12.26

BEERAM MADHURI - EE25BTECH11012

October 2025

Question

Phani starts from point **P**, goes North for 3 km, and then East for 4 km to reach point **Q**. She then turns to face point **P** and goes 15 km in that direction. She then goes North for 6 km. How far is she from point **P**, and in which direction should she go to reach point **P**?

a) 8 km, East

b) 12 km, North

c) 6 km, East

d) 10 km, North

finding the final position and direction of Phani:

Let point **P** be the origin:

$$\mathbf{P} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{1}$$

Moving from **P** to **Q**
First, move North by 3 km:

$$\mathbf{A} = \begin{bmatrix} 0 \\ 3 \end{bmatrix} \tag{2}$$

Then, move East by 4 km:

$$\mathbf{B} = \begin{bmatrix} 4 \\ 0 \end{bmatrix} \tag{3}$$

Position at point $\mathbf{Q}$ is:

$$\mathbf{Q} = \mathbf{P} + \mathbf{A} + \mathbf{B} \tag{4}$$

$$= \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 3 \end{bmatrix} + \begin{bmatrix} 4 \\ 0 \end{bmatrix} \tag{5}$$

$$= \begin{bmatrix} 4 \\ 3 \end{bmatrix} \tag{6}$$

Move 15 km Toward **P** from **Q**
Direction vector from **Q** to **P**:

$$\mathbf{D} = \mathbf{P} - \mathbf{Q} \tag{7}$$

$$= \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 4 \\ 3 \end{bmatrix} \tag{8}$$

$$= \begin{bmatrix} -4 \\ -3 \end{bmatrix} \tag{9}$$

$$|\mathbf{D}| = \sqrt{(-4)^2 + (-3)^2} \tag{10}$$

$$= \sqrt{16 + 9} \tag{11}$$

$$= \sqrt{25} = 5 \tag{12}$$

$$\hat{\mathbf{D}} = \frac{1}{5}\begin{bmatrix} -4 \\ -3 \end{bmatrix} \tag{13}$$

Now multiply by 15 km:

$$\mathbf{C} = 15 \cdot \hat{\mathbf{D}} \tag{14}$$

$$= 3 \cdot \begin{bmatrix} -4 \\ -3 \end{bmatrix} \tag{15}$$

$$= \begin{bmatrix} -12 \\ -9 \end{bmatrix} \tag{16}$$

New position :

$$\mathbf{R} = \mathbf{Q} + \mathbf{C} \tag{17}$$

$$= \begin{bmatrix} 4 \\ 3 \end{bmatrix} + \begin{bmatrix} -12 \\ -9 \end{bmatrix} \tag{18}$$

$$= \begin{bmatrix} -8 \\ -6 \end{bmatrix} \tag{19}$$

Moving North by 6 km

$$\mathbf{F} = \begin{bmatrix} 0 \\ 6 \end{bmatrix} \tag{20}$$

Final position:

$$\mathbf{S} = \mathbf{R} + \mathbf{F} \tag{21}$$

$$= \begin{bmatrix} -8 \\ -6 \end{bmatrix} + \begin{bmatrix} 0 \\ 6 \end{bmatrix} \tag{22}$$

$$= \begin{bmatrix} -8 \\ 0 \end{bmatrix} \tag{23}$$

Distance and Direction from Final Position to $\mathbf{P}$

$$\mathbf{P} - \mathbf{S} \tag{24}$$

$$= \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} -8 \\ 0 \end{bmatrix} \tag{25}$$

$$= \begin{bmatrix} 8 \\ 0 \end{bmatrix} \tag{26}$$

Distance:

$$\|\mathbf{P} - \mathbf{S}\| = \sqrt{8^2 + 0^2} = 8 \text{ km} \tag{27}$$

Direction: Since the vector is along the positive x-axis, the direction is **East**.

