4.11.10

Kartik Lahoti - EE25BTECH11032

September 13 ,2025

Question

Point **A** lies on the line segment **XY** joining **X** $(6, -6)$ and **Y** $(-4, -1)$ in such a way that $\frac{\mathbf{XA}}{\mathbf{XY}} = \frac{2}{5}$. if point **A** also lies on the line $3x + k(y + 1) = 0$, find the value of k .

Theoretical Solution

Symbol	Value	Description
X	$\begin{pmatrix} 6 \\ -6 \end{pmatrix}$	Given Point
Y	$\begin{pmatrix} -4 \\ -1 \end{pmatrix}$	Given Point
A	?	Desired Point

Table: 4.11.10

Given Line Equation,

$$\begin{pmatrix} 3 & k \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + k = 0 \tag{1}$$

Theoretical Solution

Using Section Formula,

$$\mathbf{A} = \frac{1}{1+p}\left(\mathbf{X} + p\mathbf{Y}\right) \tag{2}$$

From the Question, $p = \frac{2}{3}$
Substituting the values,

$$\mathbf{A} = \frac{1}{1+\frac{2}{3}}\left(\begin{pmatrix}6\\-6\end{pmatrix} + \frac{2}{3}\begin{pmatrix}-4\\-1\end{pmatrix}\right) = \begin{pmatrix}2\\-4\end{pmatrix} \tag{3}$$

Theoretical Solution

Putting **A** in the given line equation,

$$\begin{pmatrix} 3 & k \end{pmatrix} \begin{pmatrix} 2 \\ -4 \end{pmatrix} + k = 0 \tag{4}$$

$$6 - 4k + k = 0 \tag{5}$$

$$k = 2 \tag{6}$$

Theoretical Solution

Hence,

$$\mathbf{A} = \begin{pmatrix} 2 \\ -4 \end{pmatrix} \text{ and } k = 2 \tag{7}$$

C Code (1)

```c
void section_formula ( double *X , double *Y , double *A , double
     ratio , int m )
{
    for(int i = 0; i < m ; i++)
        A[i] = (1/(1+ratio))*(X[i] + ratio * Y[i] );
}
```

C Code (2) - Function to Generate Points on Line

```c
void linegen(double *XY, double *A , double *B , int n , int m )
{
    double temp[m] ;
    for (int i = 0 ; i < m ; i++)
    {
        temp [ i ] = (B[i]- A[i]) /(double) n ;
    }
    for (int i = 0 ; i < n ; i++ )
        for (int j = 0 ; j < m ; j++)
            XY[j*n + i ] = A[j] + temp[j] * i ;

}
```

Python Code - Using Shared Object

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

handc1 = ct.CDLL("./func.so")

handc1.section_formula.argtypes = [
    ct.POINTER(ct.c_double),
    ct.POINTER(ct.c_double),
    ct.POINTER(ct.c_double),
    ct.c_double,
    ct.c_int
]
handc1.section_formula.restype = None
```

Python Code - Using Shared Object

```
X = np.array([6,-6] , dtype = np.float64).reshape(-1,1)
Y = np.array([-4,-1] , dtype = np.float64).reshape(-1,1)
A = np.zeros(2, dtype = np.float64).reshape(-1,1)
p = 2 / 3
handc1.section_formula(
    X.ctypes.data_as(ct.POINTER(ct.c_double)),
    Y.ctypes.data_as(ct.POINTER(ct.c_double)),
    A.ctypes.data_as(ct.POINTER(ct.c_double)),
    p , 2
)
print("Vector A = ",A)
```

Python Code - Using Shared Object

```
M = np.array([-10,14] , dtype = np.float64).reshape(-1,1)
N = np.array([10,-16] , dtype = np.float64).reshape(-1,1)

def line(P: np.ndarray , Q: np.ndarray, str1 , str2):
    handc2 = ct.CDLL("./line_gen.so")

    handc2.linegen.argtypes = [
        ct.POINTER(ct.c_double),
        ct.POINTER(ct.c_double),
        ct.POINTER(ct.c_double),
        ct.c_int , ct.c_int
    ]
    handc2.linegen.restype = None
```

Python Code - Using Shared Object

