

УТВЕРЖДАЮ

Руководитель практики от Университета

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(должность, ФИО)*

МП \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(Подпись)*

Дата « \_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 202\_ год

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

ПМ.05 Проектирование и разработка информационных систем

|  |  |
| --- | --- |
| Чувашев Даниил Алексеевич | |
| *(Ф.И.О. обучающегося)* | |
| 09.02.07 Информационные системы и программирование | |
| *(специальность)* | |
|  | |
| Учебная группа | ИСПк-202-52-00 |
|  |  |
| Место прохождения практики | ФГБОУ ВО «Вятский государственный университет», |
| Колледж ВятГУ | *(наименование организации, структурного подразделения организации)* |
|  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Итоговая оценка: |  | | | | |
| Руководитель  практики от университета |  |  |  |  |  |
|  | *(дата)* |  | *(подпись)* |  | *(Ф.И.О.)* |

Киров, 2023 г.



ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ НА ПРАКТИКУ

|  |  |
| --- | --- |
| Ф.И.О. обучающегося | Порубов Кирилл Александрович |
| Специальность | 09.02.07 Информационные системы и программирование |
| Учебная группа | ИСПк-202-52-00 |
| Вид практики | учебная практика |

Сроки прохождения практики с 11.01.2023 по 16.06.2023

Место прохождения практики

ФГБОУ ВО «Вятский государственный университет», Колледж ВятГУ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | (наименование организации, структурного подразделения организации) | | |
| № | Виды работ, выполняемых обучающимися во время практики | | Объем работ (час) | Формируемые компетенции | |
| 1 | Пройти инструктаж по ознакомлению с правилами внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологическими правилами и гигиеническими нормативами, а также вводный инструктаж и инструктаж на рабочем месте | | 2 | - | |
| 2 | Постановка задачи | | 8 | ОК2, ОК6, ОК7, ОК9 | |
| 3 | Настройка рабочего окружения | | 8 | ОК1, ОК8, ОК9 | |
| 4 | Анализ предметной области, постановка расширенного технического задания | | 10 | ПК5.1, ОК3, ОК4 | |
| 5 | Проектирование | | 35 | ПК5.2, ПК5.3, | |
| 6 | Реализация программного кода | | 35 | ПК5.4, ПК5.8, | |
| 7 | Тестирование и отладка полученного кода | | 22 | ПК5.5, ПК6.4, ОК2, ОК9 | |
| 8 | Подготовка отчетной документации, включая детальное описание разработанных и использованных алгоритмов | | 10 | ПК5.6, ПК5.7, ОК9, ОК10, ОК11 | |
| 9 | Подготовка к сдаче и сдача промежуточной аттестации | | 2 | ОК4, ОК5 | |

|  |  |  |  |
| --- | --- | --- | --- |
| Индивидуальное задание на практику разработано в соответствии с рабочей программой практики. | | | |
| Руководитель  практики от университета |  |  |  | |  |  |
|  | (дата) |  | (подпись) | |  | (Ф.И.О.) |
| С индивидуальным заданием ознакомлен(а) |  |
|  | (дата, подпись обучающегося) |

**ОТЗЫВ РУКОВОДИТЕЛЯ ПРАКТИКИ**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ф.И.О. обучающегося | | Чувашев Даниил Алексеевич | | | |
| Специальность | | 09.02.07 Информационные системы и программирование | | | |
| Учебная группа | | ИСПк-202-52-00 | | | |
| Вид практики | | учебная практика | | | |
| Сроки прохождения практики с | | | 11.01.2023 | по | 16.06.2023 |
| Место прохождения практики | | | ФГБОУ ВО «Вятский государственный университет», Колледж ВятГУ | | |
|  | (наименование организации, структурного подразделения организации) | | | | |

ВИДЫ И КАЧЕСТВО ВЫПОЛНЕНИЯ РАБОТ

|  |  |  |  |
| --- | --- | --- | --- |
| Вид работ | Критерий выполнения работ | | |
| Выполнены полностью самостоятельно | Выполнены с незначительной помощью наставника | Выполнены с помощью наставника |
| Постановка задачи | V |  |  |
| Настройка рабочего окружения | V |  |  |
| Анализ предметной области, постановка расширенного технического задания | V |  |  |
| Проектирование | V |  |  |
| Реализация программного кода | V |  |  |
| Тестирование и отладка полученного кода | V |  |  |
| Подготовка отчетной документации, включая детальное описание разработанных и использованных алгоритмов | V |  |  |
| Подготовка к сдаче и сдача промежуточной аттестации | V |  |  |

Обучающийся ознакомлен с правилами внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологическими правилами и гигиеническими нормативами, а также прошел вводный инструктаж и инструктаж на рабочем месте.

Во время прохождения учебной практики обучающимся освоены следующие профессиональные и общие компетенции:

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование компетенции | Показатели оценки | Оценка | |
| Освоена | Не освоена |
| ПК 5.1. Собирать исходные данные для разработки проектной документации на информационную систему. | Способен выполнять анализ предметной области, выявляя существенные элементы, оказывающие влияние на проектируемую систему | V |  |
| ПК 5.2. Разрабатывать проектную документацию на разработку информационной системы в соответствии с требованиями заказчика. | Способен выполнять декомпозицию сущностей с целью получения наиболее полной картины о целесообразной структуре разработки | V |  |
| ПК 5.3. Разрабатывать подсистемы безопасной информационной системы в соответствии с техническим заданием. | Способен прогнозировать потенциально некорректные действия пользователя и предусматривать соответствующие реакции со стороны системы | V |  |
| ПК 5.4. Производить разработку модулей информационной системы в соответствии с техническим заданием. | Способен создавать программный код, отвечающий предъявляемым требованиям | V |  |
| ПК 5.5. Осуществлять тестирование информационной системы на этапе опытной эксплуатации с фиксацией выявленных ошибок кодирования в разрабатываемых модулях информационной системы. | Способен выполнять оценку корректности функционирования системы | V |  |
| ПК 5.6. Разрабатывать техническую документацию на эксплуатацию информационной системы. | Способен разрабатывать проектную, техническую и пользовательскую документации | V |  |
| ПК 5.7. Производить оценку информационной системы для выявления возможности ее модернизации. | Способен оценивать перспективы дальнейшего развития программной системы | V |  |
| ПК 5.8. Осуществлять интеграцию информационной системы с другими программными продуктами. | Способен повторно использовать готовые шаблонные решения при разработке программного продукта | V |  |
| ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам. | Способен анализировать специализированную информацию и находить оптимальные пути решения задач профессиональной деятельности | V |  |
| ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности. | Способен анализировать специализированную информацию и находить оптимальные пути решения задач профессиональной деятельности | V |  |
| ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие. | Способен находить пути улучшения имеющихся решений, позволяющих повысить их общий качественный уровень | V |  |
| ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами. | Способен конструктивно обмениваться информацией с коллегами, грамотно формулировать запросы в целях получения разъясняющей информации | V |  |
| ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста. | Способен конструктивно обмениваться информацией с коллегами, грамотно формулировать запросы в целях получения разъясняющей информации | V |  |
| ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей. | Способен конструктивно обмениваться информацией с коллегами, грамотно формулировать запросы в целях получения разъясняющей информации | V |  |
| ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях. | Способен эффективно использовать компьютерное время, а также материальные ресурсы, необходимые для решения поставленных задач | V |  |
| ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности. | Способен соблюдать требования внутреннего трудового распорядка организации, охраны труда и техники безопасности в целях сохранения собственного здоровья | V |  |
| ОК 09. Использовать информационные технологии в профессиональной деятельности. | Способен применять стек современных средств разработки ПО для решения задач профессиональной деятельности в заданном контексте | V |  |
| ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке. | Способен использовать в своей работе специализированную документацию | V |  |
| ОК 11. Планировать предпринимательскую деятельность в профессиональной сфере. | Способен проектировать алгоритмические решения, принимая во внимание имеющиеся ресурсные ограничения | V |  |

**Краткая характеристика работы обучающегося**

|  |
| --- |
| Программа практики выполнена в полном объеме. Все виды работ выполнялись в срок, |
| без существенных замечаний. В достаточной степени была проявлена самостоятельность |
| и умение грамотно пользоваться источниками информации, находящимися в свободном |
| доступе. |

|  |  |  |
| --- | --- | --- |
|  |  | Руководитель практики  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Подпись ФИО  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (должность)  Дата «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 202\_\_ год |

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc136702139)

[**1. СВЕДЕНИЯ О РАБОТЕ, ВЫПОЛНЕННОЙ В ПЕРИОД ПРОХОЖДЕНИЯ УЧЕБНОЙ ПРАКТИКИ** 4](#_Toc136702140)

[**2. НАСТРОЙКА РАБОЧЕГО ОКРУЖЕНИЯ** 5](#_Toc136702141)

