

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Хоружий, Ирина Ивановна

Должность: Директор института экономики и управления АПК

Дата подписания: 2025.08.28 11:21:14

Уникальный электронный ключ:

1e90b132d9b04d6c67585160b015dddf2cb1e6a9



**МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА
РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ –

МСХА имени К.А. ТИМИРЯЗЕВА»

(ФГБОУ ВО РГАУ - МСХА имени К.А. Тимирязева)

**Институт экономики и управления АПК
Кафедра статистики и кибернетики**

УТВЕРЖДАЮ:

Директор института

экономики и управления АПК

И.И. Хоружий



«28» августа 2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ФТД.02 Разработка игр на Python

для подготовки бакалавров

ФГОС ВО

Направление: 09.03.02 «Информационные системы и технологии»

Направленность: Фуллстек разработка, Системная аналитика и разработка программного обеспечения

Курс 3, 4

Семестр 6, 7

Форма обучения: очная

Год начала подготовки: 2025

Москва, 2025

Разработчики: Демичев В.В., канд. экон. наук, доцент,
Невзоров А.С., старший преподаватель
(ФИО, ученая степень, ученое звание)



(подпись)

«26» августа 2025 г.

Рецензент: Худякова Е.В., д-р.экон.наук, профессор
(ФИО, ученая степень, ученое звание)



(подпись)

«26» августа 2025 г.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.02 Информационные системы и технологии, профессионального стандарта и учебного плана 2025 года начала подготовки

Программа обсуждена на заседании кафедры статистики и кибернетики
протокол № 11 от «26» августа 2025 г.

И.о. зав. кафедрой Уколова А.В., канд. экон. наук, доцент
(ФИО, ученая степень, ученое звание)



(подпись)

«26» августа 2025 г.

Согласовано:

Председатель учебно-методической
комиссии института экономики и управления АПК
Гупалова Т.Н., канд. экон. наук, доцент
(ФИО, ученая степень, ученое звание)



(подпись)

«28» августа 2025 г.

И.о. зав. выпускающей кафедрой статистики и кибернетики
Уколова А.В., канд. экон. наук, доцент
(ФИО, ученая степень, ученое звание)



(подпись)

«26» августа 2025 г.

Заведующий отделом комплектования ЦНБ



(подпись)

СОДЕРЖАНИЕ

АННОТАЦИЯ.....	4
1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....	5
2. МЕСТО ДИСЦИПЛИНЫ В УЧЕБНОМ ПРОЦЕССЕ	5
3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	5
4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ	1
4.1 РАСПРЕДЕЛЕНИЕ ТРУДОЁМКОСТИ ДИСЦИПЛИНЫ ПО ВИДАМ РАБОТ ПО СЕМЕСТРАМ	1
4.2 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ.....	1
4.3 ЛЕКЦИИ/ПРАКТИЧЕСКИЕ ЗАНЯТИЯ.....	2
5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ	4
6. ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ	4
6.1. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ, НЕОБХОДИМЫЕ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ	4
6.2. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ КОНТРОЛЯ УСПЕВАЕМОСТИ, ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ	5
7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ.....	8
7.1 ОСНОВНАЯ ЛИТЕРАТУРА	8
7.2 ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА.....	8
8. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....	9
9. ПЕРЕЧЕНЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ.....	9
10. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ.....	9
11. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ОБУЧАЮЩИМСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ. 10	
Виды и формы отработки пропущенных занятий	11
12. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПРЕПОДАВАТЕЛЯМ ПО ОРГАНИЗАЦИИ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ.....	11

АННОТАЦИЯ

рабочей программы учебной дисциплины ФТД.02 «Разработка игр на Python» для подготовки бакалавров по направлению 09.03.02 «Информационные системы и технологии» направленности «Фуллстек разработка», «Системная аналитика и разработка программного обеспечения»

Цель освоения дисциплины. Основная цель дисциплины «Разработка игр на Python» – формирование у обучающихся знаний, умений и навыков, необходимых для создания компьютерных игр с использованием языка программирования Python, включая освоение основных концепций игрового дизайна, работы с графикой, анимацией, звуком и взаимодействием с пользователем, а также развитие способности применять полученные знания для реализации собственных игровых проектов.

Место дисциплины в учебном плане: дисциплина включена в факультативные дисциплины учебного плана по направлению подготовки 09.03.02 «Информационные системы и технологии».

Требования к результатам освоения дисциплины: в результате освоения дисциплины формируются следующие компетенции (индикаторы): Пкос-4 (Пкос-4.1; Пкос-4.2; Пкос-4.3).

Краткое содержание дисциплины:

Первый раздел знакомит студентов с базовыми концепциями создания игр на Python. Включает изучение синтаксиса Python, работу с библиотеками для разработки игр (например, Pygame), а также создание простых 2D-игр. Студенты научатся работать с графикой, анимацией, звуком и управлением в играх, создавая свои первые проекты, такие как платформеры или аркады.

