Задачи на паттерны проектирования

**Для указанного варианта задания необходимо нарисовать UML-диаграмму классов реализуемой программы и разработать консольное приложение на языке программирования.**

1 Паттерн Builder. Имеется текст статьи в формате TXT. Статья состоит из заголовка, фамилий авторов, самого текста статьи и хеш-кода текста статьи. Написать приложение, позволяющее конвертировать документ в формате ТХТ в документ формата XML, необходимо также проверять корректность хеш-кода статьи.

2 Паттерн Abstract Factory. Разработать систему Кинопрокат. Пользователь может выбрать определённую киноленту, при заказе киноленты указывается язык звуковой дорожки, который совпадает с языком файла субтитров. Система должна поставлять фильм с требуемыми характеристиками, причём при смене языка звуковой дорожки должен меняться и язык файла субтитров и наоборот.

3 Паттерн Factory Method. Фигуры игры «тетрис». Реализовать процесс случайного выбора фигуры из конечного набора фигур. Предусмотреть появление супер-фигур с большим числом клеток, чем обычные.

4. Паттерн Наблюдатель (Observer). Разработать систему подписки (с возможностью отписки) и доставки журналов и газет подписчикам. Журналы доставляются один раз в месяц с предварительным уведомлением и согласием на доставку. Газеты доставляются еженедельно без предварительного уведомления.

5. Разработать диаграмму конечных автоматов для заданного класса (таблица 1). Описать в форме таблицы варианты реакции экземпляра класса на операции, вызываемые в указанных состояниях.

Таблица 1 – Варианты заданий для разработки приложения с использованием шаблона Состояние

|  |  |  |
| --- | --- | --- |
| Номер варианта | Классы, их атрибуты и операции | Состояние |
| 1 | Телефон  Атрибуты: номер, баланс, вероятность поступления звонка.  Операции: позвонить, ответить на звонок, завершить разговор, пополнить баланс | Ожидание, Звонок, Разговор, Заблокирован (баланс отрицательный) |
| 2 | Банкомат  Атрибуты: ID, общая сумма денег в банкомате, вероятность отсутствия связи с банком.  Операции: ввести PIN-код, внять заданную сумму, завершить работу, загрузить деньги в банкомат | Ожидание, Аутентификация пользователя, Выполнение операций, Заблокирован (денег нет) |
| 3 | Грузовой лифт  Атрибуты: текущий этаж, грузоподъёмность, вероятность отключения электроэнергии.  Операции: вызвать на заданный этаж, загрузить, разгрузить, восстановить подачу энергии | Покой, Движение, Перегружен,  Нет питания |

Разработать библиотеку классов, включающую необходимые классы для реализации шаблона Состояние (класс Конечный автомат, интерфейс Состояние, классы Конкретные состояния).

Разработать приложение для управления состояниями экземпляров класса Конечный автомат.

6. На основе паттерна Decorator Разработать программу вычисления цены за сервисное техобслуживание автомобилей разных марок. При этом учитывать разную базовую цену за облуживание разных марок, и разные соответствующие коэффициенты за каждую операцию обслуживания и устранения выявленного дефекта (предусмотреть не мене трех).

7. Используя средства ОО-языка программирования решить на основе паттерна Adapter следующую задачу.

Пусть задан класс с именем Original, в котором есть 3 метода с именами OriginalDouble(), OriginalInt(), OriginalChar().

Каждый из методов получает по одному параметру:

* Метод OriginalDouble() получает значение типа double и выводит его на экран;
* Метод OriginalInt() получает значение типа int и выводит его на экран;
* Метод OriginalChar() получает значение типа char и выводит его на экран.

Нужно адаптировать класс Original к потребностям класса Client, который имеет следующие требования:

* Метод OriginalDouble() нужно переименовать в ClientDouble(). Работа метода остается неизменной;
* Метод OriginalInt() нужно заменить методом ClientInt(). Метод ClientInt() должен выводить двойное значение (умноженное на 2) параметру типа int;
* Метод OriginalChar() нужно заменить методом ClientChar(). Метод ClientChar() выводит параметр типа char 5 раз.

Класс Client должен получать ссылку на интерфейс ITarge в конструкторе.

8. Реализовать два алгоритма обхода графа (DFS — поиск в глубину и BFS — поиск в ширину), используя паттерн проектирования «Шаблонный метод».

Для переопределения должны быть доступны следующие методы:

* Начало обхода.
* Конец обхода.
* Посещение узла *V*.
* Посещение ребра *E*.

Оба алгоритма должны иметь унифицированный интерфейс (реализуя, таким образом, паттерн «Стратегия»).

Абстракцию графа желательно вынести в отдельный класс, сделав его либо параметром шаблона, либо передавая в алгоритм указатель/ссылку на объект базового класса представления графа.

9. Используя паттерн Шаблонный метод написать систему классов, которая позволит применять различные сходящиеся итерационные алгоритмы, которые все имеют схожий алгоритм:

1. Проверить некоторое условие с некоторой точностью. Если верно - вернуть текущий результат *X*k
2. Если нет - вычислить следующий член последовательности предполагаемых результатов *X*k+1 = *g*(*X*k) и т.д.

В этой задаче необходимо численно решить одномерное уравнение *f*(*x*)=0 на интервале [a, b] (в предположении, что на заданном интервале есть только один корень). Хочется уметь это делать разными методами:

* [Метод хорд](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D1%85%D0%BE%D1%80%D0%B4).
* [Метод Ньютона](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%9D%D1%8C%D1%8E%D1%82%D0%BE%D0%BD%D0%B0).

Необходимо реализовать эти два алгоритма, используя паттерн проектирования «Шаблонный метод».

Обратите внимание, что метод Ньютона кроме основной функции *f* требует задания её производной *f*'. Учтите этот факт при разработке интерфейса классов.

10. Используя шаблон Наблюдатель (Observer), разработать программу отслеживания изменений в таблице БД и отражении этих изменений на столбчатых и круговых диаграммах. За таблицей может следить произвольное число наблюдателей, причем их добавление и удаление не влияет на БД.

Класс БД реализует операции по отслеживанию изменений в таблице и, при их наличии, сразу информирует абстрактного наблюдателя. Тот, в свою очередь, вызывает операции по перерисовке соответствующих диаграмм у конкретных наблюдателей.

11. Написать декоратор, который преобразует все символы верхнего регистра во входном потоке к нижне­му регистру. Другими словами, если из потока читается строка «I know the Decorator Pattern therefore I RULE!», то наш декоратор преобразует ее к виду «i know the decorator pattern therefore i rule!»

12. Необходимо написать API для нового Пульта Домашней Автоматизации.

Пульт имеет семь программируемых ячеек (каждая из которых связывается с отдельным домашним устройством) и соответствующую кнопку «вкл/выкл» для каждой ячейки. Кроме того, устройство оснащено кнопкой глобальной отмены.

Также прилагается диск с набором классов (в решении это будут классы-заглушки), созданных разными фирмами-разработчиками для управления всевозможными домашними устройствами: светильниками, вентиляторами, ваннами-джакузи, акустическим оборудованием и т. д.

Ваша задача — создать API для программирования пульта, чтобы каждая ячейка могла быть настроена на управление устройством или группой устройств. Также следует учесть, что пульт должен поддерживать как текущий набор устройств, так и все устройства, которые могут быть добавлены в будущем.

Для выполнения задания воспользуйтесь паттерном Команда.

13. Используя паттерн Шаблонный метод реализовать систему, которая позволит применять различные методы сортировки массива. Выберите для этого два схожие метода сортировки, которые имеют общие элементы.