[**3. ФОРМУЛИРОВКА ИНДИВИДУАЛЬНОГО ЗАДАНИЯ** 8](#_Toc136702142)

[**4. ОПИСАНИЕ ВЫПОЛНЕНИЯ ИНДИВИДУАЛЬНОГО ЗАДАНИЯ** 9](#_Toc136702143)

[**4.1 Анализ предметной области и обзор аналогов** 9](#_Toc136702144)

[**4.1.1 Анализ предметной области** 9](#_Toc136702145)

[**4.1.2 Обзор аналогов** 11](#_Toc136702146)

[**4.2 Разработка алгоритмов функционирования и структуры программы** 17](#_Toc136702147)

[**4.3 Реализация программы. Разработка пользовательского интерфейса** 24](#_Toc136702148)

[**4.3.1 Разработка программного обеспечения** 24](#_Toc136702149)

[**4.3.2 Разработка пользовательского интерфейса** 26](#_Toc136702150)

[**4.4 Тестирование программного обеспечения** 29](#_Toc136702151)

[**ЗАКЛЮЧЕНИЕ** 33](#_Toc136702152)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 34](#_Toc136702153)

[**ПРИЛОЖЕНИЕ А** 35](#_Toc136702154)

[**ПРИЛОЖЕНИЕ Б** 52](#_Toc136702155)

[**ПРИЛОЖЕНИЕ В** 53](#_Toc136702156)

[**ПРИЛОЖЕНИЕ Г** 54](#_Toc136702157)

# **ВВЕДЕНИЕ**

Учебная практика ПМ.05 проходила на базе Колледжа ФГБОУ ВО «Вятский государственный университет» в период с 11.01.2023 г. по 16.06.2023 г. по субботам.

Цель практики: сформировать у обучающихся представление о процессе разработки прикладного программного обеспечения.

Задачи практики:

– дать представление о различных методах проектирования прикладного программного обеспечения;

– сформировать навыки написания документации, требуемой при разработке прикладного программного обеспечения;

– дать представление о различных инструментах, используемых при разработке прикладного программного обеспечения.

# **1. СВЕДЕНИЯ О РАБОТЕ, ВЫПОЛНЕННОЙ В ПЕРИОД ПРОХОЖДЕНИЯ УЧЕБНОЙ ПРАКТИКИ**

В период с 11.01.2023 по 16.06.2023 при прохождении учебной практики ПМ.05 на базе Колледжа ФГОБУ ВО «Вятский государственный университет» был выполнен следующий перечень работ, представленный в таблице 1.

Таблица 1 – Сведения о работе, выполненной в период практики

| Дата | Краткое содержание выполненных работ |
| --- | --- |
| 11.01.2023 | Ознакомление с правилами внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологическими правилами и гигиеническими нормативами, а также вводный инструктаж и инструктаж на рабочем месте |
| 12.01.2023-25.01.2023 | Постановка задачи |
| 26.01.2023-08.02.2023 | Настройка рабочего окружения |
| 09.02.2023-22.02.2023 | Анализ предметной области, постановка расширенного технического задания |
| 23.02.2023-29.03.2023 | Проектирование |
| 30.03.2023-03.05.2023 | Реализация программного кода |
| 04.05.2023-24.05.2023 | Тестирование и отладка полученного кода |
| 25.05.2023-13.06.2023 | Подготовка отчетной документации, включая детальное описание разработанных и использованных алгоритмов |
| 14.06.2023-16.06.2023 | Подготовка к сдаче и сдача промежуточной аттестации |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата) (подпись)

# **2. НАСТРОЙКА РАБОЧЕГО ОКРУЖЕНИЯ**

Использование Python является одним из наиболее популярных выборов для разработки прикладных приложений и WEB-сервисов в настоящее время. Это современный и универсальный язык программирования, который отличается высоким уровнем производительности и структурированным, легко читаемым кодом. Python имеет очень простой и понятный синтаксис, который легко усваивается и позволяет начинающим программистам быстро освоить этот язык.

1) Описание языка.

Одним из ключевых преимуществ Python является широкий перечень встроенных библиотек, которые предоставляют разработчикам множество полезных функций и возможностей для решения различных задач. Это делает Python удобным выбором для написания приложений любого уровня сложности, а также для создания WEB-сервисов.

Python также поддерживает различные стили программирования, включая объектно-ориентированное и функциональное программирование. Это делает язык удобным выбором для разработчиков, желающих использовать разные подходы к написанию своих программ.

PyCharm, в свою очередь, является кроссплатформенной интегрированной средой разработки, которая позволяет удобно создавать и отлаживать программы на языке Python. Благодаря мощному набору инструментов и средств, PyCharm позволяет автоматизировать множество задач и значительно ускорить процесс разработки.

2) PyQt5 - мощная библиотека для создания графических интерфейсов на Python. Она обладает обширными возможностями, кросс-платформенностью, простотой использования, хорошей интеграцией с Python и поддержкой активного разработческого сообщества.