Во втором разделе акцент делается на проектировании игровых механик и архитектуры. Студенты изучат принципы объектно-ориентированного программирования (ООП) и их применение в разработке игр. Они научатся создавать игровые циклы, обрабатывать события, управлять состоянием игры и разрабатывать пользовательские интерфейсы (UI). Раздел также включает создание более сложных игр с четкой структурой и логикой.

Третий раздел посвящен углубленным темам, таким как работа с физикой, коллизиями, оптимизацией производительности и сетевыми функциями. Студенты изучат, как создавать реалистичные взаимодействия объектов, улучшать производительность игр и добавлять мультиплеерные возможности. Также рассматриваются основы 3D-игр с использованием специализированных библиотек, таких как Ursina или Panda3D, и завершающий этап – публикация игр.

Общая трудоемкость дисциплины составляет 4 зачетные единицы (144 часов, в т. ч. 8 часов практической подготовки).

Промежуточный контроль: зачет.

1. Цель освоения дисциплины

Целью освоения дисциплины «Разработка игр на Python» является формирование у обучающихся знаний, умений и навыков, необходимых для создания компьютерных игр с использованием языка программирования Python, включая освоение основных концепций игрового дизайна, работы с графикой, анимацией, звуком и взаимодействием с пользователем, а также развитие способности применять полученные знания для реализации собственных игровых проектов.

2. Место дисциплины в учебном процессе

Дисциплина «Разработка игр на Python» включена в факультативные дисциплины учебного плана по направлению подготовки 09.03.02 «Информационные системы и технологии». Дисциплина «Разработка игр на Python» реализуется в соответствии с требованиями ФГОС ВО, ОПОП ВО и Учебного плана по направлению 09.03.02 «Информационные системы и технологии».

Предшествующими курсами, на которых непосредственно базируется дисциплина «Разработка игр на Python» являются «Введение в компьютерные науки на иностранном языке», «Моделирование информационных систем», «Разработка профессиональных приложений», «Алгоритмизация и программирование», «Программирование на языке Python» и др.

Рабочая программа дисциплины «Разработка игр на Python» для инвалидов и лиц с ограниченными возможностями здоровья разрабатывается индивидуально с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Образовательные результаты освоения дисциплины обучающимся, представлены в таблице 1.

Таблица 1

Требования к результатам освоения учебной дисциплины

№ п/п	Код компете нции	Содержание компетенции (или её части)	Индикат оры мпетенций	В результате изучения учебной дисциплины обучающиеся должны:		
				знать	уметь	владеть
1.	Пкос-4	Способен осуществлять разработку, отладку и рефакторинг кода программного обеспечения, интеграцию программных модулей и компонент, в том числе взаимодействующих с внешней средой, средствами выбранных языков программирования	Пкос-4.1	методы и приемы формализации и алгоритмизации поставленных задач; нотации и программные продукты для графического отображения алгоритмов; алгоритмы решения типовых задач, области и способы их применения; методологии разработки программного обеспечения; синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования; особенности выбранной среды программирования; методы и приемы отладки программного кода, повышения читаемости программного кода; типы и форматы сообщений об ошибках, предупреждений	-	-
			Пкос-4.2	-	использовать методы и приемы формализации и алгоритмизации поставленных задач; использовать программные продукты для графического отображения алгоритмов;	-

					применять стандартные алгоритмы в соответствующих областях; применять выбранные языки программирования для написания программного кода; использовать выбранную среду программирования; применять инструментарий для создания и актуализации исходных текстов программ; выявлять ошибки в программном коде, интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов; применять методы и приемы отладки программного кода	
			Пкос-4.3	-	-	навыками; составления формализованных описаний решений

						поставленных задач в соответствии с требованиями технического задания; разработки алгоритмов решения поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов; отладки программного кода на уровне программных модулей и межмодульных взаимодействий и взаимодействий с окружением
--	--	--	--	--	--	---

4. Структура и содержание дисциплины

4.1 Распределение трудоёмкости дисциплины по видам работ по семестрам

Общая трудоёмкость дисциплины составляет 4 зачетные единицы (144/8 часов), их распределение по видам работ семестрам представлено в таблице 2.

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 2

Распределение трудоёмкости дисциплины по видам работ по семестрам

Вид учебной работы	Трудоёмкость		
	Итого за 2 семестра	6 семестр	7 семестр
Общая трудоёмкость дисциплины по учебному плану	144/8	72/4	72/4
1. Контактная работа	32,5/4	16,25/4	16,25/4
Аудиторная работа	32,5/4	16,25/4	16,25/4
<i>практические занятия (ПЗ)</i>	32/4	16/4	16/4
<i>контактная работа на промежуточном контроле (КРА)</i>	0,5	0,25	0,25
2. Самостоятельная работа (СРС)	111,5	55,75	55,75
<i>самостоятельное изучение разделов, самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий, подготовка к практическим занятиям)</i>	111,5	55,75	55,75
Вид промежуточного контроля:		Зачет	Зачет

