

**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

Кормщикова А.О.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург  
2020

### **Цель работы.**

Разработка и реализация новых классов, разработка интерфейсов к ним, реализация взаимодействия классов, изучение перегрузки операций.

### **Постановка задачи.**

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
- *\*Для взаимодействия одного типа нейтрального объекта с разными типами юнитов используется паттерн “Стратегия”*
- *\*Для взаимодействия ландшафта с юнитами используется паттерн “Прокси”*

### **Ход работы.**

1) Файлы, добавленные в этой лабораторной работе:

NeutralObject.h/cpp - реализация нейтральных объектов,

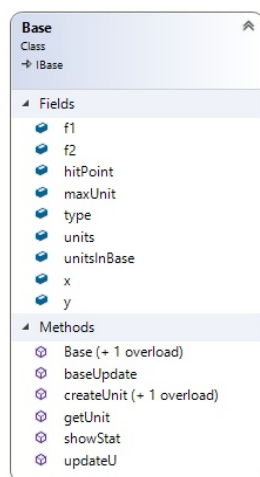
Base.h/cpp - реализация базы.

Измененные файлы:

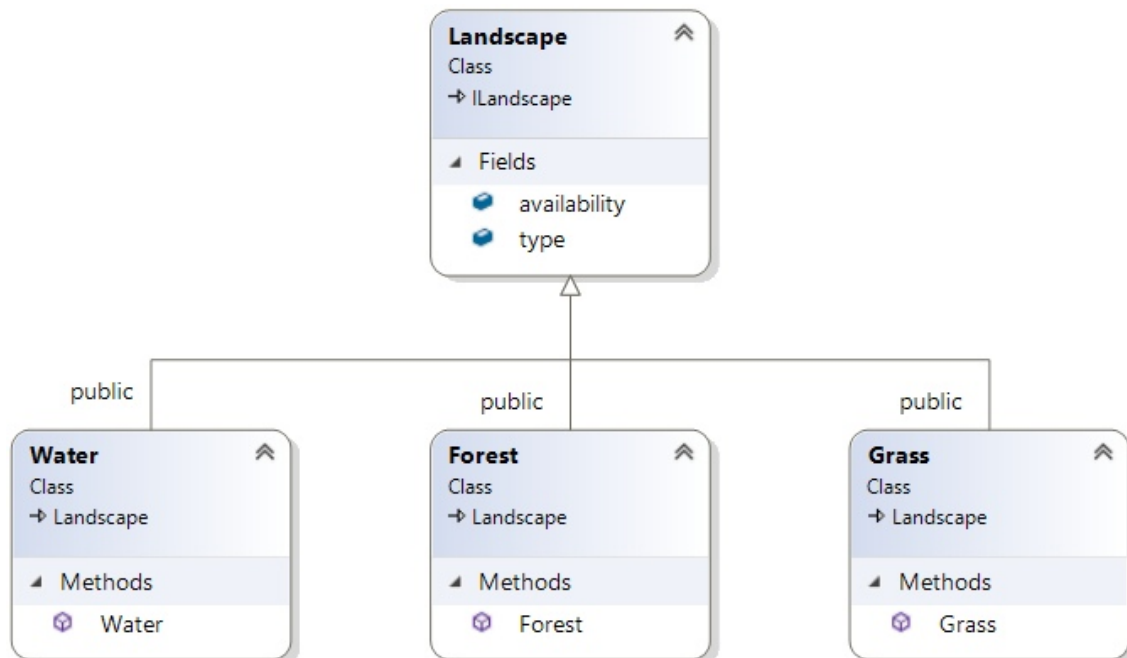
GameBlock.h/cpp - добавлены классы ландшафтов.

Interfaces.h - добавлены интерфейсы базы, ландшафта, нейтральных объектов, наблюдателя.

2) Был создан класс Base, содержащий поля для координат базы на игровом поле, ее здоровье, максимальное количество юнитов, принадлежащих этой базе, количество юнитов находящихся внутри данной базы (а не на поле), лист указателей на юнитов. Реализованы следующие методы: метод создания юнита в этой базе, использующий фабрики, метод, показывающий статистику, метод, вызывающий функцию обновления у юнитов (часть реализации паттерна Наблюдатель), метод, возвращающий указатель на выбранного юнита, метод, обновляющий сведения о юнитах в базе. На карте база обозначается символом =.

