1.5.16

J.NAVYASRI- EE25BTECH11028

August 2025

Question

Find the coordinates point A where AB is a diameter of the circle with center $= (3, -1)$ and point $B = (2, 6)$.

given data

Point	Vector
B	$\begin{pmatrix} 2 \\ 6 \end{pmatrix}$
P	$\begin{pmatrix} 3 \\ -1 \end{pmatrix}$

Theoretical Solution

Theory : Center of a circle is the mid-point of the diameter.

Let P be the center of the given circle , with AB as the diameter.
Let **A** be the Vector to be found
Given :

$$B \equiv \begin{pmatrix} 2 \\ 6 \end{pmatrix}, \quad P \equiv \begin{pmatrix} 3 \\ -1 \end{pmatrix}$$

Theoretical Solution

Center of a circle is the mid point of the diameter. For a circle with center **P** and ends of diameters represented by vectors **A** and **B**

$$\mathbf{P} = \frac{\mathbf{A} + \mathbf{B}}{2} \tag{0.1}$$

Rearranging , we get:

$$\mathbf{A} = 2\mathbf{P} - \mathbf{B} \tag{0.2}$$

Theoretical Solution

Substituting the given vectors, we get:

$$\mathbf{A} = 2\begin{pmatrix} 3 \\ -1 \end{pmatrix} - \begin{pmatrix} 2 \\ 6 \end{pmatrix} \tag{0.3}$$

$$\mathbf{A} = \begin{pmatrix} 6 \\ -2 \end{pmatrix} - \begin{pmatrix} 2 \\ 6 \end{pmatrix} \tag{0.4}$$

$$\therefore \mathbf{A} \equiv \begin{pmatrix} 4 \\ -8 \end{pmatrix}$$

Hence , Coordinates of A are

$$\begin{pmatrix} 4 \\ -8 \end{pmatrix}$$

C Code (1) - Function to find A matrix

```c
#include <stdio.h>
#include <math.h>
void func(double *P, double *B, double *A , int m )
{
    for ( int i = 0 ; i < m ; i++ )
    {
        A[i] = 2*P[i] - B[i] ;
    }

}
```

C Code (1) - Function to Find Radius

```c
double radius(double *P , double *B , int m )
{
    double sum = 0.0;
    for ( int i = 0 ; i < m ; i++ )
    {
        sum += pow(P[i]-B[i] , 2 );
    }
    return sqrt(sum) ;
}
```

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

```c
#include <math.h>

void circle_gen(double *X , double *Y , double *P, int n , double
     r)
{
// n is no. of points to generates. x stores x coor , y stores y
    coor
    for (int i = 0 ; i < n ; i++ )
    {
        double theta = 2.0 * M_PI * i / n ;
        X[i] = P[0] + r * cos(theta);
        Y[i] = P[1] + r * sin (theta);
    }
}
```

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

```
void line_gen (double *X, double *Y , 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++ )
    {
        X[i] = A[0] + temp[0] * i ;
        Y[i] = A[1] + temp[1] * i ;
    }
}
```

Python Code

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

# Load shared C library
lib = ctypes.CDLL("./formula.so")

# Define argument and return types
lib.find_A.argtypes = [
    np.ctypeslib.ndpointer(dtype=np.float64, ndim=1, flags="
        C_CONTIGUOUS"),
    np.ctypeslib.ndpointer(dtype=np.float64, ndim=1, flags="
        C_CONTIGUOUS"),
    np.ctypeslib.ndpointer(dtype=np.float64, ndim=1, flags="
        C_CONTIGUOUS"),
]
lib.find_A.restype = None
```

Python Code

```
# Input vectors
P = np.array([3.0, -1.0], dtype=np.float64)
B = np.array([2.0, 6.0], dtype=np.float64)
A = np.zeros(2, dtype=np.float64)

# Call C function
lib.find_A(P, B, A)

print("A =", A)

# Plotting
fig, ax = plt.subplots()

# Circle parameters
center = P
radius = np.linalg.norm(A - P) # radius = distance from center to
    A (or B)
```

Python Code

```
theta = np.linspace(0, 2*np.pi, 300)
x_circle = center[0] + radius*np.cos(theta)
y_circle = center[1] + radius*np.sin(theta)
ax.plot(x_circle, y_circle, 'b')