4.2 Содержание дисциплины

Таблица 3

Тематический план учебной дисциплины

Наименование разделов и тем дисциплин (укрупнённо)	Всего	Аудиторная работа			Внеаудиторная работа СР
		Л	ПЗ всего/*	ПКР	
Раздел 1 Основы разработки игр на Python	35,5	-	8/2	-	27,5
Раздел 2 Проектирование и архитектура игр	36,25	-	8/2	-	28,25
контактная работа на промежуточном контроле (КРА)	0,25	-	-	0,25	-
Всего за 5 семестр	72/4	-	16/4	0,25	55,75
Раздел 3 Продвинутое техники и оптимизация	71,75	-	16/4	-	55,75
контактная работа на промежуточном контроле (КРА)	0,25	-	-	0,25	-
Всего за 6 семестр	72/4	-	16/4	0,25	55,75
Итого по дисциплине	144/8	-	32/8	0,5	111,5

Раздел 1 Основы разработки игр на Python

Этот раздел знакомит студентов с базовыми концепциями создания игр на Python. Включает изучение синтаксиса Python, работу с библиотеками для разработки игр (например, Pygame), а также создание простых 2D-игр. Студенты научатся работать с графикой, анимацией, звуком и управлением в играх, создавая свои первые проекты, такие как платформеры или аркады.

Раздел 2 Проектирование и архитектура игр

В этом разделе акцент делается на проектировании игровых механик и архитектуры. Студенты изучат принципы объектно-ориентированного программирования (ООП) и их применение в разработке игр. Они научатся создавать игровые циклы, обрабатывать события, управлять состоянием игры и разрабатывать пользовательские интерфейсы (UI). Раздел также включает создание более сложных игр с четкой структурой и логикой.

Раздел 3 Продвинутые техники и оптимизация

Этот раздел посвящен углубленным темам, таким как работа с физикой, коллизиями, оптимизацией производительности и сетевыми функциями. Студенты изучат, как создавать реалистичные взаимодействия объектов, улучшать производительность игр и добавлять мультиплеерные возможности. Также рассматриваются основы 3D-игр с использованием специализированных библиотек, таких как Ursina или Panda3D, и завершающий этап – публикация игр.

4.3 Лекции/практические занятия

Таблица 4

Содержание лекций/практических занятий и контрольные мероприятия

№ п/п	Название раздела, темы	№ и название лекций/ лабораторных/ практических/ семинарских занятий	Формируемые компетенции	Вид контрольного мероприятия	Кол-во часов/ из них практическая подготовка
1.	Раздел 1 Основы разработки игр на Python		Пкос-4.1; Пкос-4.2; Пкос-4.3		8/2
	Тема 1 Основы разработки игр на Python	Практическая работа 1. Создание простого 2D-платформера	Пкос-4.1; Пкос-4.2; Пкос-4.3	защита практической работы	4
		Практическая работа 2. Игра "Змейка" с использованием Pygame	Пкос-4.1; Пкос-4.2; Пкос-4.3	защита практической работы	4/2
2.	Раздел 2 Проектирование и архитектура игр		Пкос-4.1; Пкос-4.2; Пкос-4.3		8/2
	Тема 1 Проектирование и	Практическая работа 3. Разработка игры "Крестики-нолики"	Пкос-4.1; Пкос-4.2; Пкос-4.3	защита практической работы	4/2

№ п/п	Название раздела, темы	№ и название лекций/ лабораторных/ практических/ семинарских занятий	Формируемые компетенции	Вид контрольного мероприятия	Кол-во часов/ из них практическая подготовка
	архитектура игр	нолики" с использованием ООП			
		Практическая работа 4. Создание интерактивного меню для игры	Пкос-4.1; Пкос-4.2; Пкос-4.3	защита практической работы	4
3.	Раздел 3 Продвинутое техники и оптимизация		Пкос-4 Пкос-5.2		16/4
	Тема 3 Продвинутое техники и оптимизация	Практическая работа 5. Реализация физики падающих объектов	Пкос-4.1; Пкос-4.2; Пкос-4.3	защита практической работы	4
		Практическая работа 6. Оптимизация производительности игры	Пкос-4.1; Пкос-4.2; Пкос-4.3	защита практической работы	4
		Практическая работа 7. Создание простого мультиплеера для игры	Пкос-4.1; Пкос-4.2; Пкос-4.3	защита практической работы	4/2
		Практическая работа 8. Разработка 3D-игры с использованием Ursina	Пкос-4.1; Пкос-4.2; Пкос-4.3	защита практической работы	4/2

