

**Автономная некоммерческая организация
дополнительного профессионального образования
«Центральный многопрофильный институт
профессиональной переподготовки и повышения квалификации»
АНО ДПО «ЦМИ»**



УТВЕРЖДАЮ
Директор АНО ДПО «ЦМИ»
А.В. Гриднева
«18» апреля 2023г.

**Дополнительная общеобразовательная
общеразвивающая программа
«Python: разработка игр с Pygame и GUI»**


*технической направленности
уровень сложности «Базовый»*

144 часа

Ставрополь 2023

**Общие данные о Дополнительной общеобразовательной программе
«Python: разработка игр с Pygame и GUI»**

Об организации

Наименование поля	Допустимые значения поля	Значение поля
ИНН организации, осуществляющей образовательную деятельность	10 арабских цифр	2632108764
Наименование организации	строка	Автономная некоммерческая организация дополнительного профессионального образования «Центральный многопрофильный институт профессиональной переподготовки и повышения квалификации» АНО ДПО «ЦМИ»
Логотип организации		
Ссылка на логотип организации		https://drive.google.com/file/d/1nobPZvgUGUtrHOVXT3yDlPdLZHXTB7nc/view?usp=drive_link
Контакты ответственного за программу (с указанием фамилии, имени, отчества).	строка от 5 до 255 символов	Гончарова Ирина Сергеевна

Контакты ответственного за программу. Должность	строка от 5 до 255 символов	РОП
Контакты ответственного за программу. Телефон	Формат +7(XXX)XXXXXXXX	+7 9296469339
Контакты ответственного за программу. E-mail	строка	Umo_sk@mail.ru

Информация о программе

Наименование поля	Допустимые значения поля	Значение поля (примеры)
Название программы (курса)	строка	Python: разработка игр с Pygame и GUI
Описание программы	строка не менее 1000 не более 5000 символов	<p>Дополнительная общеобразовательная программа «Python: разработка игр с Pygame и GUI» является неотъемлемой частью образовательной программы автономной некоммерческой организации дополнительного профессионального образования «Центральный многопрофильный институт профессиональной переподготовки и повышения квалификации» и дает возможность каждому ребенку получать дополнительное образование исходя из его интересов, склонностей и способностей, образовательных потребностей, осуществляемых за пределами федеральных государственных образовательных стандартов и федеральных государственных требований. По своему функциональному назначению программа является общеразвивающей и направлена на удовлетворение потребностей, обучающихся в интеллектуальном, нравственном совершенствовании, в организации их свободного времени.</p> <p>Направленность общеобразовательной программы «Python: разработка игр с Pygame и GUI» техническая.</p>

<p>Аннотация (для размещения на маркетплейсе, понятное и привлекательное для Потенциальных получателей поддержки, включающее полное и содержательное описание Дополнительной общеобразовательной программы:</p> <p>1) краткое описание Дополнительной общеобразовательной программы;</p> <p>2) описание требований и рекомендаций для обучения по образовательной Дополнительной общеобразовательной программе;</p> <p>3) краткое описание результатов обучения в свободной форме, включая описание практикоориентированного характера Дополнительной общеобразовательной программы)</p>	<p>строка до 1000 символов</p>	<p>Изучение Python, Pygame и GUI предоставляет детям уникальные возможности для разработки своих навыков в программировании, разработке игр и GUI. Этот курс для начинающих программистов, которые хотят изучить язык Python и создать игры с помощью библиотеки Pygame и GUI, поможет детям освоить основы программирования на Python, работу с графикой, звуком и управлением игры, что даст им широкие возможности в будущей карьере в сфере разработки приложений. Требования к ученикам (уровень знаний):</p> <ul style="list-style-type: none"> ● Базовые навыки владения компьютером (запуск приложений, создание папок, смена имени файла, скачивание картинок из интернета). ● Начальные знания английского языка и умение бегло печатать на английской раскладке. <p>Результат обучения:</p> <ul style="list-style-type: none"> ● Уверенное владение языком Python и его основными конструкциями, такими как условные операторы, циклы и функции. ● Понимание принципов программирования игр и создание собственных игр на Python с помощью библиотеки Pygame. ● Навыки работы с графическим интерфейсом пользователя (GUI) и создание собственных приложений с помощью библиотеки Tkinter. ● Умение создавать и настраивать графические элементы интерфейса, такие как кнопки, поля ввода и метки. ● Навыки обработки пользовательского ввода и отображения результатов на экране. ● Понимание принципов организации и структурирования кода, а также методологии разработки программных проектов. ● Уверенность в работе с командной строкой и управлении файлами в операционной системе.
--	--------------------------------	--

