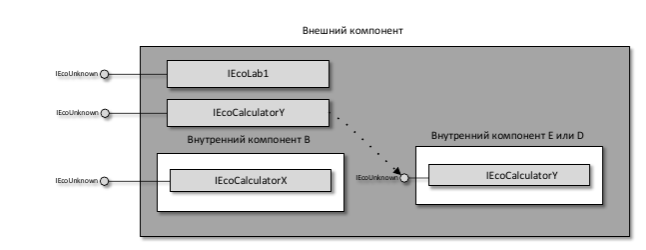
**Отчет по лабораторной работе №2**

1. **Постановка задачи**

В данной работе необходимо в ранее созданный компонент используя приемы программирования включение/агрегирование добавить компонент(-ы) калькулятора, выполняющий(-ие) операции сложения, вычитания, деления и умножения.



При организации компонентов была использована схема выше. В ней внешний компонент EcoLab1 реализует интерфейсы IEcoLab1, IEcoCalculatorxX и IEcoCalculatorY. При этом, компоненты A, E и D включаются во внешний компонент, тогда как компонент B поддерживает агрегирование внешним компонентом.

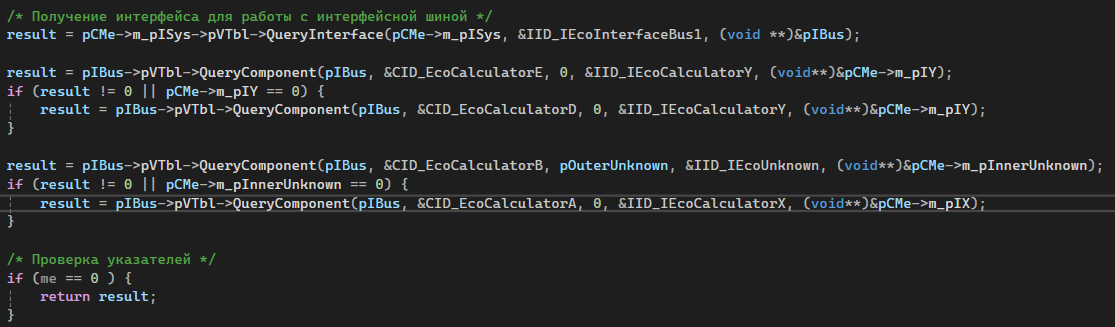
1. **Реализация**

Для того, чтобы включить/агрегировать компоненты, нужно провести несколько шагов.

1. В заголовочном файле компонента поместить указатели на интерфейсы и их виртуальные таблицы, реализуемые включаемыми компонентами.
2. Реализовать все необходимые методы во внешнем компоненте.
3. В методе initCEcoLab1 получить экземпляры компонентов через интерфейсы, которые реализуют эти компоненты. При этом в конкретно рассматриваемой среде, в зависимости от доступности, могут быть доступны следующие пары, реализующие интерфейсы X/Y: A/D; A/E; B/D; B/E.

При этом стоит сказать, что при получении компонента B в метод QueryComponent третьим параметром указывается pOuterUnknown - указатель на IEcoUnknown внешнего компонента. В остальных случаях, третьим параметром подается 0, так как компоненты A, D, E не поддерживают агрегирование и могут только включаться во внешний компонент.



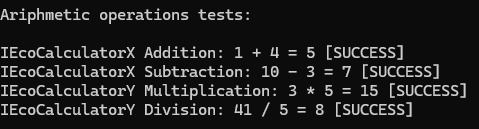


\*Примечание: компонент C имеет такой же идентификатор, что и компонент A, поэтому в компонентной среде он не представлен.

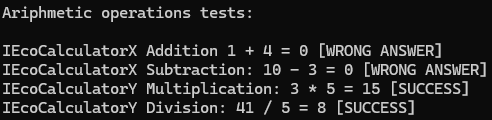
\*Примечание 2: компоненты A, B, C, D, E помещены в директорию Eco.Framework и пути до их заголовочных файлов указаны в проектах решения.

1. **Тестирование**

Первое, что тестируется, это работа компонентов на примере математических операций. Ожидается, что доступны хотя-бы по одному компоненту, реализующие используемые интерфейсы. Но даже если не доступно не одного компонента, реализующего определенный интерфейс, то программа просто будет выдавать неправильный результат, но выполнение не прекратится, и сам интерфейс остается доступным клиенту.

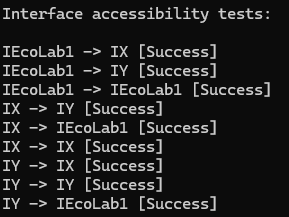


Вывод тестов, если имеется хотя-бы одна пара компонентов, один из которых реализует X, другой Y



Вывод тестов, если не доступно ни одного компонента, реализующего интерфейс X, но есть хотя бы один, реализующий Y

Далее проверяется доступность в получении каждого интерфейса из каждого интерфейса. Это гарантирует, имея хотя бы один интерфейс, реализуемый компонентом, клиент может получить все остальные интерфейсы, реализуемые тем же компонентом, что является одним из важных правил технологии АОМК.



Демонстрация доступности интерфейсов компонента EcoLab1

1. **Выводы**

В данной лабораторной работе я рассмотрел механизмы включения/агрегирования компонентов. При этом я протестировал работу компонента EcoLab1 в различных условиях доступности тех или иных компонентов, при этом построив включение/агрегирование таким образом, что внешний компонент пытается подключить любой доступный компонент, реализующий тот или иной интерфейс. При этом даже при недоступности всех таких компонентов, внешний компонент может предоставить интерфейс клиенту, и при этом не крашиться при вызове методов (но будет выдавать неверный результат, в данном случае, всегда будет 0). Помимо этого, соблюдена доступность всех интерфейсов, реализованных EcoLab1.