

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №2
по дисциплине «ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»
Тема: Интерфейсы классов; взаимодействие классов; перегрузка
операций

Студент гр. 8383

Ларин А.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2020

Цель работы.

Освоить интерфейсы классов, взаимодействие классов, перегрузку операций

Задание

Лабораторная работа №2 (Интерфейсы классов; взаимодействие классов; перегрузка операций)

Разработать и реализовать набор классов:

- Класс базы
- Набор классов ландшафта карты
- Набор классов нейтральных объектов поля

Класс базы должен отвечать за создание юнитов, а также учитывать юнитов, относящихся к текущей базе. Основные требования к классу база:

- База должна размещаться на поле
- Методы для создания юнитов
- Учет юнитов, и реакция на их уничтожение и создание
- База должна обладать характеристиками такими, как здоровье, максимальное количество юнитов, которые могут быть одновременно созданы на базе, и.т.д.

Набор классов ландшафта определяют вид поля. Основные требования к классам ландшафта:

- Должно быть создано минимум 3 типа ландшафта
- Все классы ландшафта должны иметь как минимум один интерфейс
- Ландшафт должен влиять на юнитов (например, возможно пройти по клетке с определенным ландшафтом или запрет для атаки определенного типа юнитов)
- На каждой клетке поля должен быть определенный тип ландшафта

Набор классов нейтральных объектов представляют объекты, располагаемые на поле и с которыми могут взаимодействовать юниты. Основные требования к классам нейтральных объектов поля:

- Создано не менее 4 типов нейтральных объектов
- Взаимодействие юнитов с нейтральными объектами, должно быть реализовано в виде перегрузки операций
- Классы нейтральных объектов должны иметь как минимум один общий интерфейс

Реализация

Класс базы

**Для наблюдения над юнитами в классе база используется паттерн “Наблюдатель”*

База не является самостоятельным классом, от нее наследуется класс юнитов Alpha, который выполняет роль базы. Класс Base не является интерфейсом в чистом виде т. к. содержит поле mediator, и является «наблюдателем», реализую часть логики

- База должна размещаться на поле
Юнит-база(Alpha) размещается на поле
- Методы для создания юнитов
В классе базы
`virtual bool spawnAt(UnitClass unitClass, int x, int y) = 0;`
В классе Alpha:Base
`bool spawnAt(UnitClass unitClass, int x, int y) override;`
- Учет юнитов, и реакция на их уничтожение и создание
База реализует в себе паттерн «наблюдателя», и узнает о каждом убийстве вражеского юнита или смерти союзного
- База должна обладать характеристиками такими, как здоровье, максимальное количество юнитов, которые могут быть одновременно созданы на базе, и.т.д.
Максимальное количество учитывается. Здоровье присуще юниту, являющимся базой.

Набор классов ландшафта карты

**Для взаимодействия ландшафта с юнитам используется паттерн “Прокси”*

Взаимодействие через перегрузку операторов

- Должно быть создано минимум 3 типа ландшафта
- Все классы ландшафта должны иметь как минимум один интерфейс
- Ландшафт должен влиять на юнитов (например, возможно пройти по клетке с определенным ландшафтом или запрет для атаки определенного типа юнитов)
Определенные типы юнитов(Delta, Epsilon, в планах Zeta) реагируют на изменение ландшафта, меняя свои свойства. Логика находится в перегруженном операторе +=
- На каждой клетке поля должен быть определенный тип ландшафта
Тип ландшафта определяется параметрический. Прокси (LandscapeProxy) содержит в себе координаты и указатель на объект медиатор, т. е. не содержит в себе параметров,

а получает их когда требуется. Среди параметров: Пояс(расстояние от центра), линия(расстояние от края), цвет(по принципу шахматной доски)

Таким образом для поля 8*8 существует 40 типов ландшафта, отличающихся тем или иным параметром, и по разному влияющих на юнитов.

Note: Клетка поля обернута в класс, еще более похожий на прокси: FieldCellProxy

Он содержит только координаты и указатель на посредника, но при запросе предоставляет указатель на находящийся в данной клетке объект(через медиатора), и объект типа LandscapeProxy, через который в свою очередь можно получить параметры ландшафта в текущих координатах. При этом ни указатель на объект ни параметры ландшафта не хранятся в прокси классе, но могут быть .

Набор классов нейтральных объектов представляют объекты, располагаемые на поле и с которыми могут взаимодействие юнитов. Основные требования к классам нейтральных объектов поля:

- Создано не менее 4 типов нейтральных объектов
EntityBush
EntityFire
EntityStone
EntityTree
- Взаимодействие юнитов с нейтральными объектами, должно быть реализовано в виде перегрузки операций
Для юнита += - наступил на что-то. Для объекта -= - наступили на него.
- *Классы нейтральных объектов должны иметь как минимум один общий интерфейс Entity*

Демонстрационные примеры.

1. Tho bases are set.

#12345678#

h	h
g	a g
f	f
e	e
d	d
c	c
b a	b
a	a

#12345678#

2. One sets Gamma, another Epsilon. Fire appeared all by itself

#12345678#

h		h
g	ca	g
f		f
e	*	e
d		d
c	e	c
b	a	b
a		a

#12345678#

Epsilons power(ord) is 6

3. Move Gamma to fire. Move Epsilon up

Class 2 was killed by class 7

Epsilons new power (due to landscape change): 4

#12345678#

h		h
g	a	g
f		f
e	e	e
d		d
c		c
b	a	b
a		a

#12345678#

Выводы.

В результате работы была написана полностью рабочая программа решающая поставленную задачу при использовании изученных теоретических материалов. Программа была протестирована, результаты тестов удовлетворительны.