Цель программы	строка не менее 100 СИМВОЛОВ	Цель обучения Python, Pygame и GUI заключается в том, чтобы дать начинающим программистам возможность овладеть основными принципами программирования на языке Python, освоить создание игр с использованием библиотеки Pygame и научиться работать с графическим интерфейсом пользователя (GUI).
Актуальность	строка не менее 500 СИМВОЛОВ	<p>Изучение Python, Pygame и GUI является актуальным в настоящее время из-за широкого распространения Python как языка программирования, его легкости изучения и использования, а также многих возможностей, которые он предоставляет в области разработки программного обеспечения.</p> <p>Pygame является мощной библиотекой для разработки игр на языке Python, которая позволяет создавать игры с графическим интерфейсом пользователя и работать со звуком и анимацией. Это делает ее популярным инструментом для создания игр на языке Python.</p> <p>GUI (графический интерфейс пользователя) также является важным аспектом изучения Python, так как он позволяет создавать приложения, которые могут быть легко взаимодействовать с пользователем. Многие приложения, такие как программы управления базами данных, текстовые редакторы и браузеры, используют GUI.</p> <p>Кроме того, Python и Pygame являются открытыми и доступными для всех, что делает их популярными инструментами в обучении программированию и в качестве стартовой точки для новых программистов.</p>

		<p>Таким образом, изучение Python, Pygame и GUI является актуальным в настоящее время и имеет большое значение для разработчиков программного обеспечения и начинающих программистов. Изучение Python и Pygame может быть полезным не только для создания игр, но и для различных проектов, связанных с графикой и визуализацией данных. Кроме того, Python и Pygame являются открытыми и бесплатными, что делает их доступными для любого желающего изучить программирование.</p> <p>Таким образом, изучение Python и его библиотеки Pygame является актуальным и востребованным направлением в программировании, особенно для начинающих программистов, которые хотят создавать игры и графические приложения.</p>
Дополнительная информация	строка	<p>Педагогическая целесообразность программы заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования. Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Для достижения поставленных задач занятия проводятся в формате «от простого к сложному»</p>

Формат обучения	значение из: очная форма без применения дистанционных образовательных технологий; очная форма с применением дистанционных образовательных технологий, в том числе, с применением средств электронного обучения	Очная форма без применения дистанционных образовательных технологий; Очная форма с применением дистанционных образовательных технологий, в том числе, с применением средств электронного обучения.
Уровень сложности	значение из: «Начальный» «Базовый» «Продвинутый»	Базовый
Срок освоения образовательной программы	строка, значение в ак.ч.	144 ак.ч.
Объем каждого модуля в ак.ч.	целое число	36
Объем часов в неделю в ак.ч.	целое число	4
Количество занятий	целое число	58
Направленность программы	строка	Техническая
Язык программирования	строка	Язык программирования Python
Дополнительная общеобразовательная программа не представлена для участия в иных федеральных проектах,	строка, значения: «Не представлена»	Не представлена

