals, y2, label="2x+4y=5xy")
for i in range(solutions.count):
    x, y = solutions.sols[i].x, solutions.sols[i].y
    plt.scatter(x, y, c='r', zorder=5)
    plt.text(x, y, f"({x:.0f},{y:.0f})", fontsize=10)

plt.ylim(-1,4)
plt.grid(True)
plt.legend()
plt.xlabel("x")
plt.ylabel("y")
plt.title("Solutions of nonlinear system")
plt.show()
```