**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра МО ЭВМ**

**ОТЧЕТ**

**по лабораторной работе №1**

# по дисциплине «Объектно-Ориентированное Программирование»

# Тема: Создание классов, конструкторов и методов

|  |  |  |
| --- | --- | --- |
| Студент гр. 1384 |  | Усачева Д.В. |
| Преподаватель |  | Жангиров Т.Р. |
|  |  |  |

Санкт-Петербург 2022

**Цель работы.**

Целью работы является создание основных классов, налаживание их базового взаимодействия и проработка общей архитектуры проекта.

**Задание.**

Нужно реализовать прямоугольное игровое поле, состоящее из клеток. Клетка - элемент поля, которая может быть проходима или нет (определяет, куда может стать игрок), а также содержит какое-либо событие, которое срабатывает, когда игрок становится на клетку. Для игрового поля при создании должна быть возможность установить размер (количество клеток по вертикали и горизонтали). Игровое поле должно быть зациклено по вертикали и горизонтали, то есть если игрок находится на правой границе и идет вправо, то он оказывается на левой границе (аналогично для всех краев поля).

Также нужно реализовать класс игрока. Игрок - сущность, контролируемая пользователем. Игрок должен иметь свой набор характеристик и различный набор действий (например, разные способы перемещения, попытка избежать событие, и так далее).

# Требования.

## Реализовать класс игрового поля;

* Для игрового поля реализовать конструктор с возможностью задать размер по умолчанию (то есть конструктор, который можно вызвать без аргументов);

## Реализовать класс клетки с конструктором, позволяющим задать ей начальные параметры;

* Для клетки реализовать методы реагирования на то, что игрок перешел на клетку;

## Для клетки реализовать методы, позволяющие заменять событие (то есть клетка в ходе игры может динамически меняться);

* Реализовать конструкторы копирования и перемещения, и соответствующие им операторы присваивания для игрового поля и при необходимости клетки;

## Реализовать класс игрока минимум с 3 характеристиками и соответствующие ему конструкторы;

* Реализовать перемещение игрока по полю с проверкой допустимости на переход по клеткам;
* Игровое поле должно быть зациклено по вертикали и горизонтали.

# Формат выхода.

## При написании конструкторов нужно учитывать, что события должны храниться по указателю для соблюдения полиморфизма;

* Для управления игроком можно использовать медиатор, команду, цепочку обязанностей.

**Выполнение работы.**

1. Определяется класс *Player*, объектoм которого является игрок.

Класс Player описывает сущность игрока. У него есть три характеристики: здоровье, сила и очки маны. Пока что нет возможности изменять эти характеристики в процессе игры, так как нет в программе не заданы события. Был созданы конструктор класса и заданы методы класса, которые отвечают за получение данных о текущем состоянии игрока и методы для изменения этих количественных характеристик.

Прописать с чем связан класс

Реализуются методы класса с модификатором доступа *public*:

* + *explicit Player (int health = 100, int armour = 100, int damage = 10);* - конструктор класса *Player.*
  + *Player(int health = 100,*
  + *int mana\_points = 0, int damage = 10)*
  + *int get\_health() const*
  + *int get\_mana\_points() const*
  + *int get\_damage() const*
  + *void add\_health(int arg)*
  + *void add\_mana\_points(int arg)*
  + *void add\_damage(int arg)*

Инициализируются поля класса с модификатором доступа *private*:

* + *int health;* – здоровье игрока.
  + *int mana\_points;* – очки маны игрока.
  + *int coins;* – монеты игрока.
  + *int damage*; – урон игрока.

**Выводы.**

**ПРИЛОЖЕНИЕ 1**

**ИСХОДНЫЙ КОД ПРОГРАММЫ**