

## Open-Ended Lab 13

---

### Scenario: *Smart Fitness Tracker System*

Imagine a company developing a **smart fitness tracker** that monitors a user's daily activities. Each tracker can record steps, heart rate, and calories burned. The company wants a C++ program to model these devices and provide tools for users to compare and manage their fitness data.

### Tasks:

#### **Design a base class named `FitnessDevice`**

- Decide suitable data members for device identification and battery status
- Add a function to display basic device information

#### **Create a derived class `SmartTracker`**

- Extend the base class with health-related data
- Redefine the display function to show complete tracker details

#### **Handle object creation properly**

- Implement required constructors and a destructor
- Maintain a count of how many tracker objects exist
- Provide a way to display this count

#### **Enable combining tracker data**

- Overload an operator to merge two tracker objects
- Decide how numerical health data should be combined
- Return a new object after combining

#### **Manage object copying**

- Implement a copy constructor
- Ensure copying does not cause data sharing issues

#### **Work with dynamic memory**

- Create tracker objects dynamically
- Demonstrate object combination
- Display total device count
- Release all allocated memory correctly

**Expected Outcome:**

- Students submit:
  - Code in **pasted form**
  - Sample **output** showing object creation, combination, and destruction
  - Short **discussion** explaining how each OOP concept was implemented
  - Submit this Lab in the form of Lab manual/Lab Report .

**Lab #13 Marks distribution**

	<b>ER1</b>	<b>ER6</b>	<b>ER8</b>
<b>Task</b>	3 points	3 points	4 points

**Lab #13 Rubric Evaluation Guideline:**

#	Qualities & Criteria	0 < Poor <=0.5	0.5 < Satisfactory <= 1.5	1.5 < Excellent <=3
<b>ER 1</b>	<b>Task Completion</b>	Minimal or no program functionality was achieved.	Some tasks were completed, but the program has errors or incomplete functionalities.	All tasks were completed, and the program runs without errors.
#	Qualities & Criteria	0 < Poor <=0.5	0.5 < Satisfactory <= 1.5	1.5 < Excellent <=3
<b>ER 6</b>	Program Output	Output is inaccurate or poorly presented.	Output is mostly accurate but may lack labels, captions, or formatting.	Output is clear, accurate, and well presented with labels, captions, and proper formatting.
#	Qualities & Criteria	0 < Poor <= 1.5	1.5< Satisfactory <= 3	3< Excellent <= 4
<b>ER 8</b>	Question & Answer	Answers some questions but not confidently or based on lab task knowledge.	Answers most questions confidently and based on lab task knowledge.	Answers all questions confidently and demonstrates a deep understanding of the given lab task.