направленных на дополнительное образование граждан, кроме федерального проекта «Развитие кадрового потенциала ИТ-отрасли»		
Дополнительная общеобразовательная программа не была реализована до начала отбора и/или не реализовывается в период отбора на безвозмездной основе	строка, значения «Не реализована»	Не реализована
Категория обучающихся по программе	строка не менее 10 символов	Учащиеся 8 класса Учащиеся 9 класса Учащиеся 10 класса Учащиеся 11 класса Обучающиеся по программам среднего профессионального образования
Описание планируемых результатов обучения	строка не менее 10 символов	Сформировать у обучающихся базовые знания и умения языка программирования Python, Сформировать умение применять программные средства для решения задач из различных предметных областей способствует развитию логического и комбинаторного мышления
Ссылка на лендинг Образовательной программы	строка не менее 10 символов	Курс «Python: разработка игр с Pygame и GUI» — онлайн-обучение бесплатно
Ссылка на LMS	строка не менее 10 символов	https://odin.study/ru/
Страница обучения на курсе	строка не менее 10 символов	Python: разработка игр с Pygame и GUI (odin.study)

Аттестация

Промежуточная аттестация		
Количество академических часов	строка не менее 10 символов	4 часа, текущая работа, включающая выполнение 4 практических заданий
Формы контроля	строка не менее 10 символов	Для успешного завершения обучения по курсу необходимо, чтобы 50 % заданий в рамках текущей работы были зачтены (оценены преподавателем на 3, 4 или 5 баллов).
Диагностические инструменты	строка не менее 10 символов	Оценка полноты выполнения работы. Оценка способности оперировать полученными знаниями и умениями при решении практической задачи. Проверка результата запуска программного решения.
Показатели и критерии оценивания	строка не менее 10 символов	<p>При оценке надо руководствоваться следующими критериями:</p> <ol style="list-style-type: none"> 1) полнота и правильность выполнения заданий; 2) степень осознанности, понимания изученного; 3) оформление ответа. <p>Оценка «5» ставится, если:</p> <ol style="list-style-type: none"> 1) Практическое задание выполнено без ошибок, полно изложен изученный материал; 2) учащийся понимает материал, может обосновать свои суждения и действия, хорошо применяет знания на практике, приводит необходимые примеры не только по учебному материалу, но и самостоятельно составленные. <p>Оценка «4» ставится, если ученик выполнил практическое задание, удовлетворяющее тем же требованиям, что и для оценки «5», но допускает 1—2 ошибки, которые сам же исправляет, и 1—2 недочета в последовательности излагаемого материала.</p>

		<p>Оценка «3» ставится, если ученик освоил знания и понимает основы практического задания, но:</p> <ol style="list-style-type: none"> 1) подготовил материал неполно и допускает неточности в последовательности выполнения заданий; 2) не может обосновать свои суждения и привести свои примеры; 3) допускает ошибки в оформлении. <p>Оценка «2» ставится, если ученик обнаруживает незнание большей части изучаемого материала, допускает ошибки в написании кода, беспорядочно и неуверенно излагает материал.</p> <p>Оценка «2» отмечает такие недостатки в подготовке ученика, которые являются серьезным препятствием к успешному овладению последующим материалом.</p>
Примеры заданий	строка не менее 10 символов	<p>Модуль 1. Практическое задание № 1. Введение в язык программирования Python</p> <p>Шаг 1: Установка Python</p> <p>Первым шагом необходимо установить интерпретатор языка Python на ваш компьютер. Это можно сделать, загрузив установочный файл Python с официального сайта Python (https://www.python.org/downloads/) и следуя инструкциям по установке.</p> <p>Шаг 2: Интерпретатор Python</p> <p>После установки Python на вашем компьютере, вы можете запустить интерпретатор Python. Это можно сделать, открыв терминал (командную строку) и вводя команду "python". После этого появится приглашение для ввода команд.</p> <p>Шаг 3: Изучение синтаксиса Python</p> <p>Синтаксис Python относительно простой и понятный, что делает язык очень доступным для новичков. Однако перед тем, как начать писать код на Python, важно изучить основные элементы синтаксиса, такие как комментарии, переменные, операторы, условные операторы и циклы.</p> <p>Шаг 4: Первая программа на Python</p>

После того, как вы изучили основы синтаксиса Python, вы можете написать свою первую программу. Обычно, первая программа на Python, это программа, которая выводит на экран сообщение "Hello, World!".