Таблица 5

Перечень вопросов для самостоятельного изучения дисциплины

№ п/п	Название раздела, темы	Перечень рассматриваемых вопросов для самостоятельного изучения
Раздел 1 Основы разработки игр на Python		
1.	Тема 1 Основы разработки игр на Python	Изучение библиотеки Pygame: возможности и примеры использования. Изучение форматов изображений, создание спрайтов и анимаций. Работа с библиотеками для воспроизведения звука и их интеграция в проекты.
Раздел 2 Проектирование и архитектура игр		
2.	Тема 1 Проектирование и архитектура игр	Изучение ООП на примере создания классов для игровых объектов. Создание системы управления экранами (меню, пауза, игра). Изучение библиотек для создания UI и их применение в проектах. Работа с файлами для хранения данных уровней и их динамической загрузки.
Раздел 3 Продвинутое техники и оптимизация		
3.	Тема 1 Продвинутое техники и оптимизация	Изучение алгоритмов обнаружения столкновений и их применение. Исследование методов оптимизации кода и ресурсов (например, использование спрайт-листов). Изучение основ сетевого программирования и создание простого мультиплеера. Изучение платформ для публикации игр и подготовка проекта к релизу.

5. Образовательные технологии

Таблица 6

Применение активных и интерактивных образовательных технологий

№ п/п	Тема и форма занятия	Наименование используемых активных и интерактивных образовательных технологий (форм обучения)
1.	Практическая работа 7. Создание простого мультимедиа для игры	ПЗ Компьютерная симуляция
2.	Практическая работа 5. Реализация физики падающих объектов	ПЗ Компьютерная симуляция

6. Текущий контроль успеваемости и промежуточная аттестация по итогам освоения дисциплины

6.1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности

Перечень вопросов, выносимых на промежуточную аттестацию (зачет)

1. Какие библиотеки чаще всего используются для разработки игр на Python?
2. Как установить и настроить библиотеку Pygame?
3. Какие основные компоненты необходимы для создания игрового окна в Pygame?
4. Как обрабатывать события клавиатуры и мыши в Pygame?
5. Как создать и отобразить спрайт в игре?
6. Как добавить анимацию к игровому объекту?
7. Какие форматы изображений поддерживаются в Pygame?
8. Как добавить звуковые эффекты и фоновую музыку в игру?
9. Как реализовать движение игрового персонажа?
10. Как создать простую игру, например, "Змейку" или "Платформер"?
11. Какие принципы объектно-ориентированного программирования (ООП) применяются в разработке игр?
12. Как создать класс для игрового объекта?
13. Что такое игровой цикл и как он работает?
14. Как управлять состоянием игры (меню, пауза, игра)?
15. Как создать систему уровней в игре?
16. Как загружать уровни из внешних файлов?
17. Как разработать пользовательский интерфейс (UI) для игры?
18. Как реализовать систему подсчета очков и отображения счета?
19. Как создать систему сохранения и загрузки игрового прогресса?
20. Как организовать код игры для удобства поддержки и расширения?
21. Как реализовать физику падающих объектов в игре?
22. Как обнаруживать коллизии между объектами?
23. Какие алгоритмы используются для оптимизации обнаружения коллизий?
24. Как улучшить производительность игры?

25. Что такое спрайт-листы и как их использовать?
26. Как добавить в игру искусственный интеллект (ИИ) для врагов?
27. Как реализовать мультиплеер в игре?
28. Какие библиотеки используются для создания сетевых игр?
29. Как создать простую 3D-игру на Python?
30. Какие библиотеки подходят для разработки 3D-игр?
31. Как работать с частицами и эффектами в играх?
32. Как создать procedurally generated уровни?
33. Как добавить в игру систему достижений?
34. Как интегрировать в игру сторонние API (например, для авторизации или статистики)?
35. Как подготовить игру к публикации?
36. Какие платформы подходят для распространения игр на Python?
37. Как создать инсталлятор для игры?
38. Какие инструменты используются для тестирования игр?
39. Как работать с версионным контролем (Git) при разработке игр?
40. Какие современные тенденции в разработке игр стоит учитывать?

Пример работ

Практическая работа 1. Создание простого 2D-платформера

Цель работы: Освоить базовые навыки разработки 2D-игр на Python с использованием библиотеки Pygame. Научиться создавать игровое окно, управлять персонажем, обрабатывать коллизии и разрабатывать простые игровые механики.

Исходные данные: Установленная библиотека Pygame. Набор спрайтов для персонажа, платформ и фона (можно использовать бесплатные ассеты или создать самостоятельно). Звуковые эффекты для прыжков и фоновой музыки (опционально).

Требуется:

1. Создать игровое окно:

- Инициализировать Pygame.
- Создать окно размером 800x600 пикселей.
- Установить заголовок окна (например, "Платформер").

2. Добавить фон и платформы:

- Загрузить фоновое изображение и отобразить его в окне.
- Создать несколько платформ (прямоугольников или спрайтов), на которых будет перемещаться персонаж.

