

Sistema de Gestión Polimórfica de Sensores para IoT.

Introducción

El presente documento detalla la implementación de un Sistema de Gestión Polimórfica de Sensores para Infraestructura Crítica (IC). El objetivo principal es desarrollar una solución de software de bajo nivel que aborde las limitaciones de rigidez de tipo de dato y rigidez estructural en la monitorización heterogénea de sensores (ej. temperatura, presión).

La arquitectura del sistema capitaliza los paradigmas avanzados de la Programación Orientada a Objetos (POO) y la gestión manual de estructuras de datos en C++ para crear un contenedor unificado capaz de manejar distintos tipos de sensores y sus respectivas lecturas de forma dinámica y segura. Se hace uso intensivo de polimorfismo a través de herencia de clases abstractas y plantillas (templates) para la generalización del tipo de dato de las lecturas.

Diseño

Componentes Clave

Componente	Funcionalidad
SensorBase	Define la interfaz común (<code>procesarLectura()</code> , <code>imprimirInfo()</code>) mediante métodos virtuales puros. Incluye un destructor virtual (<code>virtual ~SensorBase()</code>) esencial para la correcta liberación de memoria en cascada del polimorfismo.
SensorTemperatura	Contiene una instancia de <code>ListaSensor<float> historial</code> ; Su método <code>procesarLectura()</code> implementa la lógica de negocio específica: eliminar el valor más bajo y calcular el promedio restante.
SensorPresion	Contiene una instancia de <code>ListaSensor<int> historial</code> ; Su método <code>procesarLectura()</code> implementa la lógica específica: calcular el promedio de todas las lecturas.
ListaSensor<T>	Provee la funcionalidad dinámica de la estructura de datos. Implementa las operaciones básicas (inserción, búsqueda, promedio). Su diseño adhiere a la Regla de los Tres/Cinco para el manejo explícito de punteros (<code>Nodo<T>* cabeza</code>) y prevenir fugas de memoria.
SistemaGestion	Controla el flujo de la aplicación. Gestiona la lista polimórfica de punteros <code>SensorBase*</code> . Su método <code>procesarTodosSensores()</code> es el punto de aplicación del polimorfismo, invocando <code>procesarLectura()</code> de forma unificada.

Desarrollo

Se implementa la interfaz común en la clase base abstracta, forzando a las clases concretas a definir sus operaciones específicas de procesamiento y presentación, lo que garantiza la uniformidad en la interacción

del sistema central con sus componentes.

La liberación de recursos se maneja en cascada. El destructor de `SistemaGestion` recorre su listam elimina cada `SensorBase*`. El destructor de cada subclase, a su vez, invoca la función de liberación de memoria de su lista genérica interna (`ListaSensor<T>::liberarNodos()`).

El proyecto está configurado para ser gestionado y distribuido mediante CMake, lo que facilita la compilación multiplataforma y la creación de un sistema redistribuible.

Pantallasos de la Implementación Generada

```
clomezap@me:~/Documentos/EstructuraDatos/ds-u2-actividad-1-listasenlazadassimples-clomezap-edu/build$ ./sistema_iot
```

```
Sistema de Gestión Polimórfica de Sensores para IoT ||
```

```
=== Sistema IoT de Monitoreo Polimórfico Iniciado ===
```

```
--- Opción 1: Crear Sensores ---
```

```
[Log] SensorBase 'T-001' construido.  
[Log] ListaSensor<T> creada.  
[Log] SensorTemperatura 'T-001' creado.  
[Sistema] Sensor 'T-001' agregado a la lista de gestión.  
[Log] SensorBase 'P-105' construido.  
[Log] ListaSensor<T> creada.  
[Log] SensorPresion 'P-105' creado.  
[Sistema] Sensor 'P-105' agregado a la lista de gestión.
```

```
--- Opción 2: Registrar Lecturas ---
```

```
[T-001] Registrando lectura de temperatura: 45.3°C  
[Log] Nodo<T> insertado. Valor: 45.3  
[T-001] Registrando lectura de temperatura: 42.1°C  
[Log] Nodo<T> insertado. Valor: 42.1  
[T-001] Registrando lectura de temperatura: 47.8°C  
[Log] Nodo<T> insertado. Valor: 47.8  
[P-105] Registrando lectura de presión: 80 kPa  
[Log] Nodo<T> insertado. Valor: 80  
[P-105] Registrando lectura de presión: 85 kPa  
[Log] Nodo<T> insertado. Valor: 85  
[P-105] Registrando lectura de presión: 82 kPa  
[Log] Nodo<T> insertado. Valor: 82
```

```
===== Sensores Registrados =====
```

```
[1]  
=== Sensor de Temperatura ===  
ID: T-001  
Tipo: Temperatura (float)  
Lecturas almacenadas: 3  
Promedio actual: 45.0667°C  
=====
```

```
[2]  
=== Sensor de Presión ===  
ID: P-105  
Tipo: Presión (int)  
Lecturas almacenadas: 3  
Promedio actual: 82 kPa  
=====
```

```
===== Ejecutando Procesamiento Polimórfico =====
```

```
-> Procesando Sensor T-001 (Temperatura)...  
[Log] Nodo<T> 42.1 (menor) eliminado.  
[T-001] (Temperatura): Lectura más baja (42.1°C) eliminada. Promedio restante: 46.55°C.
```

```
-> Procesando Sensor P-105 (Presión)...  
[Sensor Presion] Promedio calculado sobre 3 lectura(s): 82 kPa.
```

```
===== Procesamiento Completado =====
```

```
--- Agregando más lecturas ---
```

```
[T-001] Registrando lectura de temperatura: 43.5°C  
[Log] Nodo<T> insertado. Valor: 43.5  
[P-105] Registrando lectura de presión: 88 kPa  
[Log] Nodo<T> insertado. Valor: 88
```

```
--- Procesamiento adicional ---
```

```
===== Ejecutando Procesamiento Polimórfico =====
```

```
-> Procesando Sensor T-001 (Temperatura)...  
[Log] Nodo<T> 43.5 (menor) eliminado
```

```
[Log] Node-V2-1015 (Sensor) Encendido.  
[T-001] (Temperatura): Lectura más baja (43.5°C) eliminada. Promedio restante: 46.55°C.  
  
-> Procesando Sensor P-105 (Presión)...  
[Sensor Presion] Promedio calculado sobre 4 lectura(s): 83 kPa.  
===== Procesamiento Completado =====  
  
--- Demostración de Búsqueda ---  
[Sistema] Sensor encontrado: T-001  
  
== Sensor de Temperatura ==  
ID: T-001  
Tipo: Temperatura (float)  
Lecturas almacenadas: 2  
Promedio actual: 46.55°C  
=====
```

```
--- Opción 5: Cerrar Sistema (Liberar Memoria) ---

--- Liberación de Memoria en Cascada ---
[Destructor General] Liberando Nodo: T-001
[Destructor SensorTemperatura] Sensor 'T-001' liberando recursos...
[Destructor ListaSensor] Liberando lista interna...
  [Log] Nodo<T> 45.3 liberado.
  [Log] Nodo<T> 47.8 liberado.
[Destructor SensorBase] Sensor 'T-001' destruido.
[Destructor General] Liberando Nodo: P-105
[Destructor SensorPresion] Sensor 'P-105' liberando recursos...
[Destructor ListaSensor] Liberando lista interna...
  [Log] Nodo<T> 80 liberado.
  [Log] Nodo<T> 85 liberado.
  [Log] Nodo<T> 82 liberado.
  [Log] Nodo<T> 88 liberado.
[Destructor SensorBase] Sensor 'P-105' destruido.
clemezap@me:~/Documentos/EstructuraDatos/ds-u2-actividad-1-listasenlazadassimples-clemezap-edu/build$
```

Archivo Editar Programa Herramientas Ayuda

sensor_simulator

/**
 * @file sensor_simu
 * @brief Simulador
 * @details Envía le
 * @platform ESP32
 *
 * INSTRUCCIONES:
 * 1. Abrir en Ardui
 * 2. Seleccionar pl
 * 3. Configurar pue
 * 4. Cargar el sket
 * 5. Abrir monitor
 */

// ===== CONFIG
#define BAUD_RATE 11
#define INTERVALO_EN
#define TEMP_MIN 20.
#define TEMP_MAX 50.
#define PRESION_MIN
#define PRESION_MAX

// ===== VARIAB
unsigned long ultimo
int contadorEnvios =

/**
 * @brief Configurac
 */
void setup() {
 // Inicializar com
 Serial.begin(BAUD

 // Esperar a que e
 delay(1000);

 // Inicializar generador de números aleatorios
 randomSeed(analogRead(0));

 // Mensaje de inicio
 Serial.println("\n=====");
 Serial.println(" ESP32 - Simulador de Sensores IoT");
 Serial.println("=====");
 Serial.println("Enviando lecturas cada 3 segundos...\n");

 delay(500);
}

Auto Formato Ctrl+T

Archivo de programa.

Reparar codificación & Recargar.

Administrar Bibliotecas... Ctrl+Mayús+I

Monitor Serie Ctrl+Mayús+M

Serial Plotter Ctrl+Mayús+L

Placa: "ESP32 Dev Module" >

Upload Speed: "115200" >

CPU Frequency: "240MHz (WiFi/BT)" >

Flash Frequency: "80MHz" >

Flash Mode: "QIO" >

Flash Size: "4MB (32Mb)" >

Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)" >

Core Debug Level: "Ninguno" >

PSRAM: "Disabled" >

Arduino Runs On: "Core 1" >

Events Run On: "Core 1" >

Erase All Flash Before Sketch Upload: "Disabled" >

JTAG Adapter: "Disabled" >

Zigbee Mode: "Disabled" >

Puerto >

Obtén información de la placa >

Programador >

Quemar Bootloader

Gestor de tarjetas...

Arduino AVR Boards (Debian packaged) >

ESP32 Arduino >