

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**МОЭВМ**

**ОТЧЕТ**  
**по практической работе № 6**  
**по дисциплине «ООП»**  
**Тема: Шаблонные классы**

Студент гр. 8383

Степанов В.Д.

Преподаватель

Жангиров Т. Р.

Санкт-Петербург

2020

### **Цель работы.**

Изучение и применение паттернов программирования для разработки игры на языке C++.

### **Постановка задачи.**

Разработка и реализация набора классов правил игры. Основные требования:

- Правила игры должны определять начальное состояние игры
- Правила игры должны определять условия выигрыша игроков
- Правила игры должны определять очередность ходов игрока
- Должна быть возможность начать новую игру

Выполнены основные требования
Реализован шаблонный класс игры, в качестве параметра шаблона передаются конкретные правила
Должно быть реализовано минимум 2 правил игры
*Класс игры в шаблоне поддерживает кол-во игроков. И для определенного кол-ва должен быть специализирован отдельно
*Передача хода между игроками реализована при помощи паттерна “Состояние”
*Класс игры один единственный и создается паттерном “Синглтон”

### **Ход работы.**

1. Был реализован класс игры, для того чтобы создать объект класса нужно казать размер игрового поля и тип игры (быстрая/небыстрая). Разница между режимами игры в том, что при быстрой игре здоровье и броня у юнитов меньше, а мощность атаки больше по сравнению со вторым режимом. Переход хода, как и первый ход определяются рандомно, с вероятностью 50% (т.е. вероятность что ход перейдет к сопернику равняется 50%). Для того, чтобы победить нужно убить юнита “Smoker” противника

(например: игрок 1 выиграет, когда убьет юнита “2\$”). Для того, чтобы завершить игру нужно ввести команду “exit”.

2. Передача хода между игроками реализована при помощи паттерна “Состояние”. Данный паттерн проверяет корректность данных введенной команды игрока, если все верно, то команда выполняется. Реализация представлена в файлах State.hpp, firstState.hpp, firstState.cpp, secondState.hpp, secondState.cpp, MainState.hpp, MainState.cpp.

3. Класс игры один единственный и создается паттерном “Синглтон”. Реализация представлена в файлах SingletonGame.cpp, SingletonGame.hpp.

### Пример работы программы.

Запустим программу и выберем быстрый режим и размер поля 9x9. Результат представлен на рисунке 1.

```
Быстрая игра y/n
y
Размер поля (6 < x < 21)
9
Запись логов в файл 'f' или в консоль 'с'. Без записи 'n'
n n
No
Загрузить сохранение y/n
Start
-----
| BA  ##  <1  B1|
| HP      !5  $1|
|           !3  <3|
|           WW  !1|
| ((  ((PC  PC((|
| 6!   WW  #####|
| 2>  4!  ((  ((|
|  2$  2!  ##  BA|
| B2  4>  ##  HP|
|-----|
$1  h = 10  a = 10  p = 3  2$  h = 10  a = 10  p = 3
<1  h = 5   a = 5   p = 6  2>  h = 5   a = 5   p = 6
<3  h = 5   a = 5   p = 6  4>  h = 5   a = 5   p = 6
!1  h = 8   a = 8   p = 7  2!  h = 8   a = 8   p = 7
!3  h = 8   a = 8   p = 7  4!  h = 8   a = 8   p = 7
!5  h = 8   a = 8   p = 7  6!  h = 8   a = 8   p = 7
Ходит игрок 1
```

Рисунок 1 – Запуск программы

В начале игры ход предоставлен игроку 1, попробуем передвинуть юнита игрока 2. Результат представлен на рисунке 2.

Начнем новую игру, вводим команду “exit” и выбираем начать игру. Результат представлен на рисунке 3.

```

Ходит игрок 1
wm 2!
Сейчас ходит другой игрок
Error motion
wm !1
-----
|  BA  ##    <1  B1|
|HP          !5  $1|
|          !3!1<3|
|          WW      |
|((  ((PC  PC((    |
|  6!    WW  #####|
|2>  4!  ((  ((    |
|  2$  2!   ##  BA|
|B2  4>   ##  HP|
|-----|
$1  h = 10  a = 10  p = 3  2$  h = 10  a = 10  p = 3
<1  h = 5   a = 5   p = 6  2>  h = 5   a = 5   p = 6
<3  h = 5   a = 5   p = 6  4>  h = 5   a = 5   p = 6
!1  h = 8   a = 8   p = 7  2!  h = 8   a = 8   p = 7
!3  h = 8   a = 8   p = 7  4!  h = 8   a = 8   p = 7
!5  h = 8   a = 8   p = 7  6!  h = 8   a = 8   p = 7
Ходит игрок 1

```

Рисунок 2 – Перемещение юнитов

### Вывод.

В ходе лабораторной работы были изучены и применены паттерны программирования для разработки игры на языке C++.

```

Ходит игрок 1
exit

-----
| BA  ##  <1  B1 |
| HP      !5  $1 |
|           !3!1<3 |
|           WW    |
| ((  ((PC  PC((  |
| 6!    WW  ####  |
| 2>  4!  ((  ((  |
| 2$  2!    ##  BA |
| B2  4>    ##  HP |
|-----|

$1  h = 10  a = 10  p = 3  2$  h = 10  a = 10  p = 3
<1  h = 5   a = 5   p = 6  2>  h = 5   a = 5   p = 6
<3  h = 5   a = 5   p = 6  4>  h = 5   a = 5   p = 6
!1  h = 8   a = 8   p = 7  2!  h = 8   a = 8   p = 7
!3  h = 8   a = 8   p = 7  4!  h = 8   a = 8   p = 7
!5  h = 8   a = 8   p = 7  6!  h = 8   a = 8   p = 7

Ходит игрок 1
Начать новую игру у/п
у
Быстрая игра у/п
п
Размер поля (6 < x < 21)
7
Запись логов в файл 'f' или в консоль 'с'. Без записи 'п'
п п
No
Загрузить сохранение у/п
Start

-----
| BA      <1  B1 |
| HP      !5  $1 |
| ((      WW!3  <3 |
| 6!PC  PC!1  |
| 2>  4!WW##  |
| 2$  2!    BA |
| B2  4>    HP |
|-----|

$1  h = 15  a = 15  p = 1  2$  h = 15  a = 15  p = 1
<1  h = 5   a = 10  p = 4  2>  h = 5   a = 10  p = 4
<3  h = 5   a = 10  p = 4  4>  h = 5   a = 10  p = 4
!1  h = 8   a = 13  p = 5  2!  h = 8   a = 13  p = 5
!3  h = 8   a = 13  p = 5  4!  h = 8   a = 13  p = 5
!5  h = 8   a = 13  p = 5  6!  h = 8   a = 13  p = 5

Ходит игрок 1

```

Рисунок 3 – Пример запуска новой игры