3. Создать персонажа:

- Загрузить спрайт персонажа.
 - Реализовать управление персонажем с помощью клавиатуры (движение влево/вправо, прыжок).
 - Добавить гравитацию, чтобы персонаж падал, если не находится на платформе.
- 4. Реализовать коллизии:**
- Проверять столкновения персонажа с платформами.
 - Обеспечить, чтобы персонаж мог стоять на платформах и не проваливался сквозь них.
- 5. Добавить завершение уровня:**
- Создать объект "финиш" (например, дверь или флаг).
 - При столкновении персонажа с этим объектом выводить сообщение об успешном завершении уровня.
- 6. Дополнительно (по желанию):**
- Добавить звуковые эффекты для прыжков и фоновую музыку.
 - Создать анимацию для персонажа (ходьба, прыжок).
 - Добавить врагов или собираемые предметы (монеты, бонусы).

Практическая работа 2. Игра "Змейка" с использованием Pygame

Цель работы: Освоить навыки работы с библиотекой Pygame, изучить основы игровой логики и реализовать классическую игру "Змейка". Научиться управлять объектами, обрабатывать пользовательский ввод и реализовывать игровую механику.

Исходные данные: Установленная библиотека Pygame. Простые графические элементы (квадраты или круги для змейки и еды). Звуковые эффекты для поедания еды и завершения игры (опционально).

Требуется:

- 1. Создать игровое окно:**
 - Инициализировать Pygame.
 - Создать окно размером 600x600 пикселей.
 - Установить заголовок окна (например, "Змейка").
- 2. Реализовать змейку:**
 - Создать начальное тело змейки (например, три квадрата или круга).
 - Реализовать управление змейкой с помощью клавиатуры (вверх, вниз, влево, вправо).
 - Обеспечить движение змейки с постоянной скоростью.
- 3. Добавить еду:**
 - Создать объект "еда", который будет случайно появляться на игровом поле.

- Реализовать механику поедания еды: при столкновении головы змейки с едой увеличивать длину змейки и перемещать еду в новую случайную позицию.

4. Реализовать игровую логику:

- Добавить проверку столкновений змейки с границами игрового поля и с самой собой.
- При столкновении завершать игру и выводить сообщение о проигрыше.
- Добавить счетчик очков, который увеличивается при поедании еды.

5. Дополнительно (по желанию):

- Добавить звуковые эффекты для поедания еды и завершения игры.
- Реализовать увеличение скорости змейки с ростом ее длины.
- Добавить меню с возможностью перезапуска игры.

6.2. Описание показателей и критериев контроля успеваемости, описание шкал оценивания

Для оценки знаний, умений, навыков и формирования компетенции по дисциплине применяется **балльно-рейтинговая** система контроля и оценки успеваемости студентов.

В основу балльно-рейтинговой системы (БРС) положены принципы, в соответствии с которыми формирование рейтинга студента осуществляется в ходе текущей работы в семестре.

Работы должны быть выполнены по своему варианту, оформлены в соответствии с требованиями стандартов по оформлению текстовых документов в текстовом редакторе MS Word. Работы сдаются в электронном виде.

По результатам защиты могут быть получены следующие баллы:

9-10 баллов – расчеты (если имеются) проведены корректно, результаты правильно интерпретированы. Полностью выполнены все пункты выданного задания. Работа оформлена в соответствии с требованиями стандартов по оформлению текстовых документов. Студент развернуто и свободно ответил на все вопросы при защите работы.

7-8 баллов – работа выполнена, выполнены все пункты выданного задания, но не полностью, либо с несущественными ошибками, имеются незначительные ошибки в интерпретации результатов и/или оформлении. Студент в целом ответил на все поставленные вопросы, ориентируется в работе.

4-6 баллов – работа в целом выполнена, выполнены основные, но не все пункты выданного задания, либо с существенными ошибками, имеются значительные ошибки в интерпретации полученных результатов и представления данных, оформления работы. Некоторые вопросы по работе вызывают затруднения.

1-3 балла – имеются грубые ошибки в методике выполнения, интерпретации полученных результатов и представления данных, оформления

работы, большая часть пунктов выданного задания не выполнена. Студент не отвечает на вопросы при защите.

Зачет по дисциплине получают студенты, набравшие не менее 60% от максимального количества баллов.

7. Учебно-методическое и информационное обеспечение дисциплины

7.1 Основная литература

1. Сергеева, О. А. Программирование на Python : учебно-методическое пособие / О. А. Сергеева. — Кемерово : КемГУ, 2025. — 157 с. — ISBN 978-5-8353-3123-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/420758> (дата обращения: 26.08.2025). — Режим доступа: для авториз. пользователей.

2. Рагимханова, Г. С. Программирование на Python : учебное пособие / Г. С. Рагимханова. — Махачкала : ДГПУ, 2022. — 126 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/330071> (дата обращения: 26.08.2025). — Режим доступа: для авториз. пользователей.

3. Гегечкори, Е. Т. Программирование на языке Python : учебное пособие / Е. Т. Гегечкори. — Омск : ОмГТУ, 2023. — 172 с. — ISBN 978-5-8149-3617-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/421673> (дата обращения: 26.08.2025). — Режим доступа: для авториз. пользователей.

