2.7.25

EE25BTECH11013 - Bhargav

August 29, 2025

Question

Find the area of quadrilateral $ABCD$ whose vertices are $A(-3,-1)$, $B(-2,-4)$, $C(4,-1)$ and $D(3,4)$.

Theoretical Solution

The area of the quadrilateral can be found by dividing it into 2 triangles, $\triangle ABC$ and $\triangle ACD$, and then summing their areas.

The area of a triangle formed by vectors $\mathbf{u}$ and $\mathbf{v}$ originating from the same vertex is given by:

$$\text{Area} = \frac{1}{2}|\mathbf{u} \times \mathbf{v}| \tag{1}$$

The given vertices are:

$$\mathbf{A} = \begin{pmatrix} -3 \\ -1 \end{pmatrix} \quad \mathbf{B} = \begin{pmatrix} -2 \\ -4 \end{pmatrix} \quad \mathbf{C} = \begin{pmatrix} 4 \\ -1 \end{pmatrix} \quad \mathbf{D} = \begin{pmatrix} 3 \\ 4 \end{pmatrix} \tag{2}$$

Calculation: Triangle ABC

$$\mathbf{A}-\mathbf{B}=\begin{pmatrix}-3+2\\-1+4\end{pmatrix}=\begin{pmatrix}-1\\3\end{pmatrix} \tag{3}$$

$$\mathbf{A}-\mathbf{C}=\begin{pmatrix}-3-4\\-1+1\end{pmatrix}=\begin{pmatrix}-7\\0\end{pmatrix} \tag{4}$$

$$(\triangle ABC)=\frac{1}{2}\left\|(\mathbf{A}-\mathbf{B})\times(\mathbf{A}-\mathbf{C})\right\| \tag{5}$$

$$(\triangle ABC)=\frac{1}{2}\left\|\begin{pmatrix}-1\\3\end{pmatrix}\times\begin{pmatrix}-7\\0\end{pmatrix}\right\|=\frac{1}{2}\cdot 21=\frac{21}{2}. \tag{6}$$

Calculation: Triangle ACD

$$\mathbf{A} - \mathbf{D} = \begin{pmatrix} -3-3 \\ -1-4 \end{pmatrix} = \begin{pmatrix} -6 \\ -5 \end{pmatrix} \tag{7}$$

$$(\triangle ACD) = \frac{1}{2} \left\| (\mathbf{A} - \mathbf{C}) \times (\mathbf{A} - \mathbf{D}) \right\| = \frac{1}{2} \cdot 35 = \frac{35}{2} \tag{8}$$

Final Result

The total area of the quadrilateral $ABCD$ is the sum of the areas of the two triangles.

$$\text{Area}(ABCD) = \text{Area}(\triangle ABC) + \text{Area}(\triangle ACD) \tag{9}$$

$$= \frac{21}{2} + \frac{35}{2} \tag{10}$$

$$= \frac{56}{2} \tag{11}$$

$$= 28 \tag{12}$$

Therefore, the area of the quadrilateral is 28 square units.

C Code

```
#include <stdio.h>
#include <math.h>


double cross2D(double x1, double y1, double x2, double y2) {
    return fabs(x1*y2 - y1*x2);
}


double triangle_area(double ax, double ay, double bx, double by, double cx
    , double cy) {
    double v1x = bx - ax;
    double v1y = by - ay;
    double v2x = cx - ax;
```

C Code

```
    double v2y = cy - ay;
    return 0.5 * cross2D(v1x, v1y, v2x, v2y);
}


double quad_area(double ax, double ay, double bx, double by, double cx,
    double cy, double dx, double dy) {
    double area1 = triangle_area(ax, ay, bx, by, cx, cy);
    double area2 = triangle_area(ax, ay, cx, cy, dx, dy);
    return area1 + area2;
}
```

Python + C Code

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


lib = ctypes.CDLL("./libquad.so")


lib.triangle_area.argtypes = [ctypes.c_double, ctypes.c_double,
                              ctypes.c_double, ctypes.c_double,
                              ctypes.c_double, ctypes.c_double]
lib.triangle_area.restype = ctypes.c_double
```

Python + C Code

```
lib.quad_area.argtypes = [ctypes.c_double, ctypes.c_double,
                          ctypes.c_double, ctypes.c_double,
                          ctypes.c_double, ctypes.c_double,
                          ctypes.c_double, ctypes.c_double]
lib.quad_area.restype = ctypes.c_double

# Vertices of quadrilateral
A = (−3.0, −1.0)
B = (−2.0, −4.0)
C = (4.0, −1.0)
D = (3.0, 4.0)



area = lib.quad_area(A[0], A[1], B[0], B[1], C[0], C[1], D[0], D[1])
print("Area of Quadrilateral ABCD =", area)
```

Python + C Code

```
points = np.array([A, B, C, D, A])
plt.plot(points[:,0], points[:,1], 'b-o')
plt.fill(points[:,0], points[:,1], color='skyblue', alpha=0.5)
plt.text(A[0], A[1], "A")
plt.text(B[0], B[1], "B")
plt.text(C[0], C[1], "C")
plt.text(D[0], D[1], "D")

plt.title(f"Quadrilateral ABCD (Area = {area:.2f})")
plt.xlabel("x")
plt.ylabel("y")
plt.grid(True)
plt.axis("equal")
plt.savefigs("/Users/bhargavkrish/Documents/ee1030-2025/
    ee25btech11013/matgeo/2.7.25/figs/Figure_1.png")
plt.show()
```

Python Code

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


def triangle_area(A, B, C):
    v1 = np.array(B) - np.array(A)
    v2 = np.array(C) - np.array(A)
    return 0.5 * abs(np.cross(v1, v2))


def quad_area(A, B, C, D):
    return triangle_area(A, B, C) + triangle_area(A, C, D)
```

Python Code

```
A = (−3, −1)
B = (−2, −4)
C = (4, −1)
D = (3, 4)

area = quad_area(A, B, C, D)
print("Area of Quadrilateral ABCD =", area)

points = np.array([A, B, C, D, A])
plt.plot(points[:,0], points[:,1], 'b−o')
plt.fill(points[:,0], points[:,1], color='lightgreen', alpha=0.5)
```

Python Code

```
for point, label in zip([A, B, C, D], ["A", "B", "C", "D"]):
    plt.text(point[0], point[1], label, fontsize=12, ha='right')

plt.title(f"Quadrilateral ABCD (Area = {area:.2f})")
plt.xlabel("x")
plt.ylabel("y")
plt.grid(True)
plt.axis("equal")
plt.savefig("/Users/bhargavkrish/Documents/ee1030-2025/
    ee25btech11013/matgeo/2.7.25/figs/Figure_1.png")
plt.show()
```

Plot