Скриншоты работы в выбранной среде приведены на рисунках ниже

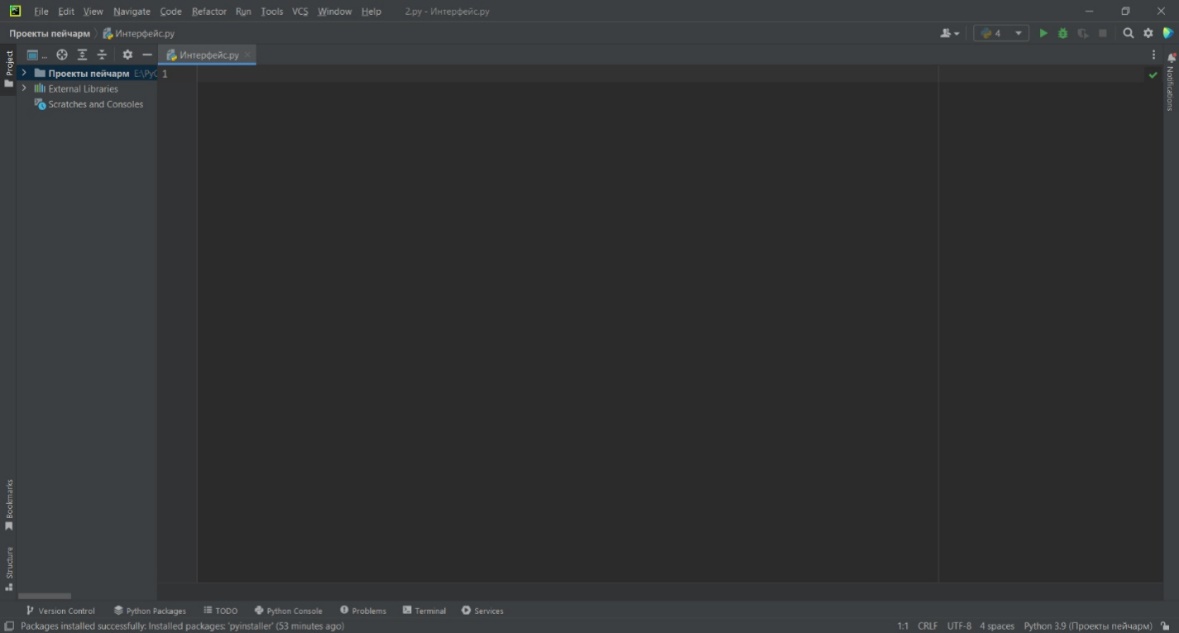


Рисунок 1. Интерфейс PyCharm Community Edition

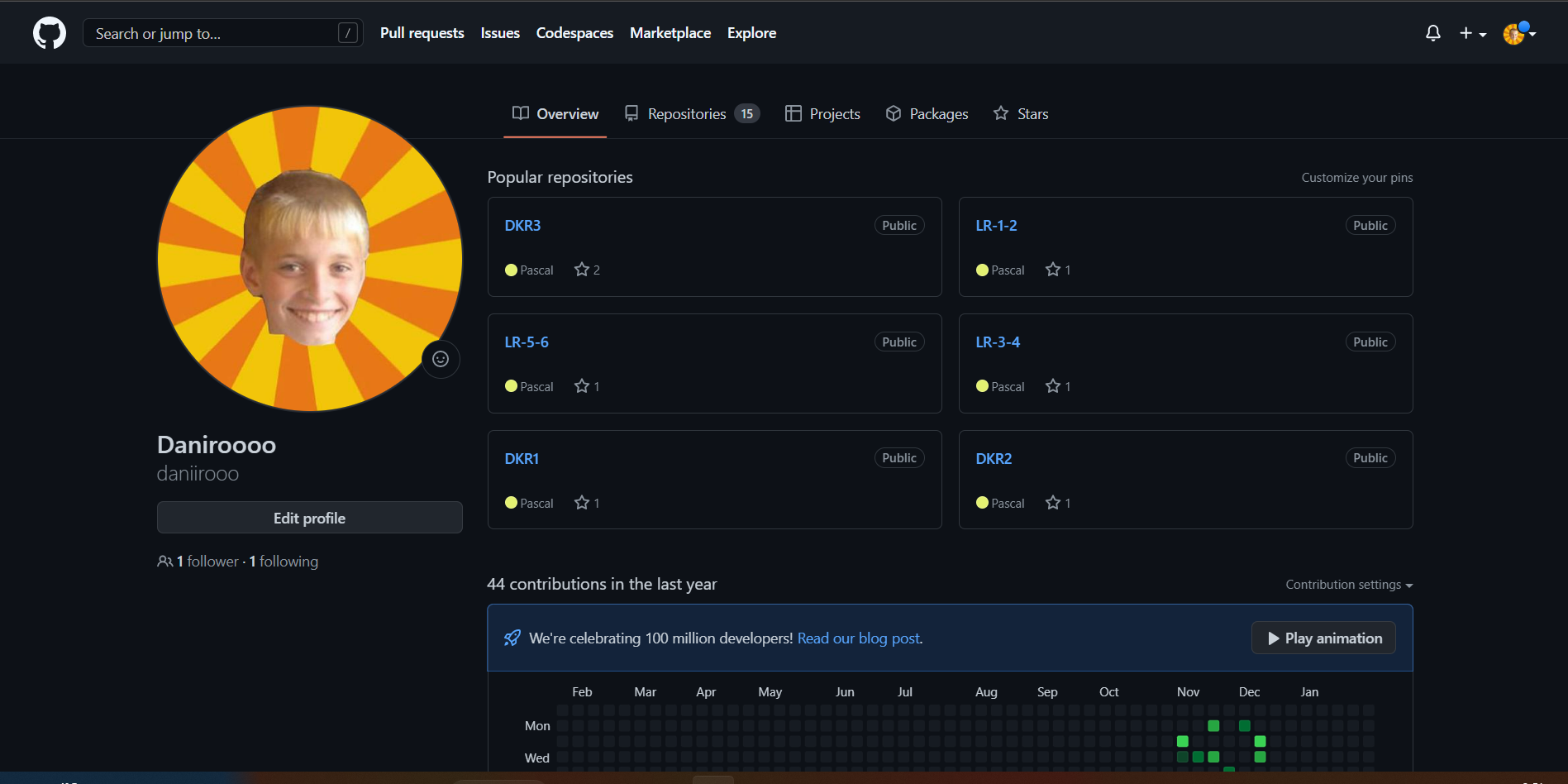
****

Рисунок 2 – Работа в github.

Ссылка на GitHub: <https://github.com/daniirooo>

Ссылка на репозиторий: <https://github.com/daniirooo/UP>

# **3. ФОРМУЛИРОВКА ИНДИВИДУАЛЬНОГО ЗАДАНИЯ**

В соответствии с индивидуальным заданием на учебную практику необходимо:

* Проанализировать предметную область
* Реализовать программный код приложения, реализующего игровой процесс игры «Арканоид с платформой»
* Провести тестирование и отладку полученного кода
* Подготовка отчетной документации, а также детальное описание разработанных и использованных алгоритмов
* Подготовить отчет к сдаче

# **4. ОПИСАНИЕ ВЫПОЛНЕНИЯ ИНДИВИДУАЛЬНОГО ЗАДАНИЯ**

## **4.1 Анализ предметной области и обзор аналогов**

### **4.1.1 Анализ предметной области**

**Арканоид с платформой** — это компьютерная игра, в которой игрок управляет платформой, которая отбивает мячик и разбивает блоки на экране. Игра была выпущена в 1986 году японской компанией Taito.

Идея игры основана на настольной игре «пинбол», где мячик отбивается от платформы и попадает в карманы на игровом поле. В «Арканоиде» игрок управляет платформой, которая двигается горизонтально, и отбивает мячик, который разрушает блоки на экране. Игра состоит из нескольких уровней, на каждом из которых уровень сложности увеличивается, и количество блоков, которые нужно разбить, увеличивается.

«Арканоид» был портирован на множество различных устройств. Игра была доступна на различных игровых консолях, компьютерах и мобильных устройствах. Были выпущены множество версий.

**Правила игры:**

Арканоид с платформой - компьютерная игра, в которой игрок управляет платформой, расположенной в нижней части экрана, и отбивает мячик, летящий сверху. На экране расположены кирпичи, которые нужно разрушить мячиком, отбивая его от платформы. Когда мячик касается кирпича, последний разрушается, а игрок получает очки. Цель игры - разрушить все кирпичи на экране и заработать максимальное количество очков. Мячик летит по прямой, отскакивая от стен и кирпичей. Игрок может управлять платформой, двигая ее по горизонтали, чтобы отбивать мячик в нужном направлении. Если мячик коснется нижней части экрана, игрок проигрывает. Игра имеет три уровня сложности:

**Разновидности**

На данный момент можно выделить такие разновидности тетриса:

* Assault (1988 video game)» Tetris»
* «Pang Adventures»
* «Reaxxion»

### **4.1.2 Обзор аналогов**

Первый аналог – это Игра «Assault (1988 video game)»

<https://vgtimes.ru/games/arkanoid/#game_description>



Рисунок 3. Игровой процесс Assault

Аркада с видом от третьего лица с элементами экшена и шутера

Управляя танком игроку, предстоит пройти каждый из одиннадцати этапов игры, стреляя по врагам и избегая снарядов.

**Достоинства:**

* Классическая механика игры: Арканоид Assault основана на классической игре Арканоид, что позволяет игрокам насладиться знакомым и привычным геймплеем.
* Множество разнообразных блоков: В игре присутствует множество различных типов блоков с разными свойствами
* Режим многопользовательской игры: Игра также предлагает режим многопользовательской игры, где вы можете соревноваться и играть с друзьями или другими игроками онлайн.

**Недостатки:**

* Отсутствие дополнительных режимов игры: Арканоид Assault может ограничиваться только основным режимом игры, без дополнительных режимов или вариаций

Второй аналог -<https://vgtimes.ru/games/pang-adventures/#game_description> Игра « Pang Adventures »

Рисунок 4. Геймплей игры Pang Adventure



Pang Adventures - аркадный экшен с видом сбоку. Игра продолжает историю Buster Bros., вышедшая ещё в 90-ых. Игроку предстоит с помощью разнообразного оружия разбивать огромные пузыри и уворачиваться от них, пока те окончательно не исчезнут.

Достоинства:

* Увлекательный геймплей
* Разнообразие уровней и мест: Pang Adventures предлагает множество разнообразных уровней и локаций для исследования.
* Мультиплеерный режим: Pang Adventures предлагает возможность играть в кооперативном режиме с другим игроком.

Недостатки:

* Повторяющийся геймплей: В ходе игры могут возникать ощущение повторения, особенно при длительном прохождении
* Отсутствие значимого сюжета: Игра Pang Adventures не предлагает глубокого сюжета или захватывающей истории

Третий аналог - [https://vgtimes.ru/games/reaxxion/](https://vgtimes.ru/games/reaxxion/%20) - Игра «Reaxxion»



Рисунок 5. Гэймплей игры Reaxxion

Достоинства:

* Инновационная механика: "Reaxxion" добавляет новые элементы в классическую формулу Арканоида. Вместо простого отскакивания шарика, игроку предоставляется возможность контролировать платформу, которая может двигаться во всех направлениях

Недостатки:

* Ограниченный игровой контент: Игра может иметь ограниченное количество уровней и вызовов

## **4.2 Разработка алгоритмов функционирования и структуры программы**

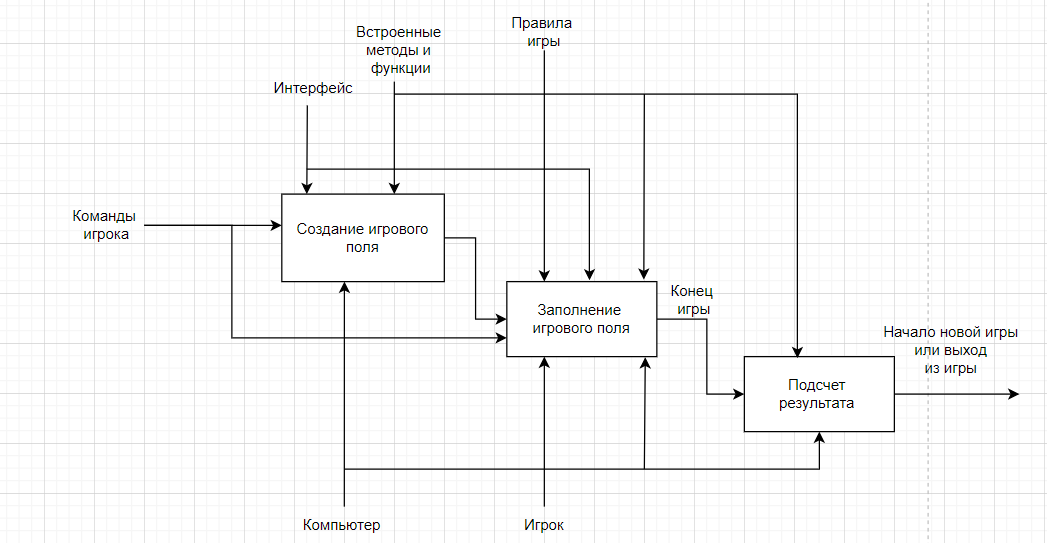


Рисунок 6. Диаграмма IEDF0

Входными данными являются команды пользователя. Элементами управления являются интерфейс, встроенные методы и функции правила игры. Для разработки приложения используются встроенные библиотеки Python tkinter. Выходными данными являются подсчет результатов и конец, а так же закрытие программы или начало новой игры. В процессе использования приложения задействованы пользователь и компьютер.

