

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

ОТЧЕТ
по лабораторной работе №3
по дисциплине «ООП»
Тема: Логическое разделение классов

Студент гр. 8383

Шишкин И.В.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2020

Цель работы.

Разработать и реализовать наборы классов для взаимодействия пользователя с юнитами и базой.

Постановка задачи.

Разработать и реализовать наборы классов для взаимодействия пользователя с юнитами и базой. Основные требования:

- Должен быть реализован функционал управления юнитами
- Должен быть реализован функционал управления базой

Ход работы.

Были реализованы: функционал управления юнитами и базой, паттерн «Фасад», через который пользователь управляет программой, паттерн «Посредник», через который юниты взаимодействуют между собой, паттерн «Команда» для передачи команд, паттерн «Цепочка обязанностей» для приема команд от пользователя.

Описание паттернов.

Взаимодействие пользователя с программой реализовано через паттерн «фасад» - класс Game (файлы Game.h, Game.cpp). В этом классе производится считывание всех необходимых данных, проверки на ошибки, если пользователь написал что-то неверно.

Для работы с классами поля и базы реализован паттерн «команда», классы которого работают по принципу «цепочка обязанностей» (файлы Command.h, Command.cpp):

Class Command – интерфейс команды;

Class CommandBase – команда для работы с методами базы. В конструктор передаются указатели на поле и на базу, и переменная, в которой содержится команда: что нужно сделать в базе. Для базы существуют 4 команды: createME – создание базы людей и эльфов, createO – создание базы

орков, addInME – добавление юнита в базу людей и эльфов, addInO – добавление юнита в базу орков.

Class CommandField – команда для управления юнитами на поле. В конструктор передаются указатель на поле и строка, в которой содержится команда. Для этого класса существуют 3 команды: add – добавление на поле определенного типа юнита, del – удаление юнита с поля, move – передвижение юнита по полю.

Так же был реализован паттерн «посредник» (файлы Mediator.h, Mediator.cpp). Через класс Mediator производится атака одного юнита на другого с помощью метода attack(Warrior w1, Warrior& w2).

Описание функционала управления юнитами.

В классе Game через CommandField производится управление юнитами. Пользователь может выбрать любого юнита, после чего программа произведет проверку с помощью метода класса Field string check(string& character): какие действия можно произвести этому юниту (на вход подается название юнита character, на выход строка из цифр – направления куда можно идти или атаковать юниту). Далее пользователь выбирает нужный ему вариант – вводит определенную цифру, затем, с помощью метода moving производится передвижение или атака юнита. При атаке так же вызывается метод unitsSpecAbil, который производит выполнение специальных возможностей юнитов (спец. возможности указаны в комментариях в файлах Elfs.h, Men.h, Orcs.h). Чтобы вывести информацию о юнитах (их здоровье, броню и урон), нужно после всех ходов написать в консоль “info” (программа предупреждает о том, когда это нужно написать).

Описание функционала управления базой.

В классе Game через CommandBase производится управление базой. Пользователь сначала должен создать базу (при этом, пока что пользователь может создавать только 1 базу за игру), написав в консоль “base”, а затем

“create”. Тогда на поле создастся база, помеченная “###”, в которой будет 2 юнита. Создание базы реализовано через метод класса Base createMEBase – для создания базы эльфов и людей и createOrcsBase – для создания базы орков. Так же пользователь может добавлять юнитов в базу, если написать в консоль “base”, а затем “add”. Добавление реализовано с помощью методов addInOrcsBase и addInMEBase класса Base. Так же, если написать в консоль “info”, то помимо данных о юнитах будет выведена информация о базах: созданы ли они, их здоровье, и информация о юнитах в базе. Вывод реализован через метод printBaseCondition класса Base.

Тестирование.

Как выглядит выбор движения:

```
1-й игрок, ваш ход:
Введите персонажа из эльфов или людей:
MS1
Введите направление:
Возможные варианты хода:
1 - вверх
2 - вправо
3 - влево
4 - вниз
5 - не двигаться
4
```

Вывод информации о юнитах и базе:

```
Введите "info", чтобы вывести информацию о юнитах и базе; Введите что угодно, чтобы ничего не вывести
info
Персонаж EA1 : здоровье - 50, броня - 12, урон - 10
Персонаж MW0 : здоровье - 25, броня - 50, урон - 3
Персонаж EI0 : здоровье - 50, броня - 12, урон - 10
Персонаж EA0 : здоровье - 50, броня - 12, урон - 10
Персонаж MS0 : здоровье - 25, броня - 50, урон - 10
Персонаж MS1 : здоровье - 25, броня - 50, урон - 10
Персонаж OD1 : здоровье - 100, броня - 8, урон - 10
Персонаж OR1 : здоровье - 100, броня - 8, урон - 10
Персонаж OR2 : здоровье - 100, броня - 8, урон - 10
Персонаж OD0 : здоровье - 100, броня - 8, урон - 10
Персонаж OR0 : здоровье - 100, броня - 8, урон - 10
Персонаж OD2 : здоровье - 100, броня - 8, урон - 10
Базы людей и эльфов нет
Базы орков нет
```

