MATGEO: 7-7.2-19

AI24BTECH11020 - Rishika Kotha

November 5, 2024

Question

Equation of the circle with centre on the Y axis and passing through the origin and the point (2, 3) is

1. $3x^2 + 3y^2 - 13y = 0$
2. $3x^2 + 3y^2 + 13x + 3 = 0$
3. $6x^2 + 6y^2 - 13x = 0$
4. $x^2 + y^2 + 13x + 3 = 0$

(MATGEO 7-7.2-19)

Solution: Theory

parameter	Description	value
C	Centre	$\begin{pmatrix} 0 \\ 13/6 \end{pmatrix}$
O	point1	$\begin{pmatrix} 0 \\ 0 \end{pmatrix}$
P	point2	$\begin{pmatrix} 2 \\ 3 \end{pmatrix}$
r	radius	$13/6$

Given Data

From the given information,

$$x_1 = \begin{pmatrix} 2 \\ 3 \end{pmatrix}, x_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, n = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, c = 0 \tag{1}$$

$$\begin{pmatrix} 4 & 6 & 1 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{pmatrix} \begin{pmatrix} u \\ f \end{pmatrix} = \begin{pmatrix} -13 \\ 0 \\ 0 \end{pmatrix} \tag{2}$$

The augmented matrix is expressed as

$$\left(\begin{array}{ccc|c} 4 & 6 & 1 & -13 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \end{array}\right) \tag{3}$$

Row Operations

performing sequences of row operations to transform into Echelon form

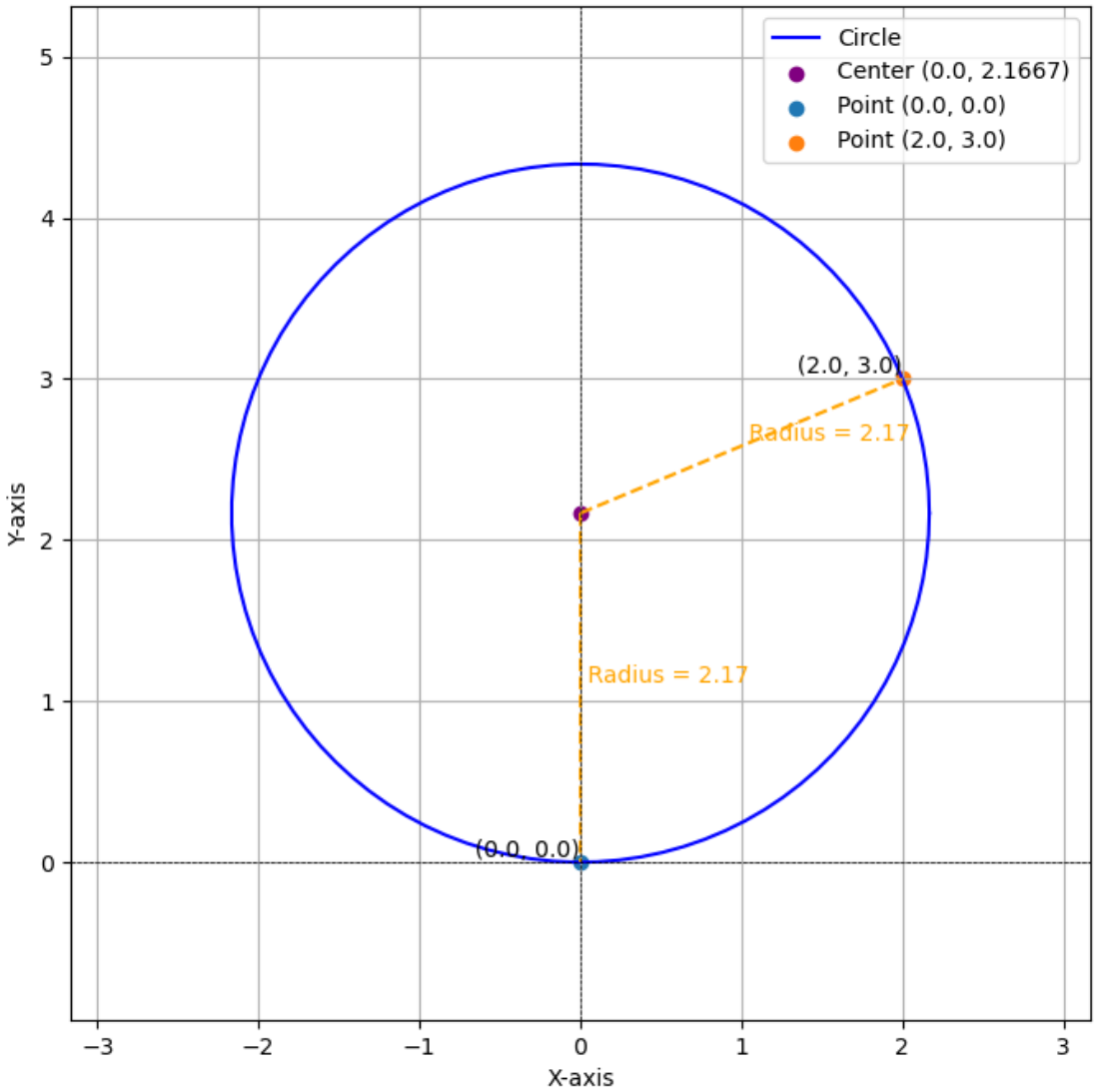
$$\xleftrightarrow{R_3 \to 4R_3 + R_1} \left(\begin{array}{ccc|c} 4 & 6 & 1 & -13 \\ 0 & 0 & 1 & 0 \\ 0 & 6 & 1 & -13 \end{array}\right) \xleftrightarrow{R_1 \to R_1 - R_3} \left(\begin{array}{ccc|c} 4 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 6 & 1 & -13 \end{array}\right) \tag{4}$$