Алгоритм работы игры «Арканоид»:

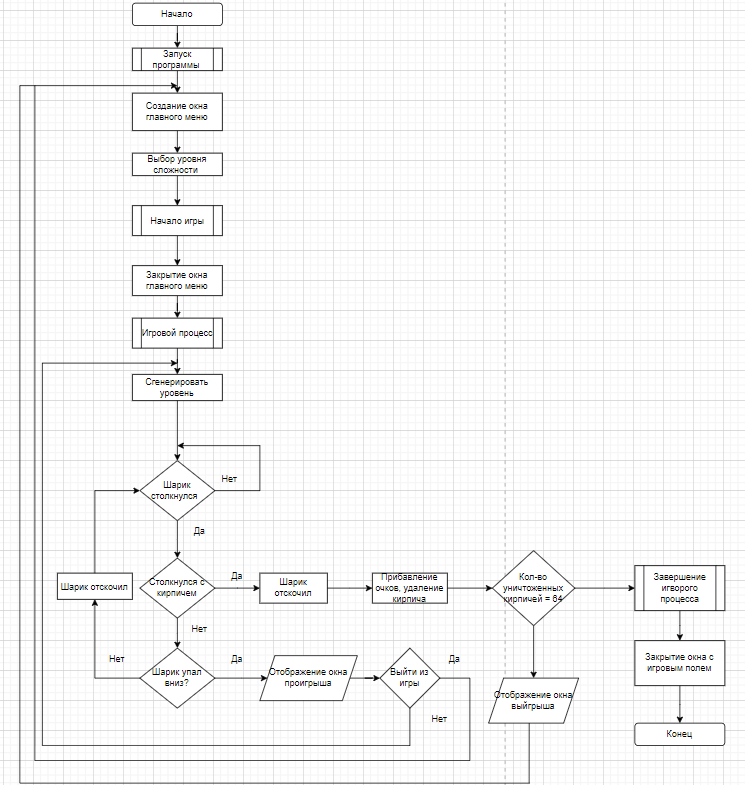


Рисунок 7 – Алгоритм работы игры «Арканоид»

Этап запуска программы включает в себя открытие игрового окна с основным интерфейсом, где игрок может выбрать сначала уровень сложности, а потом начать игру, поставить на паузу или завершить игру. В начале игры, игровом поле видны разноцветные кирпичики, которые игрок должен уничтожить при помощи мячика, которые отскакивает от платформы, которая находится внизу игрового поля. Мячик начинает свое движение при помощи отскакивания от платформы, для того чтобы уничтожить кирпичики сверху. Цель игрока в игре Арканоид с платформой заключается в уничтожении всех блоков, расположенных в верхней части экрана, с помощью мячика, отскакивающего от платформы, управляемой игроком. Платформа может двигаться только влево и вправо, а мячик должен отскакивать от нее и разрушать блоки. Игрок должен управлять платформой, чтобы не допустить падения мячика за пределы экрана и потери жизни. Если игрок уничтожает все кирпичики, то наступает победа.

После завершения игры игрок может увидеть свои результаты, такие как количество очков.

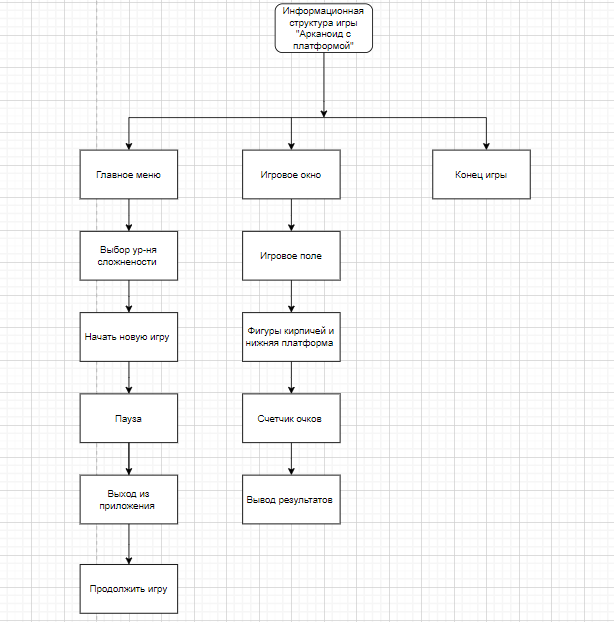


Рисунок 8 – Информационная структура приложения

Главное меню: это начальный экран игры, где игрок может выбрать опции для начала игры, доступа к настройкам или просмотра рекордов.

Начать игру: этот пункт позволяет игроку начать новую игру.

Пауза: кнопка, которая позволяет игроку приостановить игру в любой момент.

Продолжить игру: кнопка, которая позволяет игроку продолжить игру после паузы.

Выход: кнопка, позволяющая игроку выйти из игры и вернуться в главное меню.

Игровое окно: это область экрана, где отображается сама игра.

Игровое поле: это пространство на игровом экране, где игрок должен собирать фигуры тетрамино в горизонтальные линии.

Фигуры полимино: это набор различных фигур, состоящих из 4 квадратов, которые игрок должен расположить таким образом, чтобы собрать горизонтальные линии и заработать очки.

Счетчик очков: это счетчик, который отслеживает количество очков, набранных игроком.

Конец игры: это состояние, когда игрок не может больше разместить фигуры на игровом поле, и игра заканчивается.

Окно результатов: это экран, на котором отображаются результаты игрока после окончания игры.

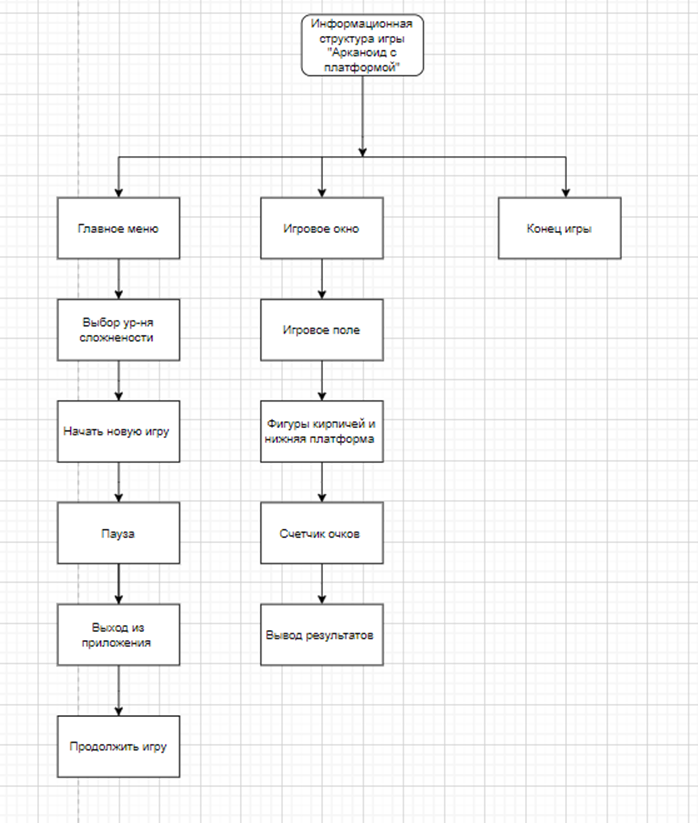


Рисунок 9 – Модульная структура приложения

Модуль игры:

игровое поле: здесь отображается игровое поле, на котором игрок управляет фигурами тетрамино.

фигуры полимино: набор различных геометрических фигур, которые игрок должен расставить на игровом поле.

счетчик очков: здесь отображается количество очков, которые заработал игрок.

конец игры: экран, который появляется при проигрыше игрока.

окно результатов: экран, который появляется после окончания игры и отображает результаты игрока.

Начать игру: кнопка, которая позволяет игроку начать игру.

Пауза: кнопка, позволяет остановить игру.

