Нуранов Айдар 931920

1. Постановка задачи

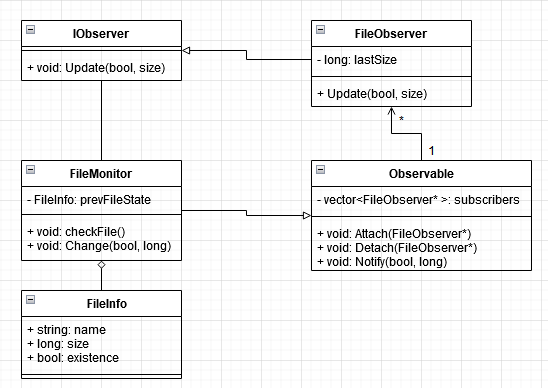
На основе паттерна Наблюдателя, написать программу для слежения за состоянием выбранного файла.   
  
Ограничимся  двумя характеристиками:  
1. Существует файл или нет;  
2. Каков размер файла.  
  
Программа будет выводить на консоль уведомление о произошедших изменениях в файле.  
Существует несколько ситуаций для наблюдаемого файла  
1. Файл существует , файл не  пустой - на экран выводится факт существования файла и его  размер.  
2. Файл существует, файл был изменен - на экран выводится факт существования файла, сообщение о том что файл был изменен и его размер.    
3. Файл не существует - на экран выводится информация о том что файл не существует.

2. Предлагаемое решение

**Наблюдатель** — это поведенческий паттерн проектирования, который создаёт механизм подписки, позволяющий одним объектам следить и реагировать на события, происходящие в других объектах.

Паттерн Наблюдатель предлагает хранить внутри объекта издателя список ссылок на объекты подписчиков, причём издатель не должен вести список подписки самостоятельно. Он предоставит методы, с помощью которых подписчики могли бы добавлять или убирать себя из списка. Когда в издателе будет происходить важное событие, он будет проходиться по списку подписчиков и оповещать их об этом, вызывая определённый метод объектов-подписчиков.

Все подписчики должны следовать общему интерфейсу и иметь единый метод оповещения.

Uml диаграмма: 

Для начала создаем шаблонный класс наблюдателя, от которого будет реализован конкретный класс FileObserver. Создадим так же класс, хранящий информацию об файле, хранящий состояние файла такие как размер и существование. Наконец реализация класса FileMonitor который будет оповещать всех наблюдателей об изменении файла.

3. Коды программ

fileinfo.h

fileinfo.cpp

observable.h

observable.cpp

observer.h

observer.cpp

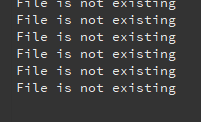
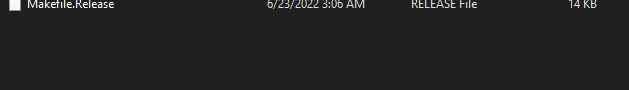
4. Инструкция пользователя

В файле main.cpp необходимо создать переменные типа FileMonitor, с аргументом пути до файла, который мы хотим отслеживать, и наблюдателя. Теперь необходимо связать эти переменные с помощью функции Attach.

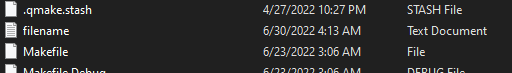
Далее идет проверка файла на его изменения, в бесконечном цикле спустя n секунд.

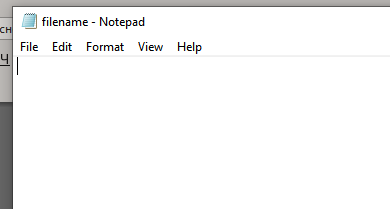
5. Тестирование

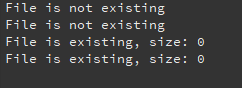
При запуске программы и не существования файла на консоль будет выводиться сообщение о том, что файла не существует: File is not existing



После того как файл будет создан выведется сообщение о его существовании и так как он пустой, его размер будет 0: File is existing, size: 0

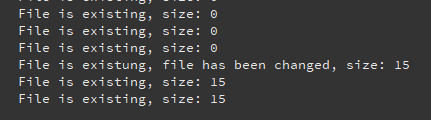
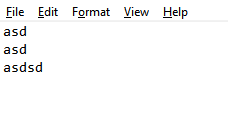






Далее каждые несколько секунд будет выводиться на консоль эта информация

Добавим в файл несколько символов и сохраним в консоль выведется сообщение о том, что файл был изменен: File is existung, file has been changed, size: 3



После того как удалим файл выведется сообщение что файл снова не существует: File is not existing