1.3.9

AI25BTECH11030 - SARVESH TAMGADE

September 3, 2025

Question

Find the coordinates of a point on Y axis which is at a distance of $5\sqrt{2}$ from the point $P(3, -2, 5)$.

Solution

Let the required point on Y axis be $\vec{Q} = \begin{pmatrix} 0 \\ y \\ 0 \end{pmatrix}$.

So,

$$\vec{P} - \vec{Q} = \begin{pmatrix} 3 \\ -2 \\ 5 \end{pmatrix} - \begin{pmatrix} 0 \\ y \\ 0 \end{pmatrix} = \begin{pmatrix} 3 \\ -2 - y \\ 5 \end{pmatrix}$$

The desired distance is:

$$d = \|\vec{P} - \vec{Q}\| = 5\sqrt{2}$$

So,

$$(\vec{P} - \vec{Q})^T(\vec{P} - \vec{Q}) = 3^2 + (-2 - y)^2 + 5^2 = 9 + (y + 2)^2 + 25$$

$$9 + (y + 2)^2 + 25 = 50$$

$$(y + 2)^2 = 16$$

$$y + 2 = \pm 4$$

Thus, $y = 2$ or $y = -6$

Answer

The required coordinates are:

$$\vec{Q_1} = \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix} \qquad \text{and} \qquad \vec{Q_2} = \begin{pmatrix} 0 \\ -6 \\ 0 \end{pmatrix}$$

Graph

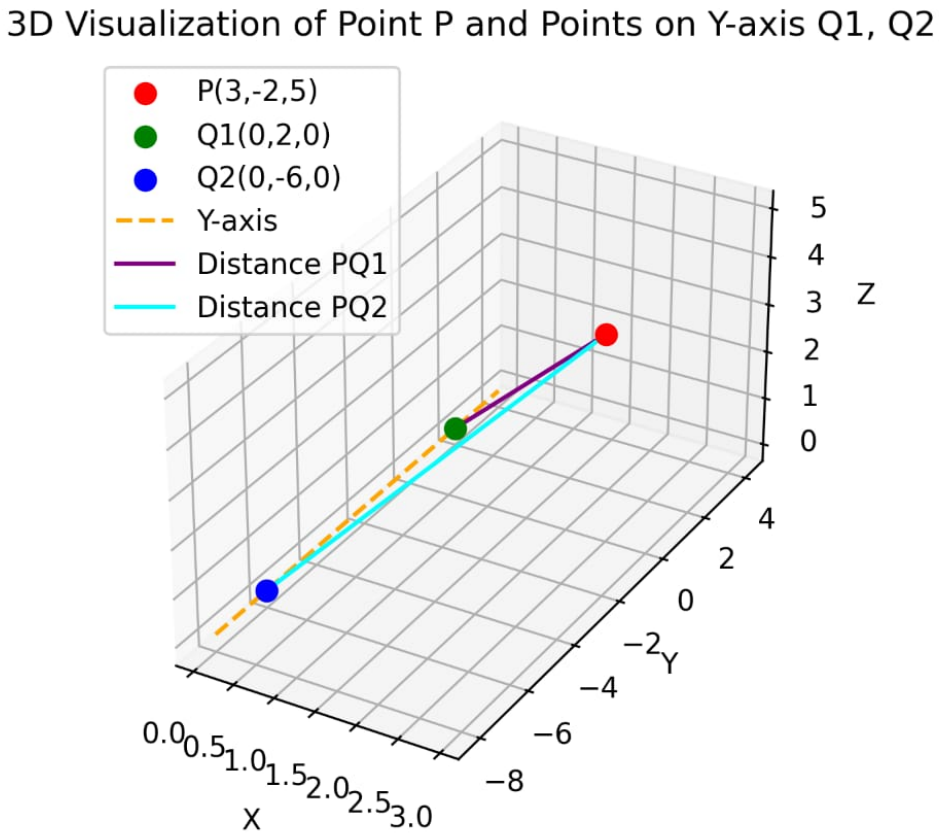

Figure: 3D Visualization of Point P and Points on Y-axis Q1,Q2

C Code

```c
#include <stdio.h>
#include <math.h>
#include "matfun.h"

int main() {
    double P[3] = {3.0, -2.0, 5.0};
    double distance = 5.0 * sqrt(2.0);

    double roots[2];
    solve_y_coordinate(P, distance, roots);

    if (isnan(roots[0]) || isnan(roots[1])) {
        printf("No real solutions exist for the given distance.\n");
    } else {
        printf("The points on the Y-axis at distance %.2f from P(3, -2,
            5) are:\n", distance);
        printf("Q1 = (0, %.2f, 0)\n", roots[0]);
        printf("Q2 = (0, %.2f, 0)\n", roots[1]);
    }
    return 0;
}
```

Python Plot

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# Points
P = np.array([3, -2, 5])
Q1 = np.array([0, 2, 0])
Q2 = np.array([0, -6, 0])

# Plot
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

# Plot points
ax.scatter(*P, color='red', label='P(3,-2,5)', s=50)
ax.scatter(*Q1, color='green', label='Q1(0,2,0)', s=50)
ax.scatter(*Q2, color='blue', label='Q2(0,-6,0)', s=50)
```

Python Plot

```
# Plot Y-axis line for reference
y_axis = np.array([[0, y, 0] for y in np.linspace(-8, 4, 100)])
ax.plot(y_axis[:,0], y_axis[:,1], y_axis[:,2], color='orange', linestyle
    ='--', label='Y-axis')

# Lines from P to Q1 and Q2
ax.plot([P[0], Q1[0]], [P[1], Q1[1]], [P[2], Q1[2]], color='purple',
    linestyle='-', label='Distance PQ1')
ax.plot([P[0], Q2[0]], [P[1], Q2[1]], [P[2], Q2[2]], color='cyan',
    linestyle='-', label='Distance PQ2')

# Labels and legend
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')
ax.legend()
```

Python Plot

```
# Set aspect ratio equal for better visualization
ax.set_box_aspect([1,2,1])

plt.title('3D Visualization of Point P and Points on Y-axis Q1, Q2')
plt.savefig('3d_points_plot.png', dpi=300)

plt.show()
```