продолжение игры: кнопка, позволяет возобновить игру.

выход: кнопка, которая закрывает приложение.

Figma - это онлайн-сервис для разработки интерфейсов и прототипирования, который позволяет работать в режиме реального времени с возможностью организации совместной работы. С помощью этого сервиса был создан прототип игры "Тетрис", который представляет собой визуальное представление интерфейса и игровой механики. Прототип позволяет проводить тестирование и доработку дизайна игры до и во время разработки. Этот проект является ценным инструментом для разработки игры.

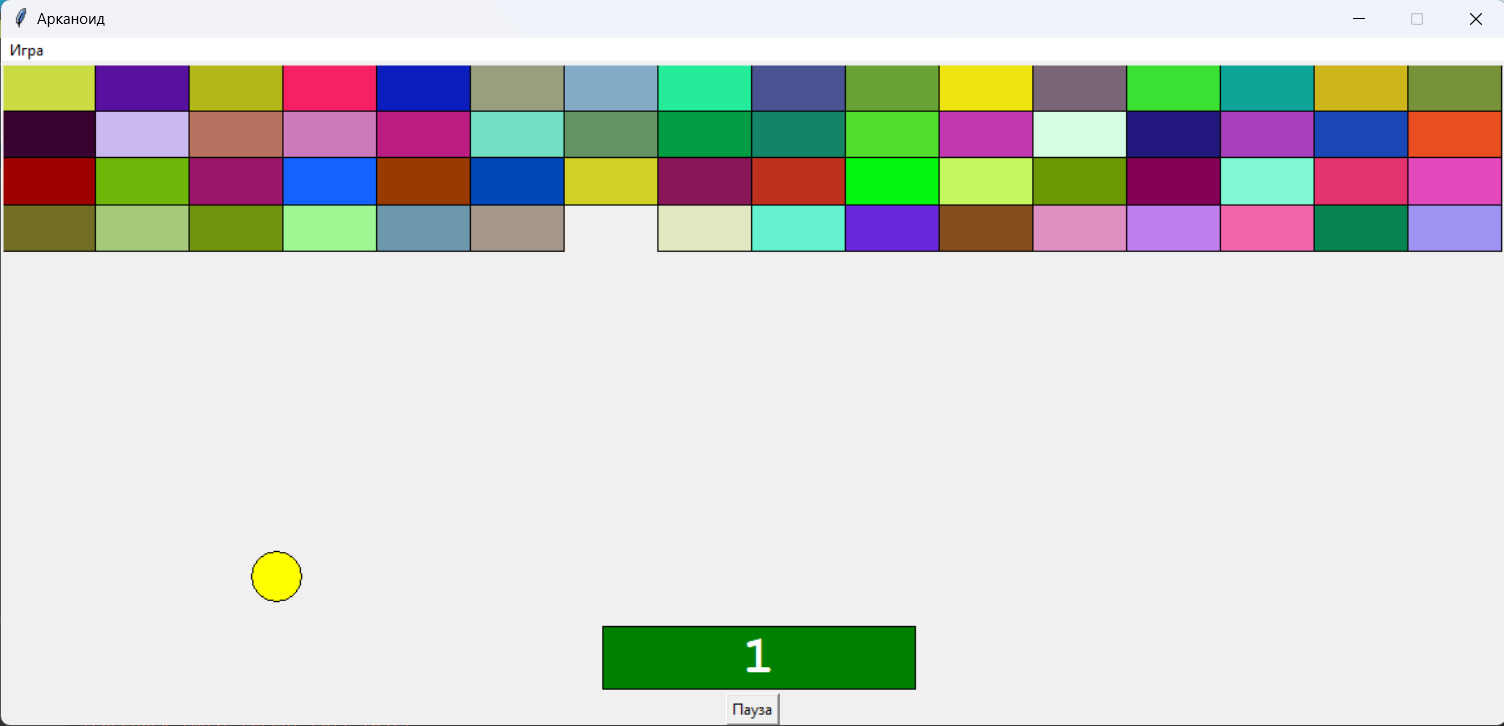


Рисунок 10 – Прототип игры «Арканоид»

## **4.3 Реализация программы. Разработка пользовательского интерфейса**

В этом разделе будут выполнены следующие задачи: создание алгоритмов для работы программы, разработка пользовательского интерфейса и внедрение основных функциональных возможностей программы.

## **4.3.1 Разработка программного обеспечения**

Программное обеспечение разработано в среде программирования PyCharm на языке Python. ПО состоит из двух частей: интерфейс и логика игры.

Функция отрисовки «Кирпичей» на игровом поле представлена на рисунке 20.

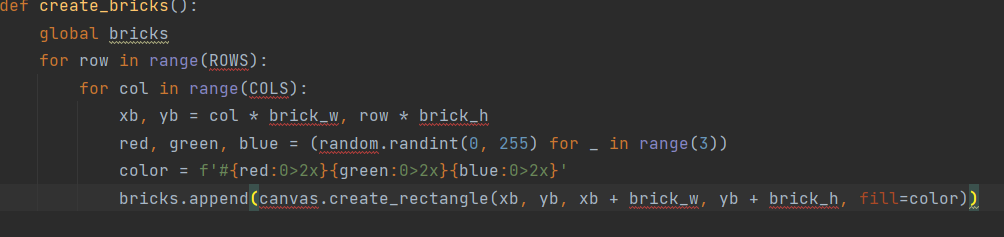


Рисунок 11. Функция отрисовки «Кирипчей» на игровом поле

Функция движения платформы и мячика на рисунках 12 и 13



Рисунок 12. Функция движения мячика

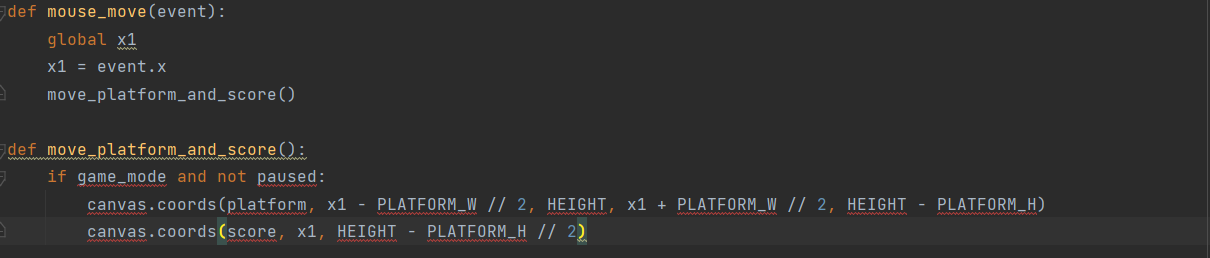


Рисунок 13. Функция движения платформы при помощи компьютерной мышки

### **4.3.2 Разработка пользовательского интерфейса**

Интерфейс игры тетрис выполнен в простых и понятных формах. В окне игры расположены заголовок «Арканоид» .Также присутствуют три кнопки: "Выход из игры", "Новая игра", "Открыть пояснения"

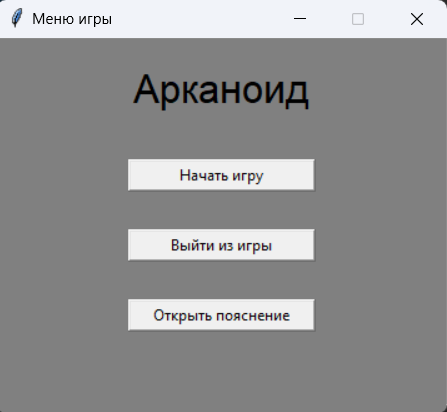


Рисунок 14. Интерфейс главного меню

При нажатии кнопки «Начать игру» будет показано окно с выбором уровня сложности при помощи кнопок

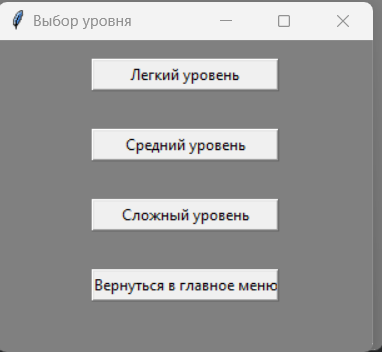


Рисунок 15. Окно выбора сложности

Кнопка «Правила» открывается окно с правилами игры и выходом из игры

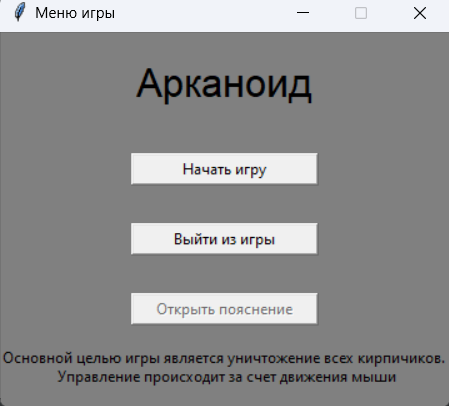


Рисунок 16. окно с правилами игры и выходом из игры

Кнопка «Пауза» ставит игровой процесс на паузу и если нажать еще раз на кнопку, игра продолжится.



