3.2.19

AI25BTECH11003 - Bhavesh Gaikwad

September 2,2025

Question

Two sides of a triangle are of lengths 5cm and 1.5cm. The length of the third side of the triangle cannot be
a) 3.6 cm
b) 4.1 cm
c) 3.8 cm
d) 3.4 cm

Theoretical Solution

Let the vector along side AB be $\mathbf{a}$
Let the vector along side BC be $\mathbf{b}$
Let the vector along side AC be $\mathbf{c}$
Let the angle between $\mathbf{a}$ *and* $\mathbf{b}$ be θ.

Given:

$$\|\mathbf{a}\| = 5, \ \|\mathbf{b}\| = 1.5 \tag{1}$$

By Triangle Law of Vector Addition,

$$\mathbf{a} + \mathbf{b} = \mathbf{c} \tag{2}$$

$$\mathbf{c}^T\mathbf{c} = (\mathbf{a} + \mathbf{b})^T(\mathbf{a} + \mathbf{b}) \tag{3}$$

Theoretical Solution

$$\mathbf{c}^T\mathbf{c} = (\mathbf{a}^T\mathbf{a}) + (\mathbf{b}^T\mathbf{b}) + (\mathbf{a}^T\mathbf{b}) + (\mathbf{b}^T\mathbf{a}) \tag{4}$$

We know that,

$$\mathbf{a}^T\mathbf{a} = \|\mathbf{a}\|^2 = 25,\ \mathbf{b}^T\mathbf{b} = \|\mathbf{b}\|^2 = 2.25,\ \mathbf{c}^T\mathbf{c} = \|\mathbf{c}\|^2, \tag{5}$$

$$\mathbf{a}^T\mathbf{b} = \mathbf{b}^T\mathbf{a} = \|\mathbf{a}\|\,\|\mathbf{b}\|\cos(\theta) \tag{6}$$

From Equation 4, 5 and 6,

$$\|\mathbf{c}\|^2 = 27.25 + 15\cos(\theta) \tag{7}$$

Since 'θ' is the angle between two vectors, Therefore

$$\theta \,\epsilon\, (0, \pi) \tag{8}$$

The maximum value of $\|\mathbf{c}\|^2$ will occur when $\cos(\theta) = 1$ OR $\theta = 0$

$$\text{Therefore the maximum value of } \|\mathbf{c}\|^2 \ \textit{is}\, 42.25. \tag{9}$$

$$\Rightarrow \text{The maximum value of } \|\mathbf{c}\| \ \textit{is}\, 6.5. \tag{10}$$

Theoretical Solution

The minimum value of $\|\mathbf{c}\|^2$ will occur when $\cos(\theta) = -1$ OR $\theta = \pi$

$$\text{Therefore the minimum value of } \|\mathbf{c}\|^2 \; is\, 12.25. \tag{11}$$

$$\Rightarrow \text{ The minimum value of } \|\mathbf{c}\| \; is\, 3.5. \tag{12}$$

$$\therefore \text{ The Range of } \|\mathbf{c}\| \text{ for triangle to exist: } \|\mathbf{c}\| \, \epsilon \, (3.5, 6.5). \tag{13}$$

From option D of the Question, $\|\mathbf{c}\| = 3.4$ cm
But by Equation 12, $\|\mathbf{c}\| \neq 3.4$, Since $\|\mathbf{c}\| > 3.5$

$$\boxed{\text{Option D of the Question is Incorrect and } \|\mathbf{c}\| \neq 3.4\, cm} \tag{14}$$

C Code

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

int main() {
    // Given values
    double magnitude_a = 5.0;
    double magnitude_b = 1.5;
    double theta = M_PI / 6; // 30 degrees in radians (you can
        change this)

    // Vector a (let's place it along x-axis for simplicity)
    double a_x = magnitude_a;
    double a_y = 0.0;

    // Vector b (at angle theta from vector a)
    double b_x = magnitude_b *(cos(theta));
    double b_y = magnitude_b *(sin(theta));
```

C Code

```c
// Resultant vector c = a + b
double c_x = a_x + b_x;
double c_y = a_y + b_y;

