1.5.15

INDHIRESH S - EE25BTECH11027

Agust, 2025

Question

The midpoint of the line segment joining $A(2a, 4)$ and $B(-2, 3b)$ is $(1, 2a + 1)$. Findthe values of a and b.

Equation

$$\mathbf{A} = \begin{pmatrix} 2a \\ 4 \end{pmatrix}, \mathbf{B} = \begin{pmatrix} -2 \\ 3b \end{pmatrix} \tag{1}$$

Let the midpoint of points A and B be C. where,

$$\mathbf{C} = \begin{pmatrix} 1 \\ 2a+1 \end{pmatrix} \tag{2}$$

Theoretical Solution

We know that the midpoint formula for the points A and B is

$$\mathbf{C_x} = \frac{\mathbf{A_x} + \mathbf{B_x}}{2} \tag{3}$$

Where C_X, A_X *and* B_X are x coordinates of point C,A and B

And also A,B and C lies in the same line so they are collinear. So,

$$rank(C - A \quad B - A) = 1 \tag{4}$$

$$rank \begin{pmatrix} 1-2a & -2-2a \\ 2a-3 & 3b-4 \end{pmatrix} = 1 \tag{5}$$

Theoretical Solution

From eq.3:

$$\mathbf{C_x} = \frac{2a - 2}{2} \tag{6}$$

$$1 = \frac{2a - 2}{2} \tag{7}$$

$$1 = a - 1 \tag{8}$$

$$a = 2 \tag{9}$$

Theoretical Solution

Now substituiting the value of a in Eq.5, we get:

$$rank \begin{pmatrix} 1-2(2) & -2-2(2) \\ 2(2)-3 & 3b-4 \end{pmatrix} = 1 \tag{10}$$

$$rank \begin{pmatrix} -3 & -6 \\ 1 & 3b-4 \end{pmatrix} = 1 \tag{11}$$

By applying row operation for the matrix
$R_2 \longrightarrow 3R_2 + R_1$
We get

$$(C-A \quad B-A) = \begin{pmatrix} -3 & -6 \\ 0 & 9b-18 \end{pmatrix} \tag{12}$$

For the rank to be 1, the second row must be a zero vector. Therefore:

$$9b - 18 = 0 \tag{13}$$

$$9b = 18 \tag{14}$$

Theoretical Solution

$$b = 2 \tag{17}$$

Therefore the final values of a and b are:

$$a = 2 \; and \; b = 2 \tag{18}$$

C Code - Midpoint formula

```c
#include <stdio.h>

// Function to calculate midpoint
void midpoint(float x1, float y1, float x2, float y2, float *mx,
    float *my) {
    *mx = (x1 + x2) / 2.0;
    *my = (y1 + y2) / 2.0;
}
```

Python Code

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

# Load the shared library
lib = ctypes.CDLL(./midpoint.so) # use midpoint.dll on Windows

# Define function signature
lib.midpoint.argtypes = [
    ctypes.c_float, ctypes.c_float, # x1, y1
    ctypes.c_float, ctypes.c_float, # x2, y2
    ctypes.POINTER(ctypes.c_float), # mx
    ctypes.POINTER(ctypes.c_float) # my
]
```

Python Code

```
# Given values from problem
a, b = 2, 2
A = (2*a, 4) # (4,4)
B = (-2, 3*b) # (-2,6)

# Prepare variables to hold midpoint
mx, my = ctypes.c_float(), ctypes.c_float()

# Call the C function
lib.midpoint(A[0], A[1], B[0], B[1], ctypes.byref(mx), ctypes.
    byref(my))
M = (mx.value, my.value)