Рисунок 17. Кнопка паузы (под платформой)

На Рисунке 18 представлено окно, которое появляется при выигрыше. Будет написана слово «Victiry», что означает «Победа». Также появятся две кнопки «Выход» и «Новая игра»

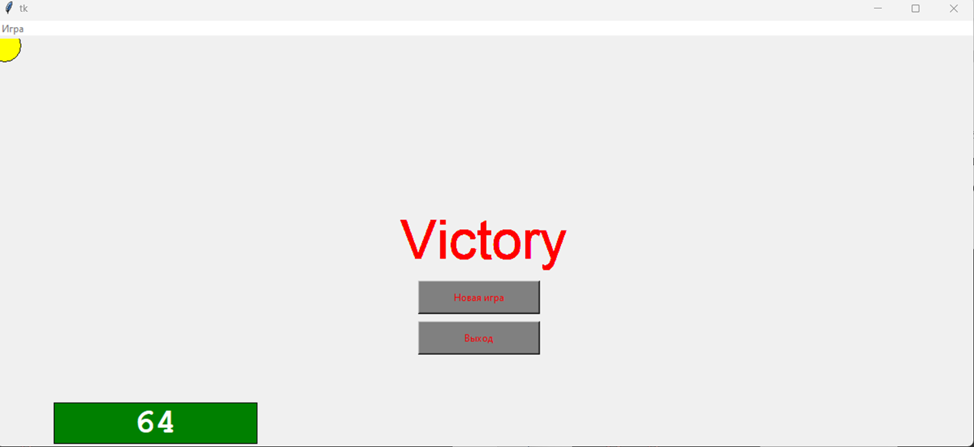


Рисунок 18. Окно, которое появляется при выигрыше

Вывод: все представленные алгоритмы были решены программным путём.

# **4.4 Тестирование программного обеспечения**

**Таблица 2 - Результаты испытаний**

| **№** | **Шаг испытаний (проверок)** | **№ пункта**  **Методики** | **Отметка о прохождении (да/нет)** | **Примечания** |
| --- | --- | --- | --- | --- |
| **1** | **Предоставление «Технического задания» отвечающего на все требования** | **1** | **Да** | **-** |
| **2** | **Предоставление «Руководства пользователя» отвечающего на все требования** | **1** | **Да** | **-** |
| **3** | **Предоставление настоящей программы и «Методики тестирования» отвечающего на все требования** | **1** | **Да** | **-** |
| **4** | **Создание нового окна после нажатия на кнопку «Открыть правила»** | **2** | **Да** | **-** |
| **5** | **Создание прописанных в коде форм (таких как текст, кнопка «Закрыть правила»)** | **2** | **Да** | **-** |
| **6** | **Вывод диалогового окна с вопросом о том, хотел бы игрок начать новую игру** | **4** | **Да** | **-** |
| **7** | **Проверка кнопок и запуск уровней сложности** | **4** | **Да** | **-** |
| **8** | **Проверка скорости шарика на каждом уровне сложности** | **4** | **Да** | **-** |
| **9** | **Проверка корректного отображения игрового поля** | **4** | **Да** | **-** |
| **10** | **Проверка корректного отображения генерации разноцветных кирпичей вверху игрового поля** | **5** | **Да** | **-** |
| **11** | **Проверка правильного движения шарика** | **5** | **Да** | **-** |
| **12** | **Проверка корректного уничтожения шариком цветных кирпичей** | **5** | **Да** | **-** |
| **13** | **При попадании шарика ниже платформы, происходит окончание игры** | **6** | **Да** | **-** |
| **14** | **При уничтожении все кирпичей игрок выигрывает** | **7** | **Да** | **-** |
| **15** | **Проверка корректного отображения счета, отображающегося внутри платформы** | **8** | **Да** | **-** |
| **16** | **Стабильная работа игры при попытках вызывать системной ошибки** | **9** | **Да** | **-** |
| **17** | **Проверка отображения игры при разрешении на полный экран** | **9** | **Да** | **-** |
| **18** | **Проверка отсутствия системных ошибок** | **9** | **Да** | **-** |

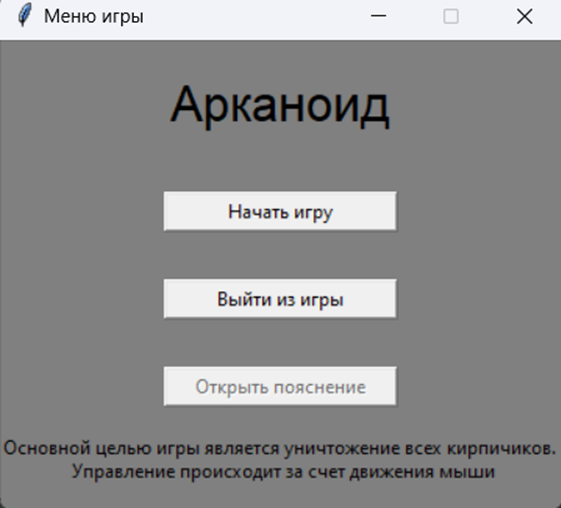
****

Рисунок 19. Результат выполнения 2 пункта испытаний

****

Рисунок 20. Результат выполнения 7-15 пунктов испытаний

# **ЗАКЛЮЧЕНИЕ**

Выполняя индивидуальное задание по учебной практике, были приобретены навыки:

* Написание технической документации;
* Проектирование и разработка кода приложения, на языке Python;
* Тестирование и отладка приложения;
* Работы с библиотекой Tkinter

Программное обеспеченье может быть использовано для игры и отдыха.

Таким образом, программа учебной практики ПМ.05 в период с 11.01.2021 по 16.06.2021 была выполнена мной в полном объеме.

При выполнении работ, предусмотренных заданием на практику, в организации ФГБОУ ВО «Вятский государственный университет» мной, Чувашевым Даниилом Алексеевичем, соблюдались правила внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологические правила и гигиенические нормативы. Пройден вводный инструктаж и инструктаж на рабочем месте. Замечаний не имел.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Python для начинающих [Электронный ресурс]. URL: [https://pythonworld.ru/](https://pythonworld.ru/%20)

2. Tkinter Tutorial [Электронный ресурс]. – URL: [https://www.pythontutorial.net/tkinter/](https://www.pythontutorial.net/tkinter/%20)

3. Tkinter – Python interface [Электронный ресурс]. – URL: [https://docs.python.org/3/library/tkinter.html#handy-reference](https://docs.python.org/3/library/tkinter.html%23handy-reference)

4. Python functions [Электронный ресурс]. – URL: [https://www.w3schools.com/python/python\_functions.asp](https://www.w3schools.com/python/python_functions.asp%20)

# **ПРИЛОЖЕНИЕ А**

**ИСХОДНЫЙ КОД**

import tkinter  
import random  
from tkinter import messagebox  
import sys  
import os  
from tkinter import \*  
  
window = tkinter.Tk()  
window.title("Меню игры")  
window.geometry("360x300")  
window.config(bg='gray')  
window.resizable(False, False)  
  
label = Label(window, text="Арканоид", font=("Helvetica", 24), bg="gray")  
label.pack(pady=20)  
  
new\_window = None  
  
def quit\_game():  
 window.destroy()  
  
def poyasnenie():  
 def close\_poyasnenye():  
 lable1.destroy()  
 close\_button.destroy()  
 poasnenue\_button.config(state='normal')  
  
 lable1 = tkinter.Label(text='Основной целью игры является уничтожение всех кирпичиков. \n Управление происходит за счет движения мыши', bg='gray')  
 lable1.pack()  
 poasnenue\_button.config(state='disabled')  
 close\_button = tkinter.Button(text='Закрыть пояснение', command=close\_poyasnenye, width=20)  
 close\_button.pack(pady=15)  
  
def createnewwindow():  
 global paused # Объявление глобальной переменной paused  
 paused = False # Инициализация переменной в начальное значение  
  
 def close\_window():  
 new\_window.destroy()  
  
  
 def lvl\_easy(): # Функция создания окна с правила игры  
 global x, x1, y, vx, vy, points, game\_mode, count  
 window.destroy()  
 vibor.destroy()  
  
 def New\_Game():  
 message2 = messagebox.askyesno('Новая игра', 'Вы действительно хотите начать новую игру?')  
 if message2:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
  
 def Vihod():  
 root.destroy()  
  
 def Vihod():  
 root.destroy()  
  
 def score\_up():  
 global points  
 points += 1  
 canvas.itemconfig(score, text=str(points))  
  
 def mouse\_move(event):  
 global x1  
 x1 = event.x  
 move\_platform\_and\_score()  
  
 def move\_platform\_and\_score():  
 if game\_mode and not paused:  
 canvas.coords(platform, x1 - PLATFORM\_W // 2, HEIGHT, x1 + PLATFORM\_W // 2, HEIGHT - PLATFORM\_H)  
 canvas.coords(score, x1, HEIGHT - PLATFORM\_H // 2)  
  
 def get\_brick():  
 for brick in bricks:  
 xb1, yb1, xb2, yb2 = canvas.coords(brick)  
 if xb1 < x < xb2 and yb1 < y < yb2:  
 return brick  
  