$\therefore$ Option a is correct

Python Code

```
import matplotlib.pyplot as plt
# Define the points in the journey
p = (0, 0)
north_point = (0, 3)
q = (4, 3)
r = (-8, -6)
s = (-8, 0)
# Create the plot
plt.figure(figsize=(10, 8))
ax = plt.gca()
```

Python Code

```
# Plot the path segments
# 1. P to North Point
plt.plot([p[0], north_point[0]], [p[1], north_point[1]], 'r-o',
    label='Path 1: 3 km North')
plt.annotate('3 km', xy=(0.1, 1.5), xytext=(0.1, 1.5))
# 2. North Point to Q
plt.plot([north_point[0], q[0]], [north_point[1], q[1]], 'g-o',
    label='Path 2: 4 km East')
plt.annotate('4 km', xy=(2, 3.1), xytext=(2, 3.1))
```

Python Code

```
# 3. Q to R (towards P and beyond)
plt.plot([q[0], r[0]], [q[1], r[1]], 'b-o', label='Path 3: 15 km
    towards P')
# Add an arrow for this segment
ax.arrow(q[0], q[1], r[0]-q[0], r[1]-q[1], head_width=0.5,
    head_length=0.7, fc='blue', ec='blue', length_includes_head=
    True)
plt.annotate('15 km', xy=(-3, -2), xytext=(-3, -2), color='blue')
```

Python Code

```
# 4. R to S
plt.plot([r[0], s[0]], [r[1], s[1]], 'm-o', label='Path 4: 6 km
    North')
plt.annotate('6 km', xy=(-8.5, -3), xytext=(-8.5, -3), color='
    purple', rotation=90)

# 5. Final path from S to P
plt.plot([s[0], p[0]], [s[1], p[1]], 'k--o', label='Final: 8 km
    East to P')
```

Python Code

```
ax.arrow(s[0], s[1], p[0]-s[0], p[1]-s[1], head_width=0.5,
    head_length=0.7, fc='black', ec='black', length_includes_head
    =True)
plt.annotate('8 km', xy=(-4, 0.2), xytext=(-4, 0.2), color='black
    ')
plt.text(p[0] - 0.5, p[1] - 0.5, 'P (Start)', fontsize=12)
plt.text(q[0] + 0.2, q[1] + 0.2, 'Q', fontsize=12)
plt.text(r[0] - 0.5, r[1] - 1, 'R', fontsize=12)
plt.text(s[0] - 1.5, s[1] + 0.2, 'S (Final)', fontsize=12)
```

Python Code

```
# Set plot limits and labels
plt.xlim(-15, 10)
plt.ylim(-10, 10)
plt.xlabel("East-West Direction (km)")
plt.ylabel("North-South Direction (km)")
plt.title("Phani's Journey")
plt.axhline(0, color='grey', lw=0.5)
plt.axvline(0, color='grey', lw=0.5)
plt.grid(True)
plt.gca().set_aspect('equal', adjustable='box')
plt.legend()
```

Python Code

```
# Add compass directions
plt.text(0, 9, 'North', ha='center', va='center', fontsize=10)
plt.text(0, -9, 'South', ha='center', va='center', fontsize=10)
plt.text(9, 0, 'East', ha='center', va='center', fontsize=10)
plt.text(-14, 0, 'West', ha='center', va='center', fontsize=10)

# Save the figure
plt.savefig("phanis_journey.png")
```

C Code

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

// A struct to represent a point with x and y coordinates
typedef struct {
    double x;
    double y;
} Point;
```

C Code

```c
int main() {
    // Let the starting point P be the origin
    Point p = {0.0, 0.0};
    Point current_pos = {0.0, 0.0};

    printf("Simulating Phani's journey...\n");
    printf("Start at P: (%.1f, %.1f)\n", current_pos.x,
        current_pos.y);
```

C Code

```c
// 1. Goes North for 3 km
current_pos.y += 3.0;
printf("1. After moving North 3 km: (%.1f, %.1f)\n",
    current_pos.x, current_pos.y);

// 2. Then East for 4 km to reach point Q
current_pos.x += 4.0;
Point q = current_pos;
printf("2. After moving East 4 km to Q: (%.1f, %.1f)\n",
    current_pos.x, current_pos.y);
```

C Code

```
    // 3. Turns to face point P and goes 15 km
    // Vector from current position (Q) towards P
    double vec_to_p_x = p.x - q.x; // 0 - 4 = -4
    double vec_to_p_y = p.y - q.y; // 0 - 3 = -3