Шаг 5: Создание функций. В Python вы можете создавать функции, чтобы организовывать повторяющийся код в отдельные блоки. Функция начинается с ключевого слова "def", за которым следует имя функции и ее параметры. Затем следует блок кода, который будет выполнен при вызове функции. Выведите надпись "Hello, World!" с помощью функции.

Шаг 6: Использование условных операторов

Условные операторы позволяют вам создавать условия в вашей программе, и выполнять различный код, в зависимости от выполнения условия. В Python условный оператор представлен ключевым словом "if". Напишите пример, который проверяет, является ли число четным.

Шаг 7. Сохраните файлы и выложите их в Odin

Модуль 2. Практическое задание № 2. Создание графических пользовательских интерфейсов (GUI)

Шаг 1. Открыть среду разработки Idle

Шаг 2. Создать новый проект

Шаг 3. Создание простого калькулятора: создайте GUI-приложение, которое имеет поле для ввода чисел и кнопки для выполнения математических операций, таких как сложение, вычитание, умножение и деление. После ввода чисел пользователь должен иметь возможность нажать на кнопку, чтобы выполнить выбранную операцию и увидеть результат в приложении.

Шаг 4. Сохраните файл и выложите его в Odin

Модуль 3. Практическое задание № 3. Создание игр на Python с использованием Pygame

Шаг 1. Открыть среду разработки Idle

Шаг 2. Создать новый проект

		<p>Шаг 3. «Арканоид»: создайте игру, где пользователь управляет платформой, которая ловит мяч и отбивает его, чтобы разрушить блоки на верхней части экрана. Игра заканчивается, когда все блоки разрушены или мяч падает вниз за пределы платформы.</p> <p>Шаг 4. Сохраните файл и выложите его в Odin/</p> <p>Модуль 4. Практическое задание № 4. Разработка плагинов и механизмов движения</p> <p>Шаг 1. Открыть среду разработки Idle</p> <p>Шаг 2. Создать новый проект</p> <p>Шаг 3. «Космический корабль»: создайте игру, где пользователь управляет космическим кораблем, который может летать в космосе и сражаться с врагами. Игрок должен управлять кораблем, чтобы избегать астероидов и атак врагов, стрелять в них и собирать ресурсы.</p> <p>Шаг 4. Сохраните файл и выложите его в Odin</p>
Шкала оценивания, нижнее значение	строка не менее 10 символов	2
Шкала оценивания, верхнее значение	строка не менее 10 символов	5
Шкала оценивания, минимальный проходной балл	строка не менее 10 символов	3

Преподаватели

Ф.И.О.	Наименование основного места работы	Должность	Высшее образование или среднее профессиональное образование по направлению - нию «Образование и педагогические науки»	Высшее образование или среднее профессиональное образование по иному направлению соответствующим направленности ДОП	Ссылка на веб-страницы с портфолио	Информация о курсах повышения квалификации по профилю преподаваемой дисциплины (за последние 3 года)	Пройдена промежуточная аттестация не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности ДОП	Отметка о полученном согласии на обработку персональных данных
Сергеев Олег Владимирович	ГБОУ Гимназия им. В.Маяковского 1274	Учитель информатики, технологии (робототехника)	да	да	https://docs.google.com/document/d/14H1queIfHj_ofd9NZhb_g8mfnnSUIM-zu/edit?usp=sharing&ouid=118085064277121638114&rtpof=true&sd=true	2023 год - Методика обучения технологии программирования объектов дополненной реальности в Unity 3D. Городской	да	да

						Методический Центр, Разработка на Unity 3D		
Патимат Захарова	Учитель информатик и и роботехник и, занятия для детей 8- 11 кл, СПО	Старший преподавател ь программиров ания	да	да	https://docs.google.com/document/d/10B7ScIRvBzQr0wTjKYyYuycPotesNDQW/edit#	нет	да	