**Assignment 1 – Single Responsibility Principle (SRP) in C++**

**Lab Assignment:**

**Objective:**

Implement a simple embedded system class related to a temperature sensor that adheres to the Single Responsibility Principle.

**Instructions:**

Create a class named TemperatureSensor with the following responsibilities:

* Capturing temperature readings.
* Calculating the average temperature.
* Displaying the temperature information.

Ensure that each responsibility is in a separate class, following the Single Responsibility Principle.

Write a program to demonstrate the functionality of the TemperatureSensor class.

Solution Code:

#include <iostream>

#include <vector>

// Responsibility 1: Capturing Temperature Readings

class TemperatureReader {

public:

static std::vector<double> readTemperatures() {

// Simulate reading temperature data from the sensor

std::vector<double> temperatures = {23.5, 24.0, 22.8, 23.2, 24.5};

return temperatures;

}

};

// Responsibility 2: Calculating Average Temperature

class TemperatureCalculator {

public:

static double calculateAverage(const std::vector<double>& temperatures) {

// Calculate the average temperature

if (temperatures.empty()) {

return 0.0; // Return 0 if no readings are available

}

double sum = 0.0;

for (const auto& temperature : temperatures) {

sum += temperature;

}

return sum / temperatures.size();

}

};

// Responsibility 3: Displaying Temperature Information

class TemperatureDisplay {

public:

static void displayTemperature(double averageTemperature) {

// Display the average temperature

std::cout << "Average Temperature: " << averageTemperature << " degrees Celsius\n";

}

};

// Embedded System Class (Adhering to SRP)

class TemperatureSensor {

public:

void operate() {

// Responsibility 1: Capturing Temperature Readings

std::vector<double> temperatureReadings = TemperatureReader::readTemperatures();

// Responsibility 2: Calculating Average Temperature

double averageTemperature = TemperatureCalculator::calculateAverage(temperatureReadings);

// Responsibility 3: Displaying Temperature Information

TemperatureDisplay::displayTemperature(averageTemperature);

}

};

int main() {

// Create and operate the temperature sensor

TemperatureSensor temperatureSensor;

temperatureSensor.operate();

return 0;

}

In this example, we have three classes, each with a single responsibility related to the temperature sensor:

* TemperatureReader: Captures temperature readings.
* TemperatureCalculator: Calculates the average temperature.
* TemperatureDisplay: Displays temperature information.

The TemperatureSensor class operates by delegating each responsibility to the corresponding specialized class. This design adheres to the Single Responsibility Principle, ensuring that each class has only one reason to change. The main function demonstrates how these responsibilities are utilized in a program.