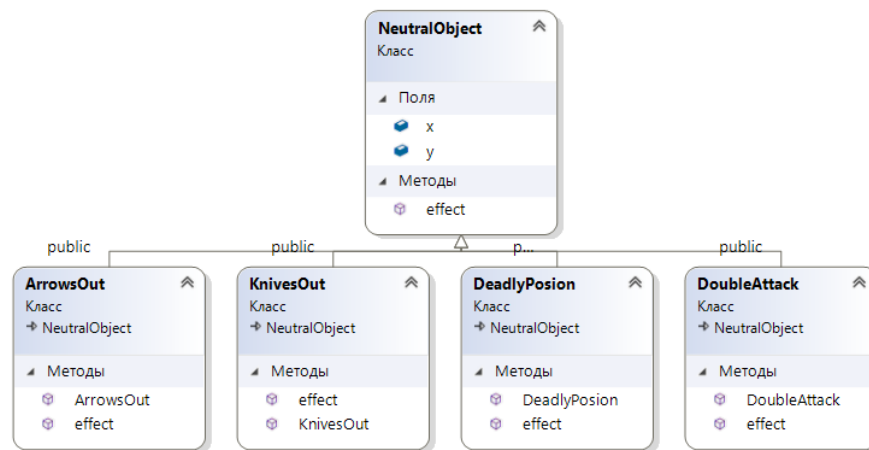


3) Был создан набор классов ландшафта (см. файл GameBlock.h), общий интерфейс. От класса Landscape, содержащих поля type и availability. Наследниками данного класса являются классы-ландшафты Grass, Forest, Water. Обозначения на игровой карте: Grass - пустая клетка, Water ~, Forest ^; Особенности ландшафта: по клеткам типа water запрещено передвижение, на клетке типа forest нельзя ставить базу.



4) Были созданы 4 типа нейтральных объектов, а также их общий интерфейс. Объект типа MrWitcher (на карте - @) - выдает случайные фразы, через некоторое количество взаимодействий с ним наносит 1000 урона. Объект Bird

(на карте - !) - чирикает и наносит 5 урона. Grave (на карте +) - выводит информацию о себе. Bonfire (на карте в рабочем состоянии &, потухший —) — один раз за игру может прибавить одному юниту 10 здоровья.



Для взаимодействия нейтральных объектов с юнитами был перегружен оператор `+=`.

5) Был реализован паттерн наблюдатель: класс `base` является издателем, а юниты, принадлежащие этой базе - подписчиками. В классе `base` содержится метод `updateU`, который у каждого юнита вызывает соответствующий перегруженный метод, который наследуется из виртуального класса `IObserver`.

### Тесты.

1) Создание и выбор юнита в базе, которому будет передано управление

```

Enter command for base 'c' - Create unit, 's' - select unit:
c

Enter type of unit:
R - Ranger
S - Slayer
W - Witch
B - Battle mage
P - Paladin
L - Lancer Knight
Default: P
W
Enter command for base 'c' - Create unit, 's' - select unit:
s

Base HP:50

Units in base:3
1
R
HP:100
MP:20
Magic resist:35
Armor:25

2
S
HP:100
MP:20
Magic resist:25
Armor:35

3
W
HP:100
MP:100
Magic resist:60
Armor:10

Count of units defined to this base:3
Enter number:

```

```

Enter number:
2

  ---  ---  ---  ---  ---  ---  ---  ---
| ^ | | ^ | | ^ | | ^ | | @ | |   | |   | | |
| ^ | | ^ | | ^ | |   | |   | |   | |   |
| ^ | | ^ | | ^ | |   | |   | | = | | ! | |
| ^ | | ^ | |   | |   | |   | |   | |   |
| ^ | |   | |   | |   | |   | |   | |   |
|   | |   | |   | | ~ | | ~ | | ~ | |   | |
|   | |   | | & | |   | | ~ | | ~ | | ~ | |   |
|   | |   | |   | | ~ | | ~ | | ~ | | + | |
|   | |   | |   | |   | |   | |   | |   |
  ---  ---  ---  ---  ---  ---  ---  ---

Enter command: w

  ---  ---  ---  ---  ---  ---  ---  ---
| ^ | | ^ | | ^ | | ^ | | @ | |   | |   | | |
| ^ | | ^ | | ^ | |   | |   | | S | |   | |
| ^ | | ^ | | ^ | |   | |   | | = | | ! | |
| ^ | | ^ | |   | |   | |   | |   | |   |
| ^ | |   | |   | |   | |   | |   | |   |
|   | |   | |   | | ~ | | ~ | | ~ | |   | |
|   | |   | | & | |   | | ~ | | ~ | | ~ | |   |
|   | |   | |   | | ~ | | ~ | | ~ | | + | |
|   | |   | |   | |   | |   | |   | |   |
  ---  ---  ---  ---  ---  ---  ---  ---

Enter command:

