**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ ГОРОДА МОСКВЫ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ**

Государственное бюджетное образовательное учреждение города Москвы   
«Школа №185 имени Героя Советского Союза, Героя Социалистического Труда   
В.С. Гризодубовой»

**Проектная работа**

**«Разработка стратегической игры „Age Of Pygame“ на модуле PyGame языка программирования Python»**

**Работу выполнили:** Синицын Виталий, Андрианов Глеб, Бондарев Павел

Ученики 10 класса ГБОУ Школа №185 г. Москвы

**Руководитель проекта:** учитель информатики

Галганова Дарья Анатольевна

Москва 2025

**Оглавление**

[Введение 3](#_Toc158198788)

[1. Теоретическая часть 4](#_Toc158198789)

[1.1. Требования к безопасному паролю 4](#_Toc158198790)

[1.2. Обзор существующих решений 4](#_Toc158198791)

[1.3. Техническое описание предполагаемого продукта 5](#_Toc158198792)

[2. Практическая часть 7](#_Toc158198793)

[2.1. Разработка интерфейса 7](#_Toc158198794)

[2.2. Написания программы 7](#_Toc158198795)

[2.3. Тестирование программы 7](#_Toc158198796)

[2.4. Оценка применимости 8](#_Toc158198797)

[Заключение 9](#_Toc158198798)

[Список ресурсов 10](#_Toc158198799)

# **Введение**

**Актуальность**: Современные стратегические игры зачастую обладают высокой сложностью, что делает их менее доступными для широкого круга пользователей. Кроме того, многие из них не обеспечивают достаточного уровня случайности в игровом процессе, что снижает интерес при повторном прохождении. Проект «Age of Pygame» предлагает решение: создание игры с интуитивно понятным управлением, динамической генерацией карты и возможностью взаимодействия с искусственным интеллектом. Это позволяет обеспечить реиграбельность, увлекательный игровой процесс и интерес к изучению программирования игр на Python.

**Цель**: Разработка простой и увлекательной стратегии с использованием Python и библиотеки Pygame, которая включает случайную генерацию карты, управление юнитами, пошаговые действия игрока и искусственного интеллекта, а также реализацию боевой системы.

**Задачи**:

* Создать карту, разбитую на регионы, с динамически генерируемыми границами.
* Реализовать систему пошаговых ходов для игрока и искусственного интеллекта.
* Разработать алгоритмы, позволяющие ИИ принимать решения о захвате регионов и управлении юнитами.
* Внедрить механику боевой системы для взаимодействия юнитов.
* Обеспечить удобное управление и отображение состояния карты, юнитов и текущего хода.
* Провести тестирование проекта на различных сценариях игрового процесса.

**Этапы реализации проекта:**

* + 1. Постановка целей и задач.
    2. Проектирование структуры.
    3. Разработка системы генерации карты.
    4. Реализация механики подконтрольных юнитов.
    5. Внедрение системы победы над другими игроками.
    6. Создание алгоритмов искусственного интеллекта.
    7. Разработка системы управления игроком.
    8. Тестирование игрового процесса.
    9. Оптимизация и доработка.
    10. Презентация и публикация.

### **1. Теоретическая часть**

#### **1.1. Требования к стратегической игре**

Стратегическая игра должна быть доступной для широкого круга пользователей, обеспечивать увлекательный и динамичный игровой процесс, а также обладать высокой реиграбельностью за счёт случайных элементов. Основные требования к стратегической игре:

* **Увлекательность:** игра должна удерживать внимание пользователя благодаря разнообразным механикам, таким как пошаговые ходы, боевая система и генерация уникальных карт.
* **Простота управления:** управление должно быть интуитивно понятным и не вызывать затруднений даже у новичков.
* **Реиграбельность:** каждая игровая сессия должна предлагать новый опыт за счёт случайной генерации карты и работы искусственного интеллекта.
* **Оптимальная сложность:** игра должна быть достаточно сложной, чтобы мотивировать игроков разрабатывать стратегию, но не отпугивать чрезмерной сложностью.

#### **1.2. Обзор существующих решений**

На рынке игр существует множество стратегий, каждая из которых обладает своими особенностями. Однако большинство из них либо слишком сложны для освоения, либо быстро теряют интерес из-за отсутствия уникальных элементов, таких как случайная генерация карты.