$$\xleftrightarrow{R_2 \to R_2 - R_3} \left(\begin{array}{ccc|c} 4 & 0 & 0 & 0 \\ 0 & -6 & 0 & 13 \\ 0 & 6 & 1 & -13 \end{array}\right) \xleftrightarrow{R_3 \to R_3 + R_2} \left(\begin{array}{ccc|c} 4 & 0 & 0 & 0 \\ 0 & -6 & 0 & 13 \\ 0 & 0 & 1 & 0 \end{array}\right) \tag{5}$$

$$\xleftrightarrow{R_1 \to R_1/4, R_2 \to R_2/-6} \left(\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -13/6 \\ 0 & 0 & 1 & 0 \end{array}\right) \tag{6}$$

$$u = \begin{pmatrix} 0 \\ -13/6 \end{pmatrix}, f = 0 \tag{7}$$

Graph


Circle
Center (0.0, 2.1667)
Point (0.0, 0.0)
Point (2.0, 3.0)
(2.0, 3.0)
Radius = 2.17
Radius = 2.17
(0.0, 0.0)
Y-axis
X-axis
5
4
3
2
1
0
−3
−2
−1
0
1
2
3

C-Code

```c
#include <stdio.h>
int main() {
    float centerX = 0.0f, centerY = 2.1667f;
    float radius = 2.1667f;
    float points[2][2] = { {0.0f, 0.0f}, {2.0f, 3.0f} };
    float *center = &centerX;
    float *radiusPtr = &radius;
    float (*pointsPtr)[2] = points;
    FILE *file = fopen("data.txt", "w");
    if (file == NULL) {
        perror("Error opening file");
        return 1;}
    fprintf(file, "Center: %.4f, %.4f\n", *center, *(center + 1));
    fprintf(file, "Radius: %.4f\n", *radiusPtr);
    fprintf(file, "Points: (%.1f, %.1f), (%.1f, %.1f)\n",
            pointsPtr[0][0], pointsPtr[0][1], pointsPtr[1][0], pointsPtr[1][1]);
    fclose(file);
    printf("Data written to data.txt successfully.\n");
    return 0;
}

```

Python Code

```python
import numpy as np
import matplotlib.pyplot as plt
def read_data(file_name):
    with open(file_name, 'r') as file:
        lines = file.readlines()
    center = tuple(map(float, lines[0].strip().split(':')[1].strip().split(',')))
    radius = float(lines[1].strip().split(':')[1].strip())
    points_line = lines[2].strip().split(':')[1].strip()
    points = [tuple(map(float, p.strip().strip('()').split(','))) for p in points_line.split('),
('')]
    return center, radius, points
center, radius, points = read_data('data.txt')
theta = np.linspace(0, 2 * np.pi, 100)
x_circle = radius * np.cos(theta) + center[0]
y_circle = radius * np.sin(theta) + center[1]
plt.figure(figsize=(8, 8))
plt.plot(x_circle, y_circle, label='Circle', color='blue')
plt.scatter(center[0], center[1], color='purple', label=f'Center {center}')
for point in points:
    plt.scatter(*point, label=f'Point {point}')
for point in points:
    plt.plot([center[0], point[0]], [center[1], point[1]], color='orange', linestyle='--')
    mid_x = (center[0] + point[0]) / 2
    mid_y = (center[1] + point[1]) / 2
    plt.text(mid_x, mid_y, f' Radius = {radius:.2f}', fontsize=10, color='orange',
verticalalignment='bottom')
for point in points:
    plt.text(point[0], point[1], f'  {point}', fontsize=10, verticalalignment='bottom',
horizontalalignment='right')
plt.axis('equal')
plt.xlim(center[0] - radius - 1, center[0] + radius + 1)
plt.ylim(center[1] - radius - 1, center[1] + radius + 1)
plt.axhline(0, color='black', linewidth=0.5, ls='--')
plt.axvline(0, color='black', linewidth=0.5, ls='--')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.legend()
plt.grid()
plt.show()
```