 WIDTH, HEIGHT = 1200, 500  
 x, y = WIDTH // 2, HEIGHT // 2  
 vx, vy = -10, -10  
 BALL\_RADIUS = 20  
 TIMEOUT = 40  
  
 x1 = WIDTH // 2  
 PLATFORM\_H = 50  
 PLATFORM\_W = 250  
  
 BRICKS\_PART = 0.3  
 ROWS, COLS = 4, 16  
  
 points = 0  
 game\_mode = True  
 paused\_label = None # Переменная для хранения объекта Label с надписью "Пауза"  
  
 root = tkinter.Tk()  
 root.resizable(False, False) # Disable window resizing  
  
 mainmenu = Menu(root)  
 root.config(menu=mainmenu)  
  
 canvas = tkinter.Canvas(root, width=WIDTH, height=HEIGHT)  
 canvas.pack()  
  
 ball = canvas.create\_oval(x - BALL\_RADIUS, y - BALL\_RADIUS, x + BALL\_RADIUS, y + BALL\_RADIUS, fill='yellow')  
  
 platform = canvas.create\_rectangle(x1 - PLATFORM\_W // 2, HEIGHT, x1 + PLATFORM\_W // 2, HEIGHT - PLATFORM\_H,  
 fill='green')  
  
 score = canvas.create\_text(x1, HEIGHT - PLATFORM\_H // 2, text='0', font=('Courier', 32, 'bold'), fill='white')  
  
 brick\_w, brick\_h = WIDTH / COLS, HEIGHT \* BRICKS\_PART / ROWS  
  
 bricks = []  
  
 for row in range(ROWS):  
 for col in range(COLS):  
 xb, yb = col \* brick\_w, row \* brick\_h  
 red, green, blue = (random.randint(0, 255) for \_ in range(3))  
 color = f'#{red:0>2x}{green:0>2x}{blue:0>2x}'  
 bricks.append(canvas.create\_rectangle(xb, yb, xb + brick\_w, yb + brick\_h, fill=color))  
  
 def game():  
 global x, y, vx, vy, points, game\_mode, count, paused, paused\_label  
 x, y = vx + x, vy + y  
 canvas.coords(ball, x - BALL\_RADIUS, y - BALL\_RADIUS, x + BALL\_RADIUS, y + BALL\_RADIUS)  
 if y <= BALL\_RADIUS:  
 vy = abs(vy)  
 if x <= BALL\_RADIUS or x >= WIDTH - BALL\_RADIUS:  
 vx = -vx  
  
 if x1 - PLATFORM\_W // 2 <= x <= x1 + PLATFORM\_W // 2 and \  
 y == HEIGHT - (BALL\_RADIUS + PLATFORM\_H):  
 vy = -vy  
  
 brick = get\_brick()  
  
 if brick:  
 vy = -vy  
 canvas.delete(brick)  
 bricks.pop(bricks.index(brick))  
 score\_up()  
  
 root.update()  
 root.title("Арканоид")  
  
  
 if points == 64:  
 game\_mode = False  
 canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='Victory', fill='red', font=(None, 50))  
 message = messagebox.askyesno(title='Победа', message='Хотите начать новую игру?')  
 if message:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
 else:  
 root.destroy()  
  
 if y < (HEIGHT - BALL\_RADIUS) and not paused:  
 root.after(TIMEOUT, game)  
 elif not paused:  
 game\_mode = False  
 canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='GAME OVER', fill='red', font=(None, 50))  
 message = messagebox.askyesno(title='Поражение', message='Хотите начать новую игру?')  
 if message:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
 else:  
 root.destroy()  
  
 elif paused:  
 if not paused\_label: # Если надписи "Пауза" нет, создаем ее  
 paused\_label = canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='Пауза', fill='red',  
 font=(None, 50))  
  
 def pause\_game():  
 global paused  
 paused = not paused  
 if not paused:  
 canvas.delete(paused\_label) # Удаляем надпись "Пауза"  
 game()  
  
 filemenu = Menu(mainmenu, tearoff=0)  
 filemenu.add\_command(label="Новая игра", command=New\_Game)  
  
 mainmenu.add\_cascade(label="Игра",  
 menu=filemenu)  
  
 pause\_button = tkinter.Button(root, text='Пауза', command=pause\_game)  
 pause\_button.pack()  
  
 game()  
 canvas.bind('<Motion>', mouse\_move)  
 root.mainloop()  
  
 def lvl\_medium (): # Функция создания окна с правила игры  
 global x, x1, y, vx, vy, points, game\_mode, count  
 window.destroy()  
 vibor.destroy()  
  
 def New\_Game():  
 message2 = messagebox.askyesno('Новая игра', 'Вы действительно хотите начать новую игру?')  
 if message2:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
  
 def Vihod():  
 root.destroy()  
  
 def score\_up():  
 global points  
 points += 1  
 canvas.itemconfig(score, text=str(points))  
  
 def mouse\_move(event):  
 global x1  
 x1 = event.x  
 move\_platform\_and\_score()  
  
 def move\_platform\_and\_score():  
 if game\_mode and not paused:  
 canvas.coords(platform, x1 - PLATFORM\_W // 2, HEIGHT, x1 + PLATFORM\_W // 2, HEIGHT - PLATFORM\_H)  
 canvas.coords(score, x1, HEIGHT - PLATFORM\_H // 2)  
  
 def get\_brick():  
 for brick in bricks:  
 xb1, yb1, xb2, yb2 = canvas.coords(brick)  
 if xb1 < x < xb2 and yb1 < y < yb2:  
 return brick  
  
 WIDTH, HEIGHT = 1200, 500  
 x, y = WIDTH // 2, HEIGHT // 2  
 vx, vy = -10, -10  
 BALL\_RADIUS = 20  
 TIMEOUT = 50  
  
 x1 = WIDTH // 2  
 PLATFORM\_H = 50  
 PLATFORM\_W = 250  
  
 BRICKS\_PART = 0.3  
 ROWS, COLS = 4, 16  
  
 points = 0  
 game\_mode = True  
 paused\_label = None # Переменная для хранения объекта Label с надписью "Пауза"  
  
 root = tkinter.Tk()  
 root.title("Арканоид")  
 root.resizable(False, False) # Disable window resizing  
  
  
  
 mainmenu = Menu(root)  
 root.config(menu=mainmenu)  
 root.resizable(False, False) # Disable window resizing  
  
  
  
  
 canvas = tkinter.Canvas(root, width=WIDTH, height=HEIGHT)  
 canvas.pack()  
  
 ball = canvas.create\_oval(x - BALL\_RADIUS, y - BALL\_RADIUS, x + BALL\_RADIUS, y + BALL\_RADIUS, fill='yellow')  
  
 platform = canvas.create\_rectangle(x1 - PLATFORM\_W // 2, HEIGHT, x1 + PLATFORM\_W // 2, HEIGHT - PLATFORM\_H,  
 fill='green')  
  
 score = canvas.create\_text(x1, HEIGHT - PLATFORM\_H // 2, text='0', font=('Courier', 32, 'bold'), fill='white')  
  
 brick\_w, brick\_h = WIDTH / COLS, HEIGHT \* BRICKS\_PART / ROWS  
  
 bricks = []  
  
 for row in range(ROWS):  
 for col in range(COLS):  
 xb, yb = col \* brick\_w, row \* brick\_h  
 red, green, blue = (random.randint(0, 255) for \_ in range(3))  
 color = f'#{red:0>2x}{green:0>2x}{blue:0>2x}'  
 bricks.append(canvas.create\_rectangle(xb, yb, xb + brick\_w, yb + brick\_h, fill=color))  
  
 def game():  
 global x, y, vx, vy, points, game\_mode, count, paused, paused\_label  
 x, y = vx + x, vy + y  
 canvas.coords(ball, x - BALL\_RADIUS, y - BALL\_RADIUS, x + BALL\_RADIUS, y + BALL\_RADIUS)  
 if y <= BALL\_RADIUS:  
 vy = abs(vy)  
 if x <= BALL\_RADIUS or x >= WIDTH - BALL\_RADIUS:  
 vx = -vx  
  
 if x1 - PLATFORM\_W // 2 <= x <= x1 + PLATFORM\_W // 2 and \  
 y == HEIGHT - (BALL\_RADIUS + PLATFORM\_H):  
 vy = -vy  
  