7.2 Дополнительная литература

1. Костер, Р. Разработка игр и теория развлечений / Р. Костер ; перевод с английского О. В. Готлиб. — Москва : ДМК Пресс, 2018. — 288 с. — ISBN 978-5-97060-478-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/111430> (дата обращения: 26.08.2025). — Режим доступа: для авториз. пользователей.

2. Аксёнова, Н. А. Компьютерная графика : учебно-методическое пособие / Н. А. Аксёнова, А. В. Воруев, О. М. Демиденко. — Гомель : ГГУ имени Ф. Скорины, 2023. — 130 с. — ISBN 978-985-577-917-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/329723> (дата обращения: 26.08.2025). — Режим доступа: для авториз. пользователей.

3. Изюмов, А. А. Информационные технологии : учебное пособие / А. А. Изюмов, В. П. Коцубинский, А. О. Шатохина. — Москва : ТУСУР, 2023. — 240 с. — ISBN 978-5-7511-2656-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/394139> (дата обращения: 26.08.2025). — Режим доступа: для авториз. пользователей.

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Официальная документация Pygame: <https://www.pygame.org/docs/>
2. Документация библиотеки Arcade: <https://api.arcade.academy/>
3. Документация Ursina Engine: <https://www.ursinaengine.org/>
4. GeeksforGeeks: Pygame Tutorials: <https://www.geeksforgeeks.org/pygame-tutorial/>
5. Medium: Game Development with Python: <https://medium.com/>
6. Python Programming: Pygame Tutorials: <https://pythonprogramming.net/>
7. GitHub: Pygame Projects: <https://github.com/search?q=pygame>
8. Repl.it: Pygame Examples: <https://replit.com/>
9. CodePen: Pygame Snippets: <https://codepen.io/>

9. Перечень программного обеспечения и информационных справочных систем

Таблица 9

Перечень программного обеспечения

№ п/п	Наименование раздела учебной дисциплины	Наименование программы	Тип программы	Автор	Год разработки
1	Разделы 1, 2, 3	Python	расчетная, обучающая, контролирующая	Python Software Foundation	Текущая версия
3	Разделы 1, 2, 3	Pygame	библиотека для разработки 2D-игр	-	Текущая версия
4	Разделы 1, 2, 3	Arcade	альтернативная библиотека для 2D-игр	-	Текущая версия
5	Разделы 1, 2, 3	Ursina Engine	библиотека для 3D-игр	-	Текущая версия

10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Для проведения практических занятий нужен компьютерный класс с доступом в «Интернет», оснащенный программным обеспечением в соответствии с разделом 9.

Таблица 10

Сведения об обеспеченности специализированными аудиториями, кабинетами, лабораториями

Наименование специальных помещений и помещений для самостоятельной работы (№ учебного корпуса, № аудитории)	Оснащенность специальных помещений и помещений для самостоятельной работы
1	2
учебная аудитория для проведения занятий лекционного типа, учебная аудитория для проведения занятий семинарского типа, учебная аудитория для проведения курсового проектирования (выполнения курсовых работ), учебная аудитория для групповых и индивидуальных консультаций, учебная аудитория для текущего контроля и промежуточной аттестации (2й учебный корпус, 102 ауд.)	1. Компьютер – 29 шт.; 2. Стенд «Сергеев Сергей Степанович 1910-1999» (Инв.№591013/25) – 1 шт.; 3. Огнетушитель порошковый (Инв. №559527) – 1 шт.; 4. Подвесное крепление к огнетушителю (Инв. № 559528) – 1 шт.; 5. Жалюзи (Инв. №1107-221225, Инв. №1107-221225) – 2шт.; 6. Стул – 29 шт.; 7. Стол компьютерный – 28 шт.; 8. Стол для преподавателя – 1 шт.; 9. Доска маркерная (Инв. № 558762/5) – 1 шт.; 10. Трибуна напольная (без инв. №) – 1 шт.
учебная аудитория для проведения занятий семинарского типа, учебная аудитория для проведения курсового проектирования (выполнения курсовых работ), учебная аудитория для групповых и индивидуальных консультаций, учебная аудитория для текущего контроля и промежуточной аттестации, помещение для самостоятельной работы (2й учебный корпус, 106 ауд.)	1. Рабочая станция FORSITE TH1516G512G, Российская Федерация A4Tech Fstyler F1512 – 16 шт.; 2. Стол наборный (Инв. №410136000010828) – 1 шт. 3. Стол компьютерный (Инв. № 410136000010813-410136000010827) – 15 шт.; 4. Стул (Инв. № 410136000010829-410136000010853) – 25 шт.; 5. Интерактивная панель (Инв. № 410124000603715) – 1 шт.
учебная аудитория для проведения занятий семинарского типа, учебная аудитория для проведения курсового проектирования (выполнения курсовых работ), учебная аудитория для групповых и индивидуальных консультаций, учебная аудитория для текущего контроля и промежуточной аттестации, помещение для самостоятельной работы (2й учебный корпус, 302 ауд.)	1. Компьютер – 16 шт. 2. Телевизор – 1 шт. 3. Стол для преподавателя – 1 шт. 4. Стол компьютерный – 16 шт. 5. Стул офисный – 17 шт. 6. Компьютер: PRO-3159209 Intel Core i5-10400 2900МГц, Intel B460, 16Гб DDR4, Intel UHD Graphics 630 (встроенная), SSD 240Гб, 500Вт, Mini-Tower – 1 шт. 7. Кондиционер HAIER HSU -24HPL03/R3 (Инв. № 210134000062198) – 1 шт. 8. Вешалка напольная (Инв.№1107-333144, Инв.№1107-333144) – 2 шт.
учебная аудитория для проведения занятий лекционного типа, учебная аудитория для проведения занятий семинарского типа, учебная аудитория для проведения курсового проектирования (выполнения курсовых работ), учебная аудитория для групповых и индивидуальных консультаций, учебная аудитория для текущего контроля и промежуточной	1. Трибуна напольная (Инв.№ 599206) – 1 шт.; 2. Жалюзи (Инв.№591110) – 1 шт.; 3. Доска маркетинговая (Инв.№ 35643/4) – 1 шт.; 4. Стол – 15 шт.; 5. Скамейка – 14 шт.; 6. Стол эрго – 1 шт.; 7. Стул – 16 шт.

