1.9.2

Bhargav - EE25BTECH11013

August 27,2025

Question

The point on the X axis which is equidistant from $(-4, 0)$ and $(10, 0)$ is

Theoretical Solution

Let the 2 points be **A** and **B** and let the desired point equidistant from both **A** and **B** be **O**:

$$\mathbf{A} = \begin{pmatrix} -4 \\ 0 \end{pmatrix} , \mathbf{B} = \begin{pmatrix} 10 \\ 0 \end{pmatrix} \tag{1}$$

$$\mathbf{O} = x\,\mathbf{e}_1 = \begin{pmatrix} x \\ 0 \end{pmatrix} \tag{2}$$

$$\tag{3}$$

If **O** lies on X axis and is equidistant from **A** and **B**

Theoretical Solution

$$\|\mathbf{O}-\mathbf{A}\| = \|\mathbf{O}-\mathbf{B}\| \tag{4}$$

$$\implies \|\mathbf{O}-\mathbf{A}\|^2 = \|\mathbf{O}-\mathbf{B}\|^2 \tag{5}$$

$$\implies (\mathbf{O}-\mathbf{A})^\top(\mathbf{O}-\mathbf{A}) = (\mathbf{O}-\mathbf{B})^\top(\mathbf{O}-\mathbf{B}) \tag{6}$$

$$\implies \mathbf{O}^\top\mathbf{O} - 2\mathbf{O}^\top\mathbf{A} + \mathbf{A}^\top\mathbf{A} = \mathbf{O}^\top\mathbf{O} - 2\mathbf{O}^\top\mathbf{B} + \mathbf{B}^\top\mathbf{B} \tag{7}$$

Theoretical Solution

$$\implies \|\mathbf{O}\|^2 - 2\mathbf{O}^\top\mathbf{A} + \|\mathbf{A}\|^2 = \|\mathbf{O}\|^2 - 2\mathbf{O}^\top\mathbf{B} + \|\mathbf{B}\|^2 \tag{8}$$

$$(\mathbf{A}-\mathbf{B})^\top\mathbf{O} = \frac{\|\mathbf{A}\|^2 - \|\mathbf{B}\|^2}{2}. \tag{9}$$

$$\mathbf{O} = x\mathbf{e}_1, \tag{10}$$

$$x = \frac{\|\mathbf{A}\|^2 - \|\mathbf{B}\|^2}{2(\mathbf{A}-\mathbf{B})^\top\mathbf{e}_1}. \tag{11}$$

Solving for x, we get $x = 3$

$$\therefore \mathbf{O} = \begin{pmatrix} 3 \\ 0 \end{pmatrix} \tag{12}$$

C Code - Equidistant point

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

// Function to compute x-coordinate of equidistant point
double equidistant_point(double ax, double ay, double bx, double
    by) {
    // Norm squared of A and B
    double normA2 = ax*ax + ay*ay;
    double normB2 = bx*bx + by*by;

    double denom = 2 * (ax - bx);


    double x = (normA2 - normB2) / denom;
    return x;
}
```

Python + C Code

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

# Load the C shared library
lib = ctypes.CDLL(./libequidistant.so)

# Define function prototype: double f(double, double, double,
    double)
lib.equidistant_point.argtypes = [ctypes.c_double, ctypes.
    c_double,
                                  ctypes.c_double, ctypes.c_double]
lib.equidistant_point.restype = ctypes.c_double

# Define A and B
A = np.array([-4.0, 0.0])
B = np.array([10.0, 0.0])
```

Python + C Code

```
# Call C function
x = lib.equidistant_point(A[0], A[1], B[0], B[1])

# Equidistant point
O = np.array([x, 0.0])

print(Equidistant point O =, O)

# ---- Plotting ----
plt.figure(figsize=(6,6))
plt.axhline(0, color='gray', linewidth=0.8)
plt.axvline(0, color='gray', linewidth=0.8)

# Plot points
plt.scatter(A[0], A[1], color='red', label='A (-4,0)')
plt.scatter(B[0], B[1], color='blue', label='B (10,0)')
plt.scatter(O[0], O[1], color='green', marker='*', s=150, label=f
    'O ({int(O[0])},0)')
```

Python + C Code

```
# Connect O to A and B
plt.plot([A[0], O[0]], [A[1], O[1]], 'r--')
plt.plot([B[0], O[0]], [B[1], O[1]], 'b--')

plt.legend(loc=upper right)
plt.grid(True, linestyle='--', alpha=0.6)
plt.title(Equidistant Point on X-axis (C + Python))
plt.xlabel(x)
plt.ylabel(y)
plt.axis(equal)
plt.savefig(/Users/bhargavkrish/Documents/ee1030-2025/
    ee25btech11013/matgeo/1.9.2/figs/Figure_1.png)
plt.show()
```