# Plot diameter line AB
ax.plot([A[0], B[0]], [A[1], B[1]], 'g--')

# Midpoint of AB for placing text "Diameter"
mid_x = (A[0] + B[0]) / 2
mid_y = (A[1] + B[1]) / 2
ax.text(mid_x + 0.5, mid_y + 0.5, "Diameter", color="green")
```

Python Code

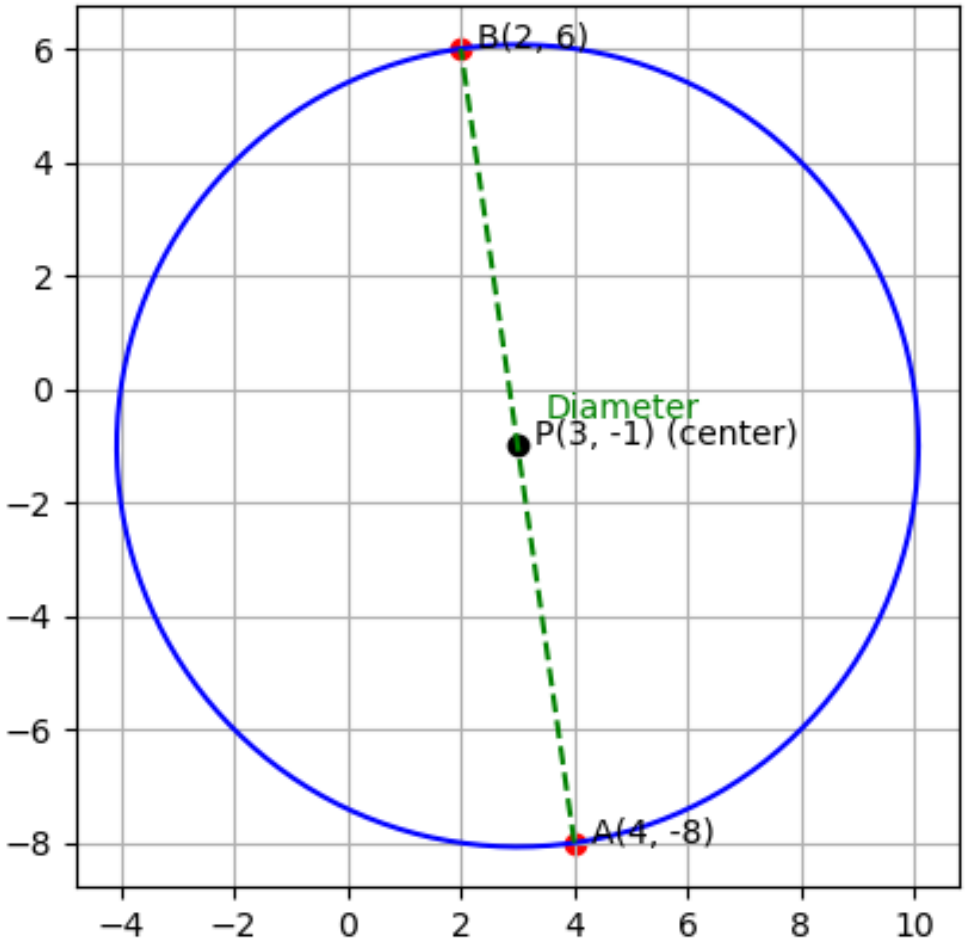
```
# Mark points with coordinates
ax.scatter(*A, color='red')
ax.text(A[0]+0.3, A[1], f"A{tuple(A.astype(int))}")

ax.scatter(*B, color='red')
ax.text(B[0]+0.3, B[1], f"B{tuple(B.astype(int))}")

ax.scatter(*P, color='black')
ax.text(P[0]+0.3, P[1], f"P{tuple(P.astype(int))} (center)")

ax.set_aspect('equal')
ax.grid(True)
plt.show()
```

Plot-Using Both C and Python



Python Code

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

P = np.array([2, 6])
B = np.array([3, -1])
A = 2*P - B

radius = np.linalg.norm(B - P)
theta = np.linspace(0, 2*np.pi, 300)
x_circle = P[0] + radius*np.cos(theta)
y_circle = P[1] + radius*np.sin(theta)
```

Python Code

```
plt.plot(x_circle, y_circle, 'r', label='Circle')
plt.plot([A[0], B[0]], [A[1], B[1]], 'g--', label='Diameter')

plt.scatter([A[0], B[0], P[0]], [A[1], B[1], P[1]], color='blue')
plt.text(A[0], A[1]-0.5, f'A{tuple(A)}')
plt.text(B[0], B[1]+0.5, f'B{tuple(B)}')
plt.text(P[0]+0.2, P[1], f'P{tuple(P)}')
```

Python Code

```
  plt.axhline(0, color='black', linewidth=0.5)
plt.axvline(0, color='black', linewidth=0.5)
plt.grid(True)
plt.gca().set_aspect('equal', adjustable='box')

# Legend moved to top-right corner
plt.legend(loc="upper right")

plt.xlabel("x")
plt.ylabel("y")
plt.title("Circle with Diameter AB and Center P")
plt.savefig("fig1.png")
plt.show()
```

Graph representation:

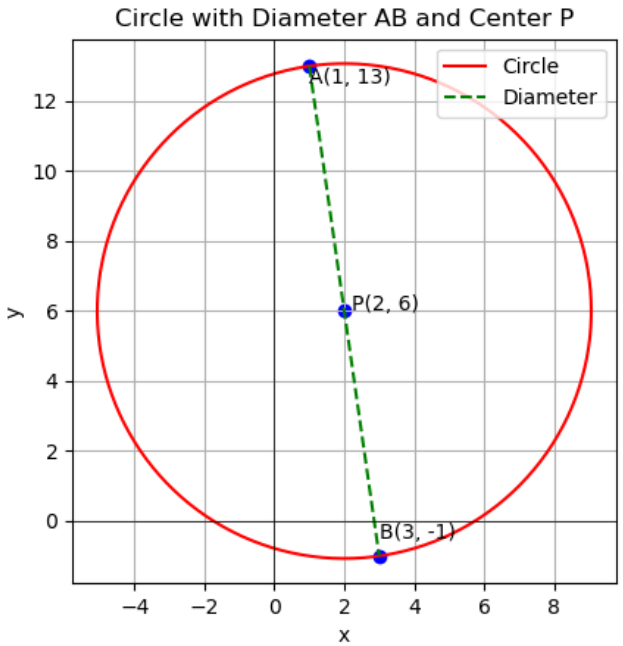


Figure: 0