<i>аттестации, помещение для самостоятельной работы (2й учебный корпус, 303 ауд.)</i>	
<i>Центральная научная библиотека имени Н.И. Железнова</i>	Читальные залы библиотеки
<i>Студенческое общежитие</i>	Комната для самоподготовки

11. Методические рекомендации обучающимся по освоению дисциплины

Предполагается, что студент выполняет практическое задание в аудитории, дома оформляет и готовится по теоретическим вопросам к защите отчета на следующем занятии.

Виды и формы отработки пропущенных занятий

Студент, пропустивший занятие, обязан предъявить преподавателю документы установленного образца, подтверждающие необходимость пропуска. Не допускается пропуск занятий без уважительной причины.

Студент, пропустивший занятия, осваивает материал самостоятельно (выполняет практическое задание по своему варианту в компьютерном классе кафедры в часы, свободные от занятий, изучает теоретические вопросы).

Студент, пропустивший лекцию, отвечает на вопросы по пропущенной теме.

12. Методические рекомендации преподавателям по организации обучения по дисциплине

Курс «Разработка игр на Python» должен давать не абстрактно-формальные, а прикладные знания. Данная цель может быть реализована только при условии соблюдения в учебных планах преемственности учебных дисциплин. Базовые знания для изучения «Разработка игр на Python» являются «Введение в компьютерные науки на иностранном языке», «Разработка профессиональных приложений», «Алгоритмизация и программирование», «Программирование на языке Python». Освоение основных тем данной дисциплины позволит студентам сформировать представление о таком сложном предмете как Большие данные в сельском хозяйстве, понять всю ширину науки и получить необходимые знания для последующего профессионального развития в этой области.

Студент может подготовить доклад по теме, представляющей его научный интерес, представить результаты в виде презентации. В случае надлежащего качества, его работа может быть заслушана на научном кружке кафедры или на студенческой научной конференции. По решению кафедры, студенты, занявшие

призовые места на научных студенческих конференциях, могут освобождаться от сдачи экзамена по этой дисциплине.

Преподаватель должен указывать, в какой последовательности следует изучать материал дисциплины, обращать внимание на особенности изучения отдельных тем и разделов, помогать отбирать наиболее важные и необходимые сведения из учебных пособий, а также давать объяснения вопросам программы курса, которые обычно вызывают затруднения. При этом преподавателю необходимо учитывать следующие моменты:

1. Не следует перегружать студентов творческими заданиями.
2. Чередовать творческую работу на занятиях с заданиями во внеаудиторное время.
3. Давать студентам четкий инструктаж по выполнению самостоятельных заданий: цель задания; условия выполнения; объем; сроки; требования к оформлению.
4. Осуществлять текущий учет и контроль за самостоятельной работой.
5. Давать оценку и обобщать уровень усвоения навыков самостоятельной, творческой работы.