// Open file for writing
FILE *file = fopen("vectors.dat", "w");
if (file == NULL) {
    printf("Error opening file!\n");
    return 1;
}

// Write header
fprintf(file, "# Vector data: a_x a_y b_x b_y c_x c_y theta\n
    ");

// Write vector components and angle
fprintf(file, "%.6f %.6f %.6f %.6f %.6f %.6f %.6f\n",
        a_x, a_y, b_x, b_y, c_x, c_y, theta);
```

C Code

```c
    fclose(file);
double root = pow(c_x*c_x + c_y*c_y, 0.5);

    printf("Vector data saved to vectors.dat\n");
    printf("Vector a: (%.3f, %.3f), magnitude: %.3f\n", a_x, a_y,
         magnitude_a);
    printf("Vector b: (%.3f, %.3f), magnitude: %.3f\n", b_x, b_y,
         magnitude_b);
    printf("Vector c: (%.3f, %.3f), magnitude: %.3f\n", c_x, c_y,
         root);
    printf("Angle theta: %.3f radians (%.1f degrees)\n", theta,
        theta * 180.0 / M_PI);

    return 0;
}
```

Python Code

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

def read_vector_data(filename):
    """Read vector data from the .dat file"""
    with open(filename, 'r') as f:
        lines = f.readlines()

    # Skip comment lines (starting with #)
    data_line = None
    for line in lines:
        if not line.strip().startswith('#') and line.strip():
            data_line = line.strip()
            break

    if data_line is None:
        raise ValueError("No data found in file")
```

Python Code

```
        # Parse the data: a_x a_y b_x b_y c_x c_y theta
    values = list(map(float, data_line.split()))

    return {
        'a': np.array([values[0], values[1]]),
        'b': np.array([values[2], values[3]]),
        'c': np.array([values[4], values[5]]),
        'theta': values[6]
    }

def plot_vectors(data):
    """Create visualization of vectors a, b, and c"""
    fig, ax = plt.subplots(1, 1, figsize=(10, 8))
```

Python Code

```
# Extract vectors
   a = data['a']
   b = data['b']
   c = data['c']
   theta = data['theta']

   # Plot vectors from origin
   # Vector a (red)
   ax.quiver(0, 0, a[0], a[1], angles='xy', scale_units='xy',
       scale=1,
           color='red', width=0.006, label='Vector a',
               linewidth=2)

   # Vector b (blue) - only once from origin
   ax.quiver(0, 0, b[0], b[1], angles='xy', scale_units='xy',
       scale=1,
           color='blue', width=0.006, label='Vector b',
               linewidth=2)
```

Python Code

```python
# Draw angle arc between vectors a and b
    angle_radius = min(np.linalg.norm(a), np.linalg.norm(b)) *
        0.3
    angle_arc = np.linspace(0, theta, 50)
    arc_x = angle_radius * np.cos(angle_arc)
    arc_y = angle_radius * np.sin(angle_arc)
    ax.plot(arc_x, arc_y, 'k-', linewidth=1.5)

    # Add angle label (just theta without specific value)
    mid_angle = theta / 2
    label_radius = angle_radius * 1.3
    label_x = label_radius * np.cos(mid_angle)
    label_y = label_radius * np.sin(mid_angle)
    ax.text(label_x, label_y, 'theta',
            fontsize=14, ha='center', va='center', weight='bold')
```

Python Code

```python
# Add vector magnitude labels
    ax.text(a[0]/2, a[1]/2 - 0.3, f'|a| = {np.linalg.norm(a):.1f}',
            fontsize=10, ha='center', color='red', weight='bold')
    ax.text(b[0]/2 - 0.3, b[1]/2, f'|b| = {np.linalg.norm(b):.1f}',
            fontsize=10, ha='center', color='blue', weight='bold')
    ax.text(c[0]/2, c[1]/2 + 0.3, f'|c| = {np.linalg.norm(c):.1f}',
            fontsize=10, ha='center', color='green', weight='bold')

    # Set equal aspect ratio and grid
    ax.set_aspect('equal')
    ax.grid(True, alpha=0.3)
    ax.legend(loc='upper right', fontsize=11)
```

Vector Representation