print(fMidpoint from C: {M})
```

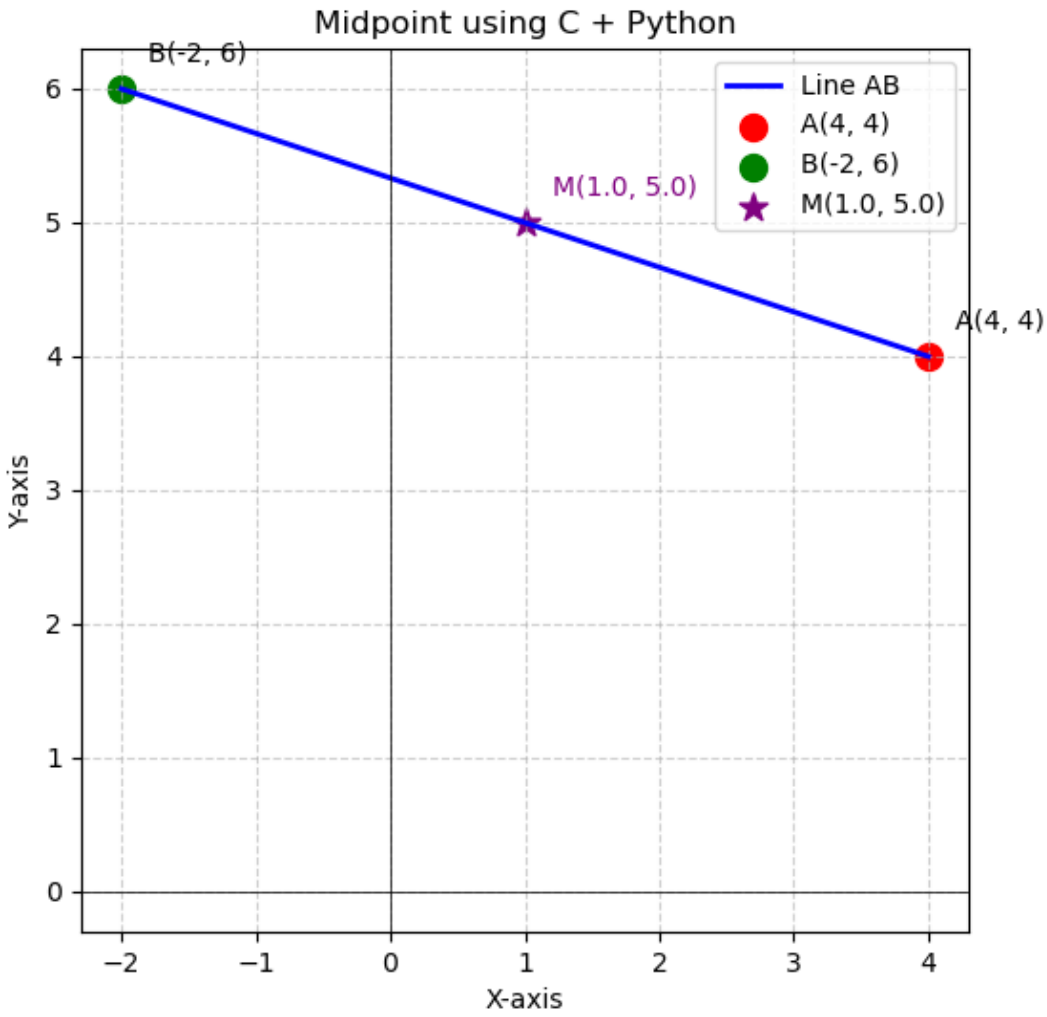
Python Code

```
# --- Plot ---
plt.figure(figsize=(6,6))
plt.plot([A[0], B[0]], [A[1], B[1]], 'b-', linewidth=2, label='
    Line AB')

# Scatter points
plt.scatter(*A, color='red', s=100, label=fA{A})
plt.scatter(*B, color='green', s=100, label=fB{B})
plt.scatter(*M, color='purple', s=120, marker='*', label=fM{M})

# Annotate
plt.text(A[0]+0.2, A[1]+0.2, fA{A}, fontsize=10)
plt.text(B[0]+0.2, B[1]+0.2, fB{B}, fontsize=10)
plt.text(M[0]+0.2, M[1]+0.2, fM{M}, fontsize=10, color=purple)
```

Python Code

```
# Axes formatting
plt.axhline(0, color='black', linewidth=0.5)
plt.axvline(0, color='black', linewidth=0.5)
plt.grid(True, linestyle=--, alpha=0.6)
plt.legend()
plt.title(Midpoint using C + Python)
plt.xlabel(X-axis)
plt.ylabel(Y-axis)
plt.show()
```

Plot


Midpoint using C + Python
B(-2, 6)
M(1.0, 5.0)
A(4, 4)
Line AB
A(4, 4)
B(-2, 6)
M(1.0, 5.0)
Y-axis
X-axis
6
5
4
3
2
1
0
−2
−1
0
1
2
3
4