Ключевые недостатки существующих решений:

* Сложные интерфейсы, не адаптированные для начинающих игроков.
* Отсутствие достаточного уровня случайности в игровом процессе, что снижает интерес при повторном прохождении.
* Отсутствие простых и доступных обучающих стратегий для начинающих программистов.

Проект «Age of Pygame» предлагает устранить эти недостатки за счёт простого интерфейса, уникальной генерации карт и пошагового игрового процесса, который легко осваивается.

#### **1.3. Техническое описание предполагаемого продукта**

Проект представляет собой пошаговую стратегию, выполненную на языке Python с использованием библиотеки Pygame. Основные элементы игры:

1. **Генерация карты:** карта состоит из сетки прямоугольных регионов, распределённых между странами с помощью случайного алгоритма.
2. **Юниты:** каждое государство управляет юнитами, которые могут перемещаться между регионами, захватывать территории и сражаться с противниками.
3. **Ходы:** игрок и искусственный интеллект совершают действия поочерёдно.
4. **Боевая система:** реализован механизм боёв между юнитами разных стран, при котором определяется победитель.
5. **Визуализация:** карта, юниты и действия отображаются в 2D-графике, что упрощает восприятие игрового процесса.

Достоинства продукта:

* Простота освоения и управления.
* Высокая реиграбельность за счёт случайной генерации карт.
* Возможность тестирования базовых алгоритмов ИИ в игровой среде.
* Подходит для обучения программированию игр.

### **2. Практическая часть**

#### **2.1. Генерация карты**

Для генерации карты используется сетка фиксированного размера, разделённая на регионы. Алгоритм случайного блуждания определяет распределение регионов между странами, что делает каждую игровую сессию уникальной.

#### **2.2. Реализация механик**

Игра написана на языке Python с использованием библиотеки Pygame. Реализованы следующие механики:

* Пошаговое управление юнитами игрока.
* Автоматические действия искусственного интеллекта.
* Боевая система, определяющая исход сражений между юнитами.
* Захват территорий вражеских государств.

#### **2.3. Тестирование**

Игровой процесс был протестирован на корректность работы ключевых функций:

1. **Генерация карты:** проверена уникальность и корректность распределения регионов.
2. **Искусственный интеллект:** протестированы алгоритмы принятия решений ИИ.
3. **Боевая система:** проверена правильность расчёта исходов боёв.
4. **Управление:** протестирована интуитивность взаимодействия игрока с игрой.

#### **2.4. Оценка применимости**

Проект может быть использован:

* Для обучения программированию игр на Python.
* В качестве базы для разработки более сложных стратегических игр.
* Как обучающий материал по основам работы с искусственным интеллектом и случайными алгоритмами.

Игра не требует дополнительных установок или затрат и совместима с любой системой, поддерживающей Python и Pygame.

**Заключение:** проект по разработке навыка для улучшения аудирования на английском языке через Яндекс Алису отвечает актуальной потребности в доступных и эффективных методах изучения иностранного языка. Благодаря интерактивным упражнениям и адаптивному подходу пользователи смогут тренировать восприятие английской речи на слух в удобное для них время и в комфортной форме. Навык, разработанный для Яндекс Алисы, не только повысит мотивацию к самостоятельному обучению, но и сделает его доступным для широкой аудитории. Реализация данного проекта будет способствовать развитию языковых компетенций, необходимых для успешного общения и профессионального роста в современном глобализированном мире.

# **Список ресурсов**

1. Как составить надёжный пароль: [Электронный ресурс] // Лаборатория Касперского. URL: <https://support.kaspersky.ru/common/windows/3730>. (Дата обращения: 01.02.24).
2. Генератор паролей: [Электронный ресурс] // Генератор паролей онлайн «azpassword». URL: <https://www.azpassword.ru/?useSymbols=true>. (Дата обращения: 01.02.24).
3. Генератор паролей: [Электронный ресурс] // Генератор случайных комбинаций «Рандомус». URL: <https://randomus.ru/password>. (Дата обращения: 01.02.24).
4. Генератор паролей: [Электронный ресурс] // Генератор паролей онлайн «PASSWORD CRAFT». URL: <https://passwordcraft.ru/>. (Дата обращения: 01.02.24).
5. Генератор паролей: [Электронный ресурс] // Генератор случайных событий и комбинаций «randstuff». URL: <https://randstuff.ru/password/>. (Дата обращения: 01.02.24)