Программу разработали:

Демичев В.В., кандидат экономических наук, доцент,

Невзоров А.С., старший преподаватель



РЕЦЕНЗИЯ

на рабочую программу дисциплины «Разработка игр на Python»
ОПОП ВО по направлению 09.03.02 «Информационные системы и технологии»,
направленность «Фуллстек разработка», «Системная аналитика и разработка
программного обеспечения» (квалификация выпускника – бакалавр)

Худяковой Елены Викторовны, профессором кафедры прикладной информатики, кандидатом экономических наук (далее по тексту рецензент), проведено рецензирование рабочей программы дисциплины «Разработка игр на Python» ОПОП ВО по направлению 09.03.02 – «Информационные системы и технологии», направленность «Фуллстек разработка», «Системная аналитика и разработка программного обеспечения», разработанной в ФГБОУ ВО «Российский государственный аграрный университет – МСХА имени К.А. Тимирязева», на кафедре статистики и кибернетики (разработчики – Демичев Вадим Владимирович, доцент, кандидат экономических наук, Невзоров Александр Сергеевич, старший преподаватель кафедры статистики и кибернетики)

Рассмотрев представленные на рецензию материалы, рецензент пришел к следующим выводам:

1. Предъявленная рабочая программа дисциплины «Разработка игр на Python» (далее по тексту Программа) соответствует требованиям ФГОС ВО по направлению 09.03.02 – «Информационные системы и технологии». Программа содержит все основные разделы, соответствует требованиям к нормативно-методическим документам.

2. Представленная в Программе актуальность учебной дисциплины в рамках реализации ОПОП ВО не подлежит сомнению – дисциплина относится к факультативной части учебного цикла – ФТД.

3. Представленные в Программе цели дисциплины соответствуют требованиям ФГОС ВО направления 09.03.02 – «Информационные системы и технологии».

4. В соответствии с Программой за дисциплиной «Разработка игр на Python» закреплена 1 профессиональная компетенция, определяемые самостоятельно (**3 индикатора**). Дисциплина «Разработка игр на Python» и представленная Программа способна реализовать их в объявленных требованиях. Результаты обучения, представленные в Программе в категориях знать, уметь, владеть соответствуют специфике и содержанию дисциплины и демонстрируют возможность получения заявленных результатов.

5. Общая трудоёмкость дисциплины «Разработка игр на Python» составляет 4 зачётных единиц (144 часа, в т.ч. 8 часов практической подготовки).

6. Информация о взаимосвязи изучаемых дисциплин и вопросам исключения дублирования в содержании дисциплин соответствует действительности. Дисциплина «Разработка игр на Python» взаимосвязана с другими дисциплинами ОПОП ВО и Учебного плана по направлению 09.03.02 – «Информационные системы и технологии» и возможность дублирования в содержании отсутствует.

7. Представленная Программа предполагает использование современных образовательных технологий, используемые при реализации различных видов учебной работы. Формы образовательных технологий соответствуют специфике дисциплины.

8. Программа дисциплины «Разработка игр на Python» предполагает проведение занятий в интерактивной форме.

9. Виды, содержание и трудоёмкость самостоятельной работы студентов, представленные в Программе, соответствуют требованиям к подготовке выпускников, содержащимся во ФГОС ВО направления 09.03.02 – «Информационные системы и технологии».

10. Представленные и описанные в Программе формы текущей оценки знаний (выполнение и защита практических заданий), соответствуют специфике дисциплины и требованиям к выпускникам.

Форма промежуточного контроля знаний студентов, предусмотренная Программой, осуществляется в форме зачета, что соответствует статусу дисциплины, как дисциплины

факультативной части учебного цикла – ФТД. ФГОС ВО направления 09.03.02 – «Информационные системы и технологии».

11. Формы оценки знаний, представленные в Программе, соответствуют специфике дисциплины и требованиям к выпускникам.

12. Учебно-методическое обеспечение дисциплины представлено: основной литературой – 3 источника (базовый учебник), дополнительной литературой – 3 наименований, Интернет-ресурсы – 9 источников и соответствует требованиям ФГОС ВО направления 09.03.02 – «Информационные системы и технологии».

13. Материально-техническое обеспечение дисциплины соответствует специфике дисциплины «Разработка игр на Python» и обеспечивает использование современных образовательных, в том числе интерактивных методов обучения.

14. Методические рекомендации студентам и методические рекомендации преподавателям по организации обучения по дисциплине дают представление о специфике обучения по дисциплине «Разработка игр на Python».

ОБЩИЕ ВЫВОДЫ

На основании проведенной рецензии можно сделать заключение, что характер, структура и содержание рабочей программы дисциплины «Разработка игр на Python» ОПОП ВО по направлению 09.03.02 «Информационные системы и технологии», направленности «Системная аналитика и разработка программного обеспечения», «Фуллстек разработка» (квалификация выпускника – бакалавр), разработанная Демичевым Вадимом Владимировичем, к.э.н., доцентом кафедры статистики и кибернетики, Невзоровым Александром Сергеевичем, старшим преподавателем кафедры статистики и кибернетики, соответствует требованиям ФГОС ВО, современным требованиям экономики, рынка труда и позволит при её реализации успешно обеспечить формирование заявленных компетенций.

Рецензент:

Худякова Елена Викторовна, профессор кафедры прикладной информатики, доктор экономических наук



(подпись)

«26» августа 2025 г.