Создание базы орков:

```

2-й игрок, ваш ход:
Введите персонажа из орков:
base
Введите "create", чтобы создать базу, либо "add", чтобы добавить юнита в базу
create

```

~~~	EI0	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	\$\$\$	OD0
~~~	EA0	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	OD1	\$\$\$
EA1	~~~	/I\	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	+AR	OR0
~~~	MS0	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	OR1	\$\$\$
MW0	~~~	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	\$\$\$	OD2
~~~	~~~	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	OR2	\$\$\$
~~~	MS1	~~~	~~~	~~~	\$\$\$	###	###	###	###
???	???	???	???	???	???	###	???	???	???
???	???	+HP	???	???	???	###	???	***	OR4
???	???	???	???	???	???	###	???	???	OR3

Добавление юнита в базу орков:

```

2-й игрок, ваш ход:
Введите персонажа из орков:
base
Введите "create", чтобы создать базу, либо "add", чтобы добавить юнита в базу
add
Введите обозначение персонажа, которого вы хотите добавить:
OR

```

EI0	~~~	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	\$\$\$	OR0
~~~	EA0	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	OR1	\$\$\$
EA1	~~~	/I\	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	+AR	OD0
~~~	MS0	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	OD1	\$\$\$
MS1	~~~	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	\$\$\$	OD2
~~~	~~~	~~~	~~~	~~~	\$\$\$	\$\$\$	\$\$\$	OD3	\$\$\$
~~~	~~~	~~~	~~~	~~~	\$\$\$	###	###	###	###
???	???	???	???	???	???	###	OR4	???	???
???	MW0	+HP	???	???	???	###	???	***	OR3
???	???	???	???	???	???	###	???	???	OR2

Вывод информации для базы орков:

```

Введите "info", чтобы вывести информацию о юнитах и базе; Вв
info
Персонаж EI0 : здоровье - 50, броня - 12, урон - 10
Персонаж EA1 : здоровье - 50, броня - 12, урон - 10
Персонаж MS1 : здоровье - 25, броня - 50, урон - 10
Персонаж EA0 : здоровье - 50, броня - 12, урон - 10
Персонаж MS0 : здоровье - 25, броня - 50, урон - 10
Персонаж MW0 : здоровье - 25, броня - 0, урон - 3
Персонаж OR4 : здоровье - 100, броня - 0, урон - 10
Персонаж OR1 : здоровье - 100, броня - 8, урон - 10
Персонаж OD1 : здоровье - 100, броня - 8, урон - 10
Персонаж OD3 : здоровье - 100, броня - 8, урон - 10
Персонаж OR0 : здоровье - 100, броня - 8, урон - 10
Персонаж OD0 : здоровье - 100, броня - 8, урон - 10
Персонаж OD2 : здоровье - 100, броня - 8, урон - 10
Персонаж OR3 : здоровье - 100, броня - 0, урон - 10
Персонаж OR2 : здоровье - 100, броня - 0, урон - 10

Базы людей и эльфов нет
Здоровье базы орков: 110 HP
На базе 3 орков:
Персонаж на базе OR2: здоровье - 100, броня - 8, урон - 10
Персонаж на базе OR3: здоровье - 100, броня - 8, урон - 10
Персонаж на базе OR4: здоровье - 100, броня - 8, урон - 10

```

Атака за юнита MS0 на юнит OD4:

```

1-й игрок, ваш ход:
Введите персонажа из эльфов или людей:
MS0
Введите направление:
Возможные варианты хода:
1 - вверх
7 - атака юнита справа
3 - влево
4 - вниз
5 - не двигаться
7

```

```

info
Персонаж EI0 : здоровье - 50, броня - 12, урон - 10
Персонаж EA1 : здоровье - 50, броня - 12, урон - 10
Персонаж MS1 : здоровье - 25, броня - 50, урон - 10
Персонаж EA0 : здоровье - 50, броня - 12, урон - 10
Персонаж MS0 : здоровье - 25, броня - 50, урон - 10
Персонаж MW0 : здоровье - 25, броня - 0, урон - 3
Персонаж OD4 : здоровье - 90, броня - 8, урон - 10
Персонаж OR4 : здоровье - 100, броня - 0, урон - 10
Персонаж OR1 : здоровье - 100, броня - 8, урон - 10
Персонаж OD1 : здоровье - 100, броня - 8, урон - 10
Персонаж OD3 : здоровье - 100, броня - 8, урон - 10
Персонаж OR0 : здоровье - 100, броня - 8, урон - 10
Персонаж OD0 : здоровье - 100, броня - 8, урон - 10
Персонаж OD2 : здоровье - 100, броня - 8, урон - 10
Персонаж OR3 : здоровье - 100, броня - 0, урон - 10
Персонаж OR2 : здоровье - 100, броня - 0, урон - 10

```

### **Выводы.**

В ходе выполнения лабораторной работы были реализованы наборы классов для взаимодействия пользователя с юнитами и базой.