

4.10.17

Bhoomika V - EE25BTECH11015

September 14,2025

Question

Compute the area bounded by the lines $x + 2y = 2$, $y - x = 1$, and $2x + y = 7$.

equations of the line

$$L_1 : x + 2y = 2, \quad L_2 : y - x = 1, \quad L_3 : 2x + y = 7$$

Finding intersection points using RREF

Point A

Intersection of L_1 and L_2 :

$$\begin{cases} x + 2y = 2 \\ -x + y = 1 \end{cases}$$

$$\left[\begin{array}{cc|c} 1 & 2 & 2 \\ -1 & 1 & 1 \end{array} \right] \xrightarrow{\text{RREF}} \left[\begin{array}{cc|c} 1 & 0 & 0 \\ 0 & 1 & 1 \end{array} \right]$$

$$A = (0, 1)$$

Intersection of L_2 and L_3 :

$$\begin{cases} -x + y = 1 \\ 2x + y = 7 \end{cases}$$

$$\left[\begin{array}{cc|c} -1 & 1 & 1 \\ 2 & 1 & 7 \end{array} \right] \xrightarrow{\text{RREF}} \left[\begin{array}{cc|c} 1 & 0 & 2 \\ 0 & 1 & 3 \end{array} \right]$$

$$B = (2, 3)$$

Intersection of L_1 and L_3 :

$$\begin{cases} x + 2y = 2 \\ 2x + y = 7 \end{cases}$$

$$\left[\begin{array}{cc|c} 1 & 2 & 2 \\ 2 & 1 & 7 \end{array} \right] \xrightarrow{\text{RREF}} \left[\begin{array}{cc|c} 1 & 0 & 4 \\ 0 & 1 & -1 \end{array} \right]$$

$$C = (4, -1)$$

Area of the triangle

$$\Delta = \frac{1}{2} |(\mathbf{A} - \mathbf{B}) \times (\mathbf{B} - \mathbf{C})|$$

Compute:

$$\mathbf{A} - \mathbf{B} = \begin{bmatrix} -2 \\ -2 \end{bmatrix}, \quad \mathbf{B} - \mathbf{C} = \begin{bmatrix} -2 \\ 4 \end{bmatrix}$$

$$(\mathbf{A} - \mathbf{B}) \times (\mathbf{B} - \mathbf{C}) = \begin{vmatrix} -2 & -2 \\ -2 & 4 \end{vmatrix} = (-2)(4) - (-2)(-2) = -12$$

$$\Delta = \frac{1}{2} |-12| = 6$$

```
#include <math.h>

// Function to compute area of a triangle given 3 points (x,y,z)
float triangle_area(float x1, float y1, float z1,
                    float x2, float y2, float z2,
                    float x3, float y3, float z3) {

    // Using cross product method: Area = 0.5 * |AB x AC|
    float ABx = x2 - x1;
    float ABx = y2 - y1;
    float ABz = z2 - z1;

    float ACx = x3 - x1;
    float ACy = y3 - y1;
    float ACz = z3 - z1;
```



```
2 // Cross product AB x AC
3 float cross_x = ABy*ACz - ABz*ACy;
4 float cross_y = ABz*ACx - ABx*ACz;
5 float cross_z = ABx*ACy - ABy*ACx;
6
7 // Magnitude of cross product
8 float area = 0.5 * sqrt(cross_x*cross_x + cross_y*cross_y +
9     cross_z*cross_z);
10 return area;
11 }
```

Python Code

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

# --- Load the C library ---
try:
    c_lib = ctypes.CDLL('./code.so')
except OSError:
    print("Error: 'code.so' not found. Compile using: gcc -shared  
-o code.so -fPIC triangle.c -lm")
    exit()
```

Python Code

```
# Define argument and return types
c_lib.triangle_area.argtypes = [ctypes.c_float, ctypes.c_float,
                                ctypes.c_float,
                                ctypes.c_float, ctypes.c_float,
                                ctypes.c_float,
                                ctypes.c_float, ctypes.c_float,
                                ctypes.c_float]
c_lib.triangle_area.restype = ctypes.c_float

# --- Triangle vertices (intersection points of the lines) ---
A = np.array([0, 1, 0], dtype=np.float32) #  $x+2y=2$  &  $y-x=1$ 
B = np.array([2, 3, 0], dtype=np.float32) #  $y-x=1$  &  $2x+y=7$ 
C = np.array([3, -0.5, 0], dtype=np.float32) #  $x+2y=2$  &  $2x+y=7$ 

# --- Call C function ---
area = c_lib.triangle_area(A[0], A[1], A[2],
                           B[0], B[1], B[2],
                           C[0], C[1], C[2])

print(f"Area of triangle = {area:.2f}")
```

Python Code

```
# --- Plotting ---
fig, ax = plt.subplots(figsize=(6,6))

# Triangle edges
ax.plot([A[0], B[0]], [A[1], B[1]], color="black")
ax.plot([B[0], C[0]], [B[1], C[1]], color="black")
ax.plot([C[0], A[0]], [C[1], A[1]], color="black")

# Fill triangle
ax.fill([A[0], B[0], C[0]], [A[1], B[1], C[1]], color="cyan",
        alpha=0.3)

# Points
ax.scatter(A[0], A[1], color="red", s=60)
ax.scatter(B[0], B[1], color="blue", s=60)
ax.scatter(C[0], C[1], color="green", s=60)
```

Python Code

```
# Labels
ax.text(A[0]+0.1, A[1], "A(0,1)", color="red")
ax.text(B[0]+0.1, B[1], "B(2,3)", color="blue")
ax.text(C[0]+0.1, C[1], "C(3,-0.5)", color="green")

# Area annotation
ax.text(1.5, 1.5, f"Area = {area:.2f}", color="purple", fontsize
       =12)

# Formatting
ax.set_xlabel("X-axis")
ax.set_ylabel("Y-axis")
ax.set_title("Triangle formed by intersection of lines")
ax.grid(True)
ax.set_aspect("equal")

plt.show()
```