    // The distance between Q and P (magnitude of the vector)
    double dist_qp = sqrt(vec_to_p_x * vec_to_p_x + vec_to_p_y *
        vec_to_p_y);
```

C Code

```c
// Create a unit vector for the direction
double unit_vec_x = vec_to_p_x / dist_qp;
double unit_vec_y = vec_to_p_y / dist_qp;

// Move 15 km along this direction
current_pos.x += 15.0 * unit_vec_x;
current_pos.y += 15.0 * unit_vec_y;
printf("3. After moving 15 km towards P: (%.1f, %.1f)\n",
    current_pos.x, current_pos.y);
```

C Code

```
    // 4. Then goes North for 6 km
    current_pos.y += 6.0;
    printf("4. After moving North 6 km (Final Position): (%.1f,
        %.1f)\n\n", current_pos.x, current_pos.y);

    // Question 1: How far is she from point P?
    double final_distance = sqrt(pow(current_pos.x - p.x, 2) +
        pow(current_pos.y - p.y, 2));
```

C Code

```c
// Question 2: In which direction should she go to reach
    point P?
// Vector from her final position back to P
double dir_to_p_x = p.x - current_pos.x; // 0 - (-8) = 8

printf("Final Answer:\n");
printf("Distance from start point P: %.0f km\n",
    final_distance);
printf("Direction to reach point P: ");
```

C Code

```
    if (dir_to_p_x > 0) {
        printf("East\n");
    } else if (dir_to_p_x < 0) {
        printf("West\n");
    } // simplified for this problem

    printf("This corresponds to option a) 8 km, East.\n");
    return 0;
}
```

Python and C Code

```
import ctypes
import math

# Define the Point struct using ctypes
class Point(ctypes.Structure):
    _fields_ = [("x", ctypes.c_double),
                ("y", ctypes.c_double)]
```

Python and C Code

```
def main():
    # Let the starting point P be the origin
    p = Point(0.0, 0.0)
    current_pos = Point(0.0, 0.0)

    print("Simulating Phani's journey...")
    print(f"Start at P: ({current_pos.x:.1f}, {current_pos.y:.1f
        })")
```

Python and C Code

```
    # 1. Goes North for 3 km
    current_pos.y += 3.0
    print(f"1. After moving North 3 km: ({current_pos.x:.1f}, {
        current_pos.y:.1f})")
    # 2. Then East for 4 km to reach point Q
    current_pos.x += 4.0
    q = Point(current_pos.x, current_pos.y)
    print(f"2. After moving East 4 km to Q: ({current_pos.x:.1f},
         {current_pos.y:.1f})")
```

Python and C Code

```
    # 3. Turns to face point P and goes 15 km
    vec_to_p_x = p.x - q.x # -4.0
    vec_to_p_y = p.y - q.y # -3.0
    dist_qp = math.sqrt(vec_to_p_x**2 + vec_to_p_y**2)
    unit_vec_x = vec_to_p_x / dist_qp
    unit_vec_y = vec_to_p_y / dist_qp
```

Python and C Code

```
    current_pos.x += 15.0 * unit_vec_x
    current_pos.y += 15.0 * unit_vec_y
    print(f"3. After moving 15 km towards P: ({current_pos.x:.1f
        }, {current_pos.y:.1f})")
    # 4. Then goes North for 6 km
    current_pos.y += 6.0
    print(f"4. After moving North 6 km (Final Position): ({
        current_pos.x:.1f}, {current_pos.y:.1f})\n")
```

Python and C Code

```
    # Question 1: Distance from point P
    final_distance = math.sqrt((current_pos.x - p.x)**2 + (
        current_pos.y - p.y)**2)
    # Question 2: Direction to reach point P
    dir_to_p_x = p.x - current_pos.x
    print("Final Answer:")
    print(f"Distance from start point P: {final_distance:.0f} km"
        )
```

Python and C Code

