МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ

ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информатики и прикладной математики

Кафедра прикладной математики и экономико-математических методов

**КУРСОВАЯ РАБОТА**

по дисциплине:

**«Языки и методы программирования»**

Тема: «Игра «Жизнь»»

Направление 01.03.02 «Прикладная Математика и Информатика»

Направленность «Прикладная математика и информатика в экономике и управлении»

Обучающийся Терентьев Антон Евгеньевич

Группа ПМ-2201 Подпись\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Проверил Карахтанова Анна Борисовна

Должность к.т.н., доцент

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Санкт-Петербург

2023 г.

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc154010462)

[**1. ИГРА ЖИЗНЬ** 4](#_Toc154010463)

[**2. ПОЛЕ И ПОКОЛЕНИЯ** 5](#_Toc154010464)

[**2.1 Поле** 5](#_Toc154010465)

[**2.2 Поколения** 5](#_Toc154010466)

[**2.3 Следующее поколение** 5](#_Toc154010467)

[**3. РЕАЛИЗАЦИЯ** 6](#_Toc154010468)

[**3.1 Поколения** 6](#_Toc154010469)

[**3.1.1 Произвольное начало** 6](#_Toc154010470)

[**3.2 Элементы управления** 6](#_Toc154010471)

[**3.3 Редактирование** 6](#_Toc154010472)

[**3.4 Фигуры** 6](#_Toc154010473)

[**3.4.1 Космические корабли** 6](#_Toc154010474)

[**3.4.2 Осцилляторы** 7](#_Toc154010475)

[**3.4.3 Ружье** 7](#_Toc154010476)

[**4. РЕЗУЛЬТАТЫ** 8](#_Toc154010477)

[**4.1 Интерфейс** 8](#_Toc154010478)

[**4.2 Жизнь** 8](#_Toc154010479)

[**ЗАКЛЮЧЕНИЕ** 10](#_Toc154010480)

[**БИБЛИОГРАФИЧЕСКИЙ СПИСОК** 11](#_Toc154010481)

# **ВВЕДЕНИЕ**

# 

Целью курсовой работы является визуальная реализация клеточного автомата известного под названием Игра «Жизнь», в среде разработки Python.

Задачи курсовой работы:

1. разработка системы хранения информации о поколениях;
2. разработка визуализации каждого поколения;
3. разработка интерфейса взаимодействия пользователя с автоматом;
4. добавление поколений, для изучения пользователем.

**1. ИГРА ЖИЗНЬ**

Игра «Жизнь» представляет из себя клеточное поле неопределенного размера, каждая клетка которого находится в одном из двух состояний – мертвая или живая. В каждом следующем поколении живость клетки определяется из информации о живости ее в предыдущем поколении и количестве живых клеток-соседей. Клетка остается живой если у нее было два или три живых соседа, иначе умирает. Мертвая клетка оживает, если у нее было ровно три живых соседа, иначе остается мертвой.

# **2. ПОЛЕ И ПОКОЛЕНИЯ**

## **2.1 Поле**

В оригинальной постановке задачи Жизнь происходит на бесконечном клеточном поле. Поле, представленное в реализации, имеет изначальные параметры 36 x 64 клетки и может быть проинтерпретировано как изображение поверхности тора. Это происходит по причине, что в написанных алгоритмах клетки на горизонтальных краях считаются соседними друг для друга. Аналогично вертикальные края тоже являются соседями.

В результате происходящая Жизнь, перетекает через края не в бесконечность, а на другую часть экрана.

**2.2 Поколения**

В ядре игры «Жизнь» лежат поколения клеток. Каждое следующее поколение зависит только от предыдущего. По этой причине, для визуализации существует необходимость только в сохранении двух поколений: нынешнего и предыдущего.

Каждое поколение это набор данных о состоянии каждой клетки поля «живая» или «мертвая».

## **2.3 Следующее поколение**

Следующее поколение вычисляется на основе данных о состоянии клетки и ее соседей.

# **3. РЕАЛИЗАЦИЯ**

## **3.1 Поколения**

Поколения в реализации хранятся в виде двух двумерных массивов, состоящих из булевых переменных. Все клетки находятся в мертвом состоянии при начальном запуске программы.

**3.1.1 Произвольное начало**

В любой момент можно запустить Жизнь из произвольно сгенерированного поля и наблюдать за изменениями от поколения к поколению. Данное поколение генерируется как двумерный массив, заполненный произвольными значениями: истина, ложь.

**3.2 Элементы управления**

Для удобства наблюдения за развитием поколений были добавлены такие элементы управления как кнопка «Пауза / Запуск», кнопка «Ускорение» и кнопка «Замедление».

**3.3 Редактирование**

В любой момент наблюдения за поколениями есть опция поменять состояние любой из клеток. Данная возможность добавлена для взаимодействия пользователя и программы и для создания собственных поколений клеток.

**3.4 Фигуры**

Воспользовавшись справочными материалами, были добавлены различные фигуры.

**3.4.1 Космические корабли**

Это такие фигуры, которые при запуске по бесконечному пустому полю будут двигаться, циклически повторяя формы, при этом передвигаясь в пространстве.

В реализации добавлены такие космические корабли как «Глайдер», «Легкий космический корабль» и другие.

**3.4.2 Осцилляторы**

Такие фигуры, которые повторяют по некоторому порядку состояния, сохраняя положение от поколения к поколению.

В реализации данный класс фигур представлен представителями «Восьмерка», «Юникс», «Пульсар» и другие.

**3.4.3 Ружье**

Последняя фигура, которая не относится ни к одном из ранее названных классов это «Планерное ружье Госпера». Это фигура, которая непрерывно создает «Глайдеры» уходящие бесконечным потоком от самого «Ружья».

**4. РЕЗУЛЬТАТЫ**

**4.1 Интерфейс**

Для удобства и простоты реализации интерфейса была использована библиотека tkinter.

Лаконичность результата позволяет наблюдать за поколениями, не отвлекаясь на громоздкие конструкции интерфейса.

Изображение выглядит как текст, снимок экрана, График, линия

Автоматически созданное описание

Рисунок 1 – Интерфейс

**4.2 Жизнь**

Рассмотрим на рисунках 2 и 3 поля, заполненные произвольным образом и заполненные фигурой «Дротик».

Произвольное поле интересно наблюдать, после большого числа поколений, когда живые клетки образуют колонии, периодически взаимодействующие друг с другом. Для этого имеет смысл ускорять процесс и наблюдать его с большей скоростью.

Изображение выглядит как кроссворд, текст, шов

Автоматически созданное описание

Рисунок 2 – Произвольное поле, после нескольких поколений перерождения

Изображение выглядит как диаграмма, График, линия, текст

Автоматически созданное описание

Рисунок 3 – Космический корабль «Дротик».

# **ЗАКЛЮЧЕНИЕ**

Реализация «Жизни» произведена успешно. Построена система отображения и изменения поколений. Имеется гибкая возможность изменения поля.

Путей для развития данного проекта видится несколько. Добавление различных вариаций поля, возможность загружать начальное положение в стандартизированном виде и многое другое.

# **БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

1. N. Johnston Conway’s Game of Life. Mathematics and Construction. / N. Johnston and D. Green, 2022.
2. Conway Life [Электронный ресурс]: URL: https://conwaylife.com/
3. Python [Электронный ресурс]:URL: https://docs.python.org/3/