```

## 2) Смерть юнита, передача управления базе

```

  ---  ---  ---  ---  ---  ---  ---  ---
| ^ || ^ || ^ || ^ || ^ || @ ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
| ^ || ^ || ^ || ^ ||   || s ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
| ^ || ^ || ^ ||   ||   ||   || = || ! ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
| ^ || ^ ||   ||   ||   ||   ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
| ^ ||   ||   ||   ||   ||   ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
|   ||   ||   ||   || ~ || ~ || ~ ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
|   ||   || & ||   || ~ || ~ || ~ ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
|   ||   ||   ||   || ~ || ~ || ~ || + ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
|   ||   ||   ||   ||   ||   ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
Enter command: w

Ведьмак: Хм.
Unit S get damage and die
Enter command for base 'c' - Create unit, 's' -
select unit: |

```

После смерти предыдущего был создан еще один юнит (В). Старый был автоматически стерт с карты:

```

  ---  ---  ---  ---  ---  ---  ---  ---
| ^ || ^ || ^ || ^ || ^ || @ ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
| ^ || ^ || ^ || ^ ||   ||   ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
| ^ || ^ || ^ ||   ||   ||   || = || ! ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
| ^ || ^ ||   ||   ||   ||   ||   || B ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
| ^ ||   ||   ||   ||   ||   ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
|   ||   ||   ||   || ~ || ~ || ~ ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
|   ||   || & ||   || ~ || ~ || ~ ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
|   ||   ||   ||   || ~ || ~ || ~ || + ||   |
  ---  ---  ---  ---  ---  ---  ---  ---
|   ||   ||   ||   ||   ||   ||   ||   ||   |
  ---  ---  ---  ---  ---  ---  ---  ---

```

### 3) Взаимодействие с одним из нейтральных объектов - костром.

```

  ^ | ^ | ^ | ^ | ^ | @ | | | |
  ^ | ^ | ^ | ^ | | | | | |
  ^ | ^ | ^ | | | | | = | ! | |
  ^ | ^ | | | | | | | | |
  ^ | | | | | | | | | |
  | | | B | | ~ | ~ | ~ | | | |
  | | | & | | ~ | ~ | ~ | | |
  | | | | | | ~ | ~ | ~ | + | |
  | | | | | | | | | | |

Enter command: s

Bonfire: Unit Bget rest. HP was increased by 10New
HP: 110

  ^ | ^ | ^ | ^ | ^ | @ | | | |
  ^ | ^ | ^ | ^ | | | | | |
  ^ | ^ | ^ | | | | | = | ! | |
  ^ | ^ | | | | | | | | |
  ^ | | | | | | | | | |
  | | | | | | ~ | ~ | ~ | | |
  | | | B | | ~ | ~ | ~ | | |
  | | | | | | ~ | ~ | ~ | + | |
  | | | | | | | | | | |

Enter command:
```

Повторное взаимодействие:

```

  ^ | ^ | ^ | ^ | ^ | @ | | | |
  ^ | ^ | ^ | ^ | | | | | |
  ^ | ^ | ^ | | | | | = | ! | |
  ^ | ^ | | | | | | | | |
  ^ | | | | | | | | | |
  | | | | | | ~ | ~ | ~ | | |
  | | | ~ | | ~ | ~ | ~ | | |
  | | | B | | ~ | ~ | ~ | + | |
  | | | | | | | | | | |

Enter command: w

Bonfire: the fire has already gone out

  ^ | ^ | ^ | ^ | ^ | @ | | | |
  ^ | ^ | ^ | ^ | | | | | |
  ^ | ^ | ^ | | | | | = | ! | |
  ^ | ^ | | | | | | | | |
  ^ | | | | | | | | | |
  | | | | | | ~ | ~ | ~ | | |
  | | | B | | ~ | ~ | ~ | | |
  | | | | | | ~ | ~ | ~ | + | |
  | | | | | | | | | | |

Enter command: |
```



### **Выводы.**

Были созданы новые классы: классы ландшафта, класс базы, классы нейтральных объектов, реализованы общие интерфейсы для групп классов. Был реализован паттерн наблюдатель.