

ФГБОУ ВО Уральский государственный горный университет

Инженерно-экономический факультет Кафедра информатики

Курсовой проект

По дисциплине «Программирование»

На тему «Разработка приложения «Цифровой кузнечик»»

Выполнил:

Студент гр. ИНФ-xx Рукомойников В.А.

Проверила:

ст. преп. каф. информатики Волкова Е.А.

Екатеринбург, 2016г.

**Содержание**

[1. Постановка задачи 5](#_Toc167817181)

[1.1. Формулировка задания 5](#_Toc167817182)

[1.2. Алгоритмическое разрешение задачи 6](#_Toc167817183)

[1.3. Контрольные примеры 10](#_Toc167817184)

[2. Решение задачи 13](#_Toc167817185)

[2.1. Выбор средств реализации 13](#_Toc167817186)

[2.2. Описание основных классов 14](#_Toc167817187)

[2.3. Интерфейс приложения 19](#_Toc167817188)

[3. Тестирование приложения 23](#_Toc167817189)

[3.1. Контрольный робот 23](#_Toc167817190)

[3.2. Работа приложения на контрольных тестах 24](#_Toc167817191)

[3.3. Результаты работы 25](#_Toc167817192)

# Постановка задачи

## Формулировка задания

**«****Bomberman»**

### Написать программу реализующую игру «Bomberman». Пример игрового поля:

### Рисунок 1. Пример поля

**Постановка задачи**

## Написать Алгоритмическое разрешение задачи

### Для решения поставленной задачи используются разветвляющий и циклический алгоритмы.

### В данной задаче используется двумерный массив, элементами которого могут быть значения: 0,1,2,3,4,11,12:

### 0 - соответствует пустому полю;

### 1- Не разрушаемому блоку;

### 2- Разрушаемому блоку;

### 3- Бомбе.

### 4-центр взрыва

### 11-Игрок

### 12-Враг

### программу реализующую игру «Bomberman».

1.3. Контрольные примеры

• Проверка корректности движения игрового персонажа во всех направлениях.

• Проверка установки бомб и их взрыва, а также проверка реакции персонажа на взрывы (поражение).

• Тестирование взаимодействия персонажа с препятствиями, врагами и бонусами на игровом поле.

2. Решение задачи

2.1. Выбор средств реализации

В качестве языка программирования для решения этой задачи был выбран высокоуровневый язык JAVA, так как в нем много встроенных полезных функций. Также имеются коллекции, которые облегчают работу с программой. Как альтернативу JAVA можно было использовать C# или С++, с помощью которых тоже возможно решить поставленную задачу.

Java — объектно-ориентированный язык программирования, разработанный компанией Sun Microsystems (в последующем приобретённой компанией Oracle). Приложения Java обычно компилируются в специальный байт-код, поэтому они могут работать на любой виртуальной Java-машине (JVM) вне зависимости от компьютерной архитектуры.

Программы на Java транслируются в байт-код, выполняемый виртуальной машиной Java (JVM) — программой, обрабатывающей байтовый код и передающей инструкции оборудованию как интерпретатор. Достоинством подобного способа выполнения программ является полная независимость байт-кода от операционной системы и оборудования, что позволяет выполнять Java-приложения на любом устройстве, для которого существует соответствующая виртуальная машина. Другой важной особенностью технологии Java является гибкая система безопасности благодаря тому, что исполнение программы полностью контролируется виртуальной машиной. Любые операции, которые превышают установленные полномочия программы (например, попытка несанкционированного доступа к данным или соединения с другим компьютером) вызывают немедленное прерывание.

В качестве IDE для разработки был использован IntelliJ IDEA. По качеству и возможностям. IntelliJ IDEA является одним из лучших, крупных и комерчески успешных IDE, поддерживая рефакторинг, профилирование, выделение синтаксических конструкций цветом, автодополнение набираемых конструкций на лету, множество предопределённых шаблонов кода и др..

2.2. Описание основных классов

При решении задачи было разработано 10 классов показанны ниже (рисунок 5).

Рисунок 5. Диаграмма классов

Описание основных классов

Player: Класс, представляющий игрового персонажа. Отвечает за его движение, управление и отрисовку на экране.

Enemy: Класс, представляющий врагов на игровом поле. Реализует их движение, поведение и отрисовку.

Map: Класс, отвечающий за генерацию и отрисовку игрового поля, размещение препятствий и врагов.

Arena: Класс, определяющий размеры и параметры игрового поля.

Keyhandler: Класс, обрабатывающий нажатия клавиш и управляющий движением игровых персонажей.

Father: Абстрактный класс, от которого наследуются Player и Enemy. Содержит общие свойства и методы.

2.3. Интерфейс приложения

Интерфейс приложения прост в использовании. Графический интерфейс приложения позволяет игроку взаимодействовать с игровым миром. Управление осуществляется с помощью клавиатуры, а отображение игровых объектов происходит на экране.

3. Результат работы

3.1. Результаты работы

Результатом работы является полноценное приложение, позволяющее играть в игру "Bomberman". Приложение реализует основные механи