Министерство высшего образования и науки Российской Федерации

Новосибирский государственный технический университет

Лабораторная работа №1 по дисциплине  
«Технология программирования» на тему

***Основы программирования на Java.***

***Обработка событий. Механизм делегирования событий.***

Вариант №7

Факультет: АВТФ Преподаватель:

Группа: АВТ-810 Михайленко Д. А.

Студенты: Степаненко Владислав

Новосибирск

2020

#### Практические задания

1. Познакомиться с особенностями технологии Java и изучить синтаксис языка Java.
2. Изучить основные понятия и термины обработки событий в модели делегирования событий на Java.
3. Разобрать приведенные примеры **FontsList**, **LinesDraw, KeyCodes, Timer**, **DrawFig**.
4. Разработать программу. Основная задача – разработка упрощенной имитации поведения объектов (все последующие лабораторные работы будут расширять это задание). Объекты реализуются через наследование: абстрактный класс + интерфейс → наследники.

Рабочий цикл программы:

* запускается процесс симуляции по клавише, генерируются объекты классов согласно заданию;
* симуляция завершается по другой клавише, выводится статистическая информация.

1. Для решения задачи:

* Разработать абстрактный класс объекта, согласно варианту индивидуального задания.
* Создать интерфейс IBehaviour, задающий поведение объекта (далее будут реализоваться алгоритмы движения объектов в окне программы).
* Реализовать иерархию классов, определяющих объекты по варианту и реализующие интерфейс IBehaviour.
* Создать интерфейс AbstractFactory для создания объектов классов.
* Реализовать от него класс ConcreteFactory, возвращающий нужный объект, согласно варианту индивидуального задания.
* По желанию можно использовать паттерн Factory Method.
* Создать класс Habitat (среда), определяющий размер рабочей области и хранящий массив объектов, с параметрами заданными вариантом. Предусмотреть в классе метод Update, вызывающийся по таймеру и получающий на вход время, прошедшее от начала симуляции. В данном методе должны генерироваться новые объекты и помещаться в поле визуализации в случайном месте. Визуализация объекта – схематично, плюсом будет, если объект будет похож на оригинал (можно использовать готовые небольшие картинки);

Рабочее окно программы – область визуализации среды обитания объектов;

1. Симуляция должна запускаться по клавише **B** и останавливаться по клавише **E**. При остановке симуляции список уничтожается. Время симуляции должно отображаться текстом в области визуализации и скрываться/показываться по клавише **T**;
2. По завершению симуляции в поле визуализации должна выводиться информация о количестве и типе сгенерированных объектов, а также время симуляции. Текст должен быть форматирован, т.е. выводиться с использованием разных шрифтов и цветов.
3. Параметры симуляции задаются в классе Habitat.

При необходимости или желанию можно использовать другие паттерны.

***Вариант 7***

Список жилых домов города состоит из двух типов: капитальный, деревянный. Капитальные дома генерируются каждые N1 секунд с вероятностью P1. Деревянные дома генерируются каждые N2 секунд с вероятностью P2.

**Ход работы:**

Для создание домов разных типов реализованы отдельные классы-наследники (Деревянный – Wood/ Каменный - Stone)абстрактного класса House хранящего поля – координаты. Интерфейс AbstractFactory от которого наследуются “фабрики” для обоих типов домов, фабрики в свою очередь нужны для создания экземпляров каменных и деревянных домов. Каждый дом имеет свое изображение формата .png

В классе Habitat задаются параметры “среды обитания”, симуляции (вероятности и время появления домов). В конструкторе класса описан “слушатель” который будет считывать нажатия клавиатуры и выполнять соответствующие задачи, заданы параметр окна (фрейма), текстовая область JTextArea которая будет использоваться для вывода времени и статистики, массив всех домов. Время сэмулировано при помощи классов TimerTask и Timer, они выполняют метод update каждые 1000 мс. Метод update при каждом вызове проверяет условия среды обитания и при необходимости создает определенные типы домов со случайными координатами и увеличивает их счетчик.

При нажатии B начинается отсчет таймера и сама симуляция.

При нажатии T включается отображение таймера.

При нажатии E происходит остановка таймера и симуляции, очищение массива и области отрисовки, вывод статистики.

В классе main создается объект типа Habitat.

**Вывод:**

В результате выполнения лабораторной работы было проведено ознакомление особенностями технологии Java и начато изучение синтаксиса языка Java. Были изучены основные понятия и термины обработки событий в модели делегирования событий на Java.

Была разработана программа для упрощенной симуляции поведения объектов, которая будет расширяться и дополняться в дальнейшем. Было проведено ознакомление с паттернами программирования. Основными сложностями при выполнении работы стали работа с изображениями и таймерами.