 brick = get\_brick()  
  
 if brick:  
 vy = -vy  
 canvas.delete(brick)  
 bricks.pop(bricks.index(brick))  
 score\_up()  
  
 root.update()  
  
 if points == 64:  
 game\_mode = False  
 canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='Victory', fill='red', font=(None, 50))  
 message = messagebox.askyesno(title='Победа', message='Хотите начать новую игру?')  
 if message:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
 else:  
 root.destroy()  
  
 if y < (HEIGHT - BALL\_RADIUS) and not paused:  
 root.after(TIMEOUT, game)  
 elif not paused:  
 game\_mode = False  
 canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='GAME OVER', fill='red', font=(None, 50))  
 message = messagebox.askyesno(title='Поражение', message='Хотите начать новую игру?')  
 if message:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
 else:  
 root.destroy()  
  
 elif paused:  
 if not paused\_label: # Если надписи "Пауза" нет, создаем ее  
 paused\_label = canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='Пауза', fill='red',  
 font=(None, 50))  
  
 def pause\_game():  
 global paused  
 paused = not paused  
 if not paused:  
 canvas.delete(paused\_label) # Удаляем надпись "Пауза"  
 game()  
  
 filemenu = Menu(mainmenu, tearoff=0)  
 filemenu.add\_command(label="Новая игра", command=New\_Game)  
  
 mainmenu.add\_cascade(label="Игра",  
 menu=filemenu)  
  
 pause\_button = tkinter.Button(root, text='Пауза', command=pause\_game)  
 pause\_button.pack()  
  
 game()  
 canvas.bind('<Motion>', mouse\_move)  
 root.mainloop()  
  
 def lvl\_hard(): # Функция создания окна с правила игры  
 global x, x1, y, vx, vy, points, game\_mode, count  
 window.destroy()  
 vibor.destroy()  
  
 def New\_Game():  
 message2 = messagebox.askyesno('Новая игра', 'Вы действительно хотите начать новую игру?')  
 if message2:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
  
 def Vihod():  
 root.destroy()  
  
 def score\_up():  
 global points  
 points += 1  
 canvas.itemconfig(score, text=str(points))  
  
 def mouse\_move(event):  
 global x1  
 x1 = event.x  
 move\_platform\_and\_score()  
  
 def move\_platform\_and\_score():  
 if game\_mode and not paused:  
 canvas.coords(platform, x1 - PLATFORM\_W // 2, HEIGHT, x1 + PLATFORM\_W // 2, HEIGHT - PLATFORM\_H)  
 canvas.coords(score, x1, HEIGHT - PLATFORM\_H // 2)  
  
 def get\_brick():  
 for brick in bricks:  
 xb1, yb1, xb2, yb2 = canvas.coords(brick)  
 if xb1 < x < xb2 and yb1 < y < yb2:  
 return brick  
  
 WIDTH, HEIGHT = 1200, 500  
 x, y = WIDTH // 2, HEIGHT // 2  
 vx, vy = -10, -10  
 BALL\_RADIUS = 20  
 TIMEOUT = 20  
  
 x1 = WIDTH // 2  
 PLATFORM\_H = 50  
 PLATFORM\_W = 250  
  
 BRICKS\_PART = 0.3  
 ROWS, COLS = 4, 16  
  
 points = 0  
 game\_mode = True  
 paused\_label = None # Переменная для хранения объекта Label с надписью "Пауза"  
  
 root = tkinter.Tk()  
 root.resizable(False, False) # Disable window resizing  
  
 mainmenu = Menu(root)  
 root.config(menu=mainmenu)  
  
 canvas = tkinter.Canvas(root, width=WIDTH, height=HEIGHT)  
 canvas.pack()  
  
 ball = canvas.create\_oval(x - BALL\_RADIUS, y - BALL\_RADIUS, x + BALL\_RADIUS, y + BALL\_RADIUS, fill='yellow')  
  
 platform = canvas.create\_rectangle(x1 - PLATFORM\_W // 2, HEIGHT, x1 + PLATFORM\_W // 2, HEIGHT - PLATFORM\_H,  
 fill='green')  
  
 score = canvas.create\_text(x1, HEIGHT - PLATFORM\_H // 2, text='0', font=('Courier', 32, 'bold'), fill='white')  
  
 brick\_w, brick\_h = WIDTH / COLS, HEIGHT \* BRICKS\_PART / ROWS  
  
 bricks = []  
  
 for row in range(ROWS):  
 for col in range(COLS):  
 xb, yb = col \* brick\_w, row \* brick\_h  
 red, green, blue = (random.randint(0, 255) for \_ in range(3))  
 color = f'#{red:0>2x}{green:0>2x}{blue:0>2x}'  
 bricks.append(canvas.create\_rectangle(xb, yb, xb + brick\_w, yb + brick\_h, fill=color))  
  
 def game():  
 global x, y, vx, vy, points, game\_mode, count, paused, paused\_label  
 x, y = vx + x, vy + y  
 canvas.coords(ball, x - BALL\_RADIUS, y - BALL\_RADIUS, x + BALL\_RADIUS, y + BALL\_RADIUS)  
 if y <= BALL\_RADIUS:  
 vy = abs(vy)  
 if x <= BALL\_RADIUS or x >= WIDTH - BALL\_RADIUS:  
 vx = -vx  
  
 if x1 - PLATFORM\_W // 2 <= x <= x1 + PLATFORM\_W // 2 and \  
 y == HEIGHT - (BALL\_RADIUS + PLATFORM\_H):  
 vy = -vy  
  
 brick = get\_brick()  
  
 if brick:  
 vy = -vy  
 canvas.delete(brick)  
 bricks.pop(bricks.index(brick))  
 score\_up()  
  
 root.update()  
 root.title("Арканоид")  
  
 if points == 64:  
 game\_mode = False  
 canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='Victory', fill='red', font=(None, 50))  
 message = messagebox.askyesno(title='Победа', message='Хотите начать новую игру?')  
 if message:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
 else:  
 root.destroy()  
  
 if y < (HEIGHT - BALL\_RADIUS) and not paused:  
 root.after(TIMEOUT, game)  
 elif not paused:  
 game\_mode = False  
 canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='GAME OVER', fill='red', font=(None, 50))  
 message = messagebox.askyesno(title='Поражение', message='Хотите начать новую игру?')  
 if message:  
 python = sys.executable  
 os.execl(python, python, \*sys.argv)  
 else:  
 root.destroy()  
  
 elif paused:  
 if not paused\_label: # Если надписи "Пауза" нет, создаем ее  
 paused\_label = canvas.create\_text(WIDTH // 2, HEIGHT // 2, text='Пауза', fill='red',  
 font=(None, 50))  
  
 def pause\_game():  
 global paused  
 paused = not paused  
 if not paused:  
 canvas.delete(paused\_label) # Удаляем надпись "Пауза"  
 game()  
  
 filemenu = Menu(mainmenu, tearoff=0)  
 filemenu.add\_command(label="Новая игра", command=New\_Game)  
  
 mainmenu.add\_cascade(label="Игра",  
 menu=filemenu)  
  
 pause\_button = tkinter.Button(root, text='Пауза', command=pause\_game)  
 pause\_button.pack()  
  
 game()  
 canvas.bind('<Motion>', mouse\_move)  
 root.mainloop()  
  
 def quit\_vibor():  
 vibor.destroy()  
  
 vibor = tkinter.Tk()  
 vibor.title("Выбор уровня")  
 vibor.geometry('300x250')  
 vibor.config(bg='gray')  
 easy\_button = tkinter.Button(vibor, text="Легкий уровень", command=lvl\_easy, width=20)  
 easy\_button.pack(pady=15)  
  
 medium\_button = tkinter.Button(vibor, text="Средний уровень", command=lvl\_medium, width=20)  
 medium\_button.pack(pady=15)  
  
 hard\_button = tkinter.Button(vibor, text="Сложный уровень", command=lvl\_hard, width=20)  
 hard\_button.pack(pady=15)  
  
 back\_button = tkinter.Button(vibor, text="Вернуться в главное меню", command=quit\_vibor, width=20)  
 back\_button.pack(pady=15)  
  
 vibor.mainloop()  
  
start\_button = tkinter.Button(text="Начать игру", command=createnewwindow, width=20)  
start\_button.pack(pady=15)  
  
quit\_button = tkinter.Button(text="Выйти из игры", command=quit\_game, width=20)  
quit\_button.pack(pady=15)  
  
poasnenue\_button = tkinter.Button(text="Открыть пояснение", command=poyasnenie, width=20)  
poasnenue\_button.pack(pady=15)  
  
  
  
window.mainloop()

# **ПРИЛОЖЕНИЕ Б**

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

# **ПРИЛОЖЕНИЕ В**

**ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ**

# **ПРИЛОЖЕНИЕ Г**

**РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ**