

Отчет по лабораторной работе №2

Выполнил: Холев Артем Юрьевич, группа 22ПИ2

1. Постановка задачи

- Цель работы расширить компонент сортировки Eco.Lab1, механизмами включения/агрегирования калькуляторов и подтвердить корректность работы через модульный клиент
- Добавить поддержку операций сложения, вычитания, умножения и деления за счет компонентов Eco.CalculatorA-E
- Реализовать схему, в которой внешнему IEcoLab1 доступны IEcoCalculatorX и IEcoCalculatorY как напрямую, так и через агрегированные внутренние компоненты
- Продемонстрировать в UnitTest, что запросы QueryInterface позволяют получить любой интерфейс из любого другого

2. Архитектура решения

Компонент CEcoLab1 (SourceFiles/CEcoLab1.c) расширен по сравнению с первой лабораторной работой и комбинирует приемы включения/агрегирования

- Операции Addition/Subtraction используют IEcoCalculatorX из Eco.CalculatorA с резервным переходом на Eco.CalculatorB
- Интерфейсы Multiplication/Division предоставляет агрегированный Eco.CalculatorE (IEcoCalculatorY) с fallback на Eco.CalculatorD
- В экземпляре CEcoLab1 хранится IEcoUnknown внутреннего агрегата (m_pinnerUnknown), что позволяет проксировать запросы QueryInterface
- Метод QueryInterface делегирует запросы к внутренним компонентам

3. Реализация

3.1 Инициализация и подключение калькуляторов

Метод initCEcoLab1 последовательно обращается к IEcoInterfaceBus1, выбирает реализацию EcoCalculatorX (A или B) и агрегирует Eco.CalculatorE. При ошибке агрегирования используется включение Eco.CalculatorD. Таблицы виртуальных методов сохраняются для повторного использования. Функция deleteCEcoLab1 освобождает все зависимые интерфейсы и память через m_plMem

3.2 QueryInterface

CEcoLab1_QueryInterface проверяет входные параметры, возвращает IEcoLab1 и IEcoUnknown, а запросы IEcoCalculatorX/IEcoCalculatorY перенаправляет либо во внутренний, либо в включенные интерфейсы. Таким образом любой интерфейс может запросить остальные

3.3 API компонента

```
int16_t Addition(int16_t x, int16_t y);
```

```
int16_t Subtraction(int16_t x, int16_t y);
```

```
int16_t Multiplication(int16_t x, int16_t y);
```

```
int16_t Division(int16_t x, int16_t y);
```

Арифметические методы выступают обертками над IEcoCalculatorX/IEcoCalculatorY, что позволяет прозрачно менять внутренние реализации без правок клиента

4. Тестирование

- (UnitTestFiles/SourceFiles/EcoLab1.c) выполняет сборку данных, замер сортировки и проверку интерфейсов
- Проводятся цепочки QueryInterface: IEcoLab1, IEcoCalculatorX/IEcoCalculatorY, перекрестные запросы между калькуляторами
- На примере значений 6 и 7 выводятся результаты всех арифметических операций

Выводы

Компонент Eco.Lab1 реализует схему включения и агрегирования калькуляторов A, B, D, E и обеспечивает доступ к IEcoCalculatorX/IEcoCalculatorY из любых интерфейсов

Арифметические операции делегируются внутренним модулям; модульные тесты подтверждают корректность вычислений

Interfaces:

IEcoCalculatorX from EcoLab1 - correct

IEcoCalculatorY from EcoLab1 - correct

IEcoCalculatorX from IEcoCalculatorY - correct

IEcoCalculatorY from IEcoCalculatorX - correct

Operations:

Addition test

$6 + 7 == 13$

Subtraction

$6 - 7 == -1$

Multiplication

$6 * 7 == 42$

Division

$6 / 7 == 0$