```
    if dir_to_p_x > 0:
        print("Direction to reach point P: East")
    elif dir_to_p_x < 0:
        print("Direction to reach point P: West")
    else:
        print("Direction to reach point P: Same line")
    print("This corresponds to option a) 8 km, East.")
if __name__ == "__main__":
    main()
```

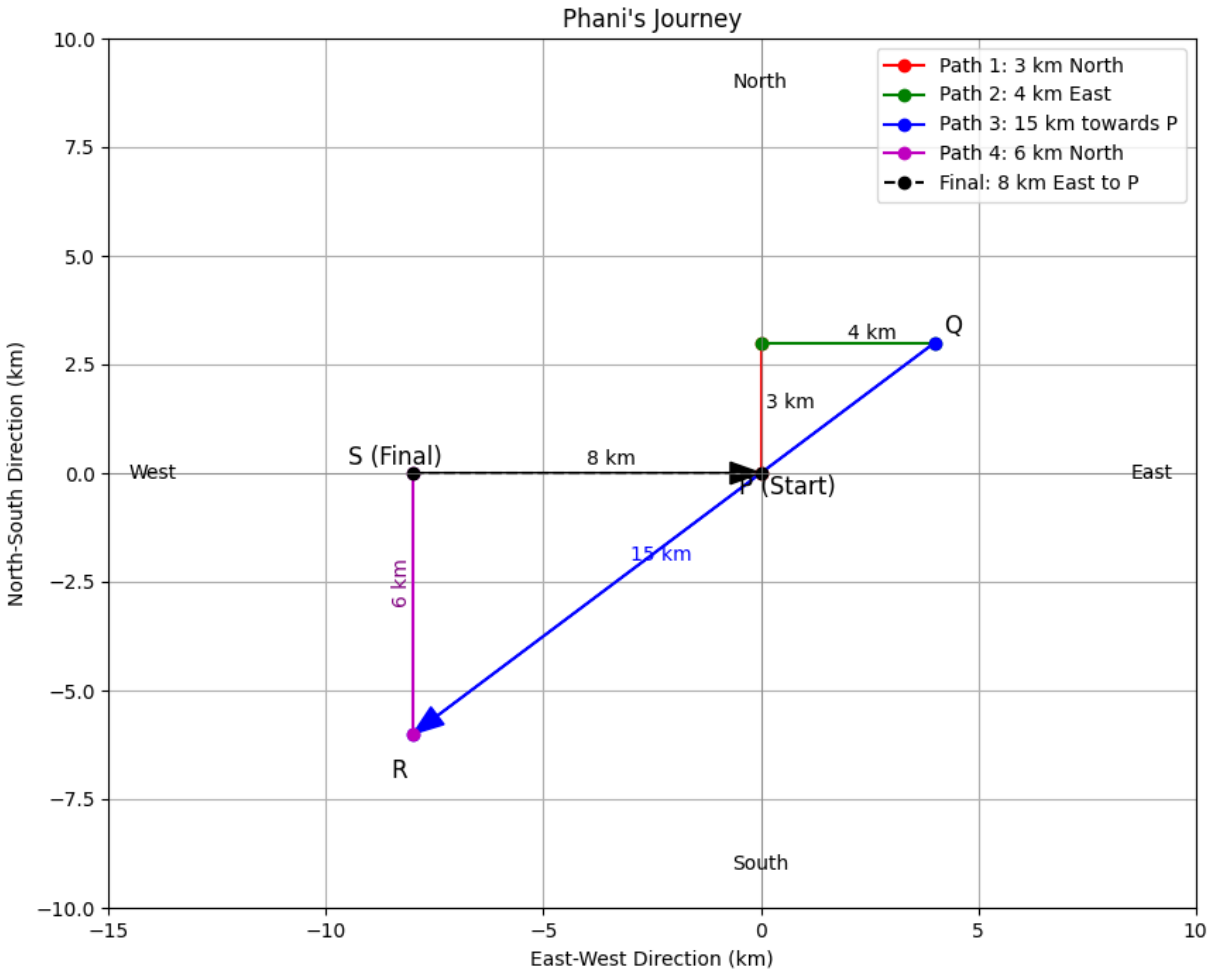
Phani's Journey
Path 1: 3 km North
Path 2: 4 km East
Path 3: 15 km towards P
Path 4: 6 km North
Final: 8 km East to P
North
South
West
East
Q
4 km
3 km
S (Final)
8 km
P (Start)
15 km
6 km
R
North-South Direction (km)
East-West Direction (km)
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−15
−10
−5
0
5
10

Figure: Plot