

## Задания по курсу “Программная инженерия”.

1. Реализуйте шаблон “агрегатор событий” на базе паттернов проектирования “посредник” и “наблюдатель”. Примените шаблон на примере отслеживания изменений данных в базе данных.

2. Реализуйте оконное или web-приложение, позволяющее строить карту городов и дорог между ними (дорог между двумя городами может быть произвольное количество). В интерактивном режиме пользователь может выполнять следующие действия: добавить город с уникальным именем, удалить существующий город, изменить название города (необходимо сохранение уникальности имени среди всех имён в рамках карты), добавить дорогу между городами (с указанием стоимости пути по дороге), удалить дорогу между городами, изменить стоимость пути. Обеспечьте функционал сохранения/загрузки карты (с сохранением истории изменений) и undo-redo (используйте шаблоны проектирования “команда” и “хранитель”).

3. Реализуйте интерфейс, предоставляющий описание функционала по вычислению значения выражения, токенами которого могут являться вещественные числа, операции, а также круглые скобки. Функционал при реализации должен делегировать запросы методам сервиса-декоратора, реализующего интерфейс, либо задекорированному сервису. Реализуйте следующие декораторы:

- предоставляющий стандартные арифметические операции над вещественными значениями (сложение, умножение, вычитание, деление, взятие остатка от деления)
- предоставляющий тригонометрические функции (синус, косинус, тангенс, котангенс);
- предоставляющий битовые операции (поразрядные конъюнкция, дизъюнкция, функция Вебба, штрих Шеффера, стрелка Пирса, инверсия);
- предоставляющий операции возведения в положительную целую степень, взятия факториала (гамма-функции), взятия модуля.

Функционал представляет собой два перегруженных метода: выражение может быть передано в виде объекта дерева выражения (примените шаблон проектирования “компоновщик”), либо в виде строки,

которую необходимо преобразовать в объект дерева выражения встроеной в объект реализацией паттерна “адаптер”. Для генерации цепочки вложенных декораторов используйте “фабричный метод”, делегирующий задачу по построению цепочки паттерну “строитель”.

4. Смоделируйте систему “копировальный сервис”. Сервис умеет обрабатывать запросы пользователей (запросы могут поступать не последовательно) на печать (фотографии разных форматов, документы формата А4 (например, диплом) в цветном или чёрно-белом варианте и т. д.). Для печати по определённому запросу необходимо отправить запрос на определённый принтер (первоначально запрос поступает на ч/б принтер, далее на цветной по необходимости (в зависимости от запроса) - примените шаблон проектирования “цепочка обязанностей”), далее настроить целевой принтер под тип документа в запросе (примените паттерн “состояние”) и выполнить печать (имитируйте процесс печати ожиданием). При печати фотографий рассмотрите варианты, когда в запросе уже есть фотография или же когда её нет (когда нет, необходимо проделегировать запрос сервису фотографирования - примените шаблон проектирования “заместитель”). Время в системе дискретно. Начальные настройки частей системы должны быть псевдослучайными. Продемонстрируйте работу системы. Обеспечьте наглядный вывод информации о процессе работы и результатах работы системы.

5. Смоделируйте систему “сервисный центр”. В СЦ могут обращаться клиенты с запросами на ремонт техники (вид техники выберите самостоятельно) (запросы могут поступать не последовательно). При ремонте может понадобиться замена некоторых компонентов, которые должны быть запрошены из пула, а при их отсутствии - компоненты необходимо произвести на базе существующих “чертежей” (примените паттерны проектирования “абстрактная фабрика”, “одиночка” (для фабрики) и “прототип”). При замене компонентов, сломанные компоненты отправляются “в мастерскую” (отдельный сервис) для ремонта (ремонт занимает некоторое количество времени) и при успешном ремонте помещаются в пул, а при неуспешном - отправляются на утилизацию. Механики ремонта, утилизации, поступления новых клиентов определите самостоятельно. Для модели реализуйте фасад, позволяющий получать: состояние пула, активные запросы клиентов, состояние мастерской,

информацию об утилизированных компонентах. Время в системе дискретно. Начальные настройки частей системы должны быть псевдослучайными. Продемонстрируйте работу системы. Обеспечьте наглядный вывод информации о процессе работы и результатах работы системы.