```
n = 200
XY = np.zeros((2,n),dtype=np.float64)

handc2.linegen (
    XY.ctypes.data_as(ct.POINTER(ct.c_double)),
    P.ctypes.data_as(ct.POINTER(ct.c_double)),
    Q.ctypes.data_as(ct.POINTER(ct.c_double)),
    n,2
)
plt.plot(XY[0,:],XY[1,:], str1 , label = str2 )
```

Python Code - Using Shared Object

```
plt.figure()

line(X,Y,"g--","Line Segment XY")
line(M,N,"r-","Line Containing A")
coords = np.block([[X,Y,A]])

plt.scatter(coords[0,:] , coords[1,:])
vert_label = ['X', 'Y' , 'A' ]

for i , txt in enumerate(vert_label) :
    plt.annotate(f"{txt}\n({coords[0,i]:.1f},{coords[1,i]:.1f})",
                 (coords[0,i], coords[1,i]),
                 textcoords = "offset points" ,
                 xytext = (0,-30),ha = "center")
```

Python Code - Using Shared Object

```
plt.xlim([-6,8])
plt.ylim([-8,0])
plt.xlabel("$x$")
plt.ylabel("$y$")
plt.grid()

plt.legend(loc="best")

plt.title("4.11.10")

plt.savefig("../figs/section1.png")
plt.show()

#plt.savefig('../figs/section1.png')
#subprocess.run(shlex.split("termux-open ../figs/section1.png"))
```

Python Code

```
mport math
import sys
sys.path.insert(0, '/home/kartik-lahoti/matgeo/codes/CoordGeo')
import numpy as np
import numpy.linalg as LA
import matplotlib.pyplot as plt

from line.funcs import *

#if using termux
#import subprocess
#import shlex
```

Python Code

```
X = np.array([6,-6] , dtype = np.float64).reshape(-1,1)
Y = np.array([-4,-1] , dtype = np.float64).reshape(-1,1)
k = 2 / 3
A = (1/(1+k))*(X + k * Y)

print("Vector A = " , A )

M = np.array([-10,14] , dtype = np.float64).reshape(-1,1)
N = np.array([10,-16] , dtype = np.float64).reshape(-1,1)
```

Python Code

```
def plot_it(P,Q,str1,str2):
    x_l = line_gen_num(P,Q,20)
    plt.plot(x_l[0,:],x_l[1,:] , str1 , label = str2)

plt.figure()

plot_it(X,Y,"g--","Line Segment XY")
plot_it(M,N,"r-","Line Containing A")
coords = np.block([[X,Y,A]])
plt.scatter(coords[0,:],coords[1,:])
vert_labels = ['X','Y','A']
```

Python Code

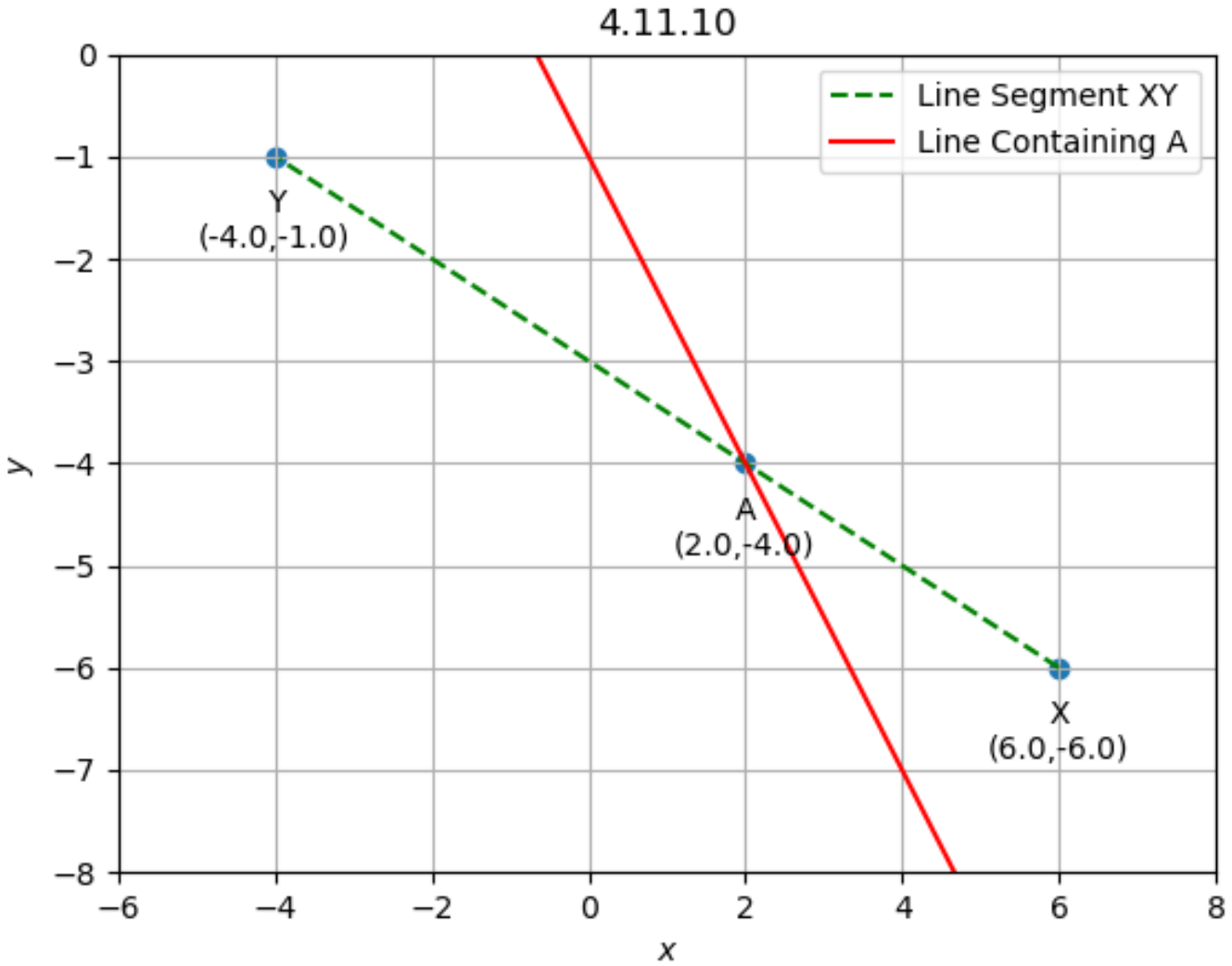
```
for i, txt in enumerate(vert_labels):
    plt.annotate(f'{txt}\n({coords[0,i]:.1f}, {coords[1,i]:.1f})'
        ,
                 (coords[0,i], coords[1,i]),
                 textcoords="offset points",
                 xytext=(0,-30),
                 ha='center')
```

Python Code

```
plt.xlim([-6,8])
plt.ylim([-8,0])
plt.xlabel('$x$')
plt.ylabel('$y$')
plt.grid()
plt.legend(loc = "best")
plt.title("Fig:4.11.10")
plt.savefig("../figs/section2.png")
plt.show()

#plt.savefig('../figs/section1.png')
#subprocess.run(shlex.split("termux-open ../figs/section1.png"))
```

Plot


4.11.10
Line Segment XY
Line Containing A
Y
(-4.0,-1.0)
A
(2.0,-4.0)
X
(6.0,-6.0)
x
y
0
−1
−2
−3
−4
−5
−6
−7
−8
−6
−4
−2
0
2
4
6
8