Python Code

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


A = np.array([-4, 0])
B = np.array([10, 0])


e1 = np.array([1, 0])


num = np.linalg.norm(A)**2 - np.linalg.norm(B)**2
den = 2 * np.dot(A - B, e1)
x = num / den


O = x * e1
```

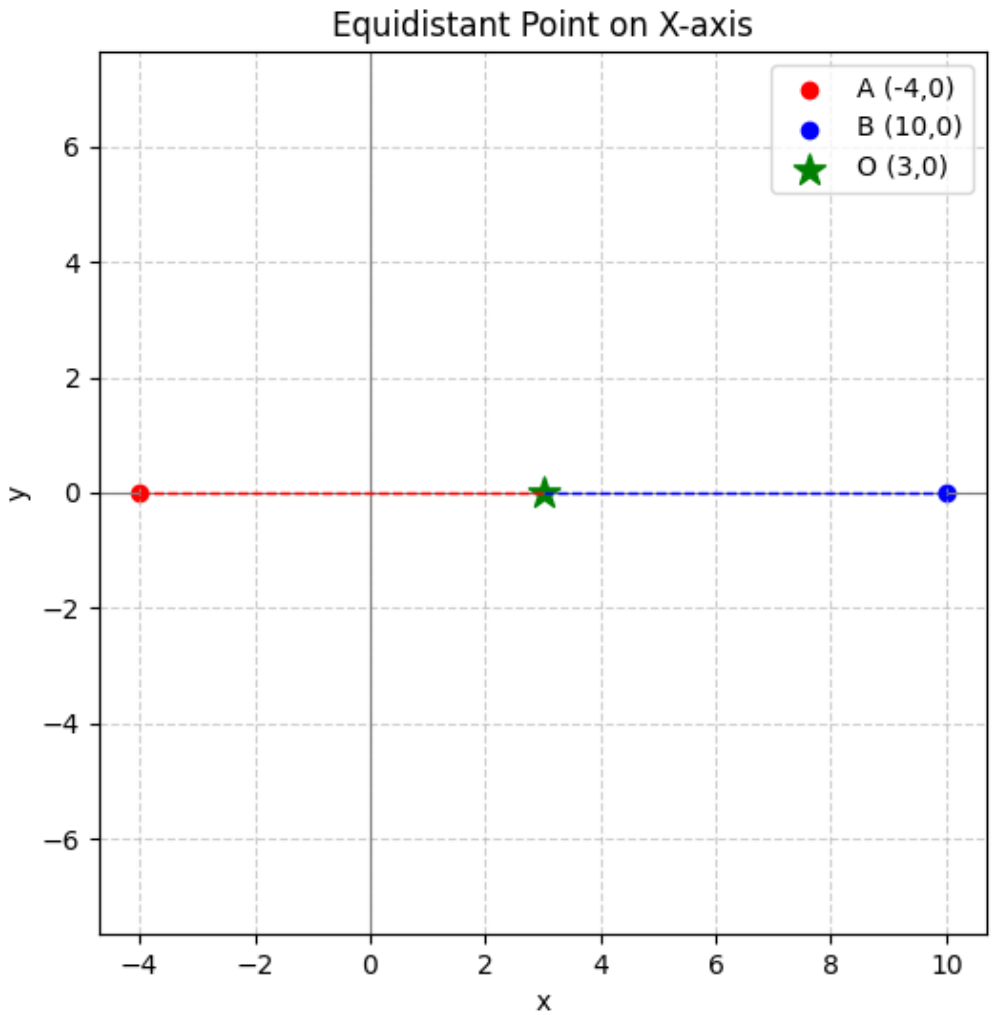
Python Code

```
print(Point equidistant from A and B on x-axis:, O)


plt.figure(figsize=(6,6))
plt.axhline(0, color='gray', linewidth=0.8) # x-axis
plt.axvline(0, color='gray', linewidth=0.8) # y-axis

# Plot points
plt.scatter(A[0], A[1], color='red', label='A (-4,0)')
plt.scatter(B[0], B[1], color='blue', label='B (10,0)')
plt.scatter(O[0], O[1], color='green', marker='*', s=150, label=f
    'O ({int(O[0])},0)')


plt.plot([A[0], O[0]], [A[1], O[1]], 'r--', linewidth=1)
plt.plot([B[0], O[0]], [B[1], O[1]], 'b--', linewidth=1)
```

Python Code

```
plt.legend(loc=upper right)
plt.grid(True, linestyle='--', alpha=0.6)
plt.title(Equidistant Point on X-axis)
plt.xlabel(x)
plt.ylabel(y)
plt.axis(equal)
plt.savefig(/Users/bhargavkrish/Documents/ee1030-2025/
    ee25btech11013/matgeo/1.9.2/figs/Figure_1.png)
plt.show()
```

Plot


Equidistant Point on X-axis
A (-4,0)
B (10,0)
O (3,0)
6
4
2
0
−2
−4
−6
y
−4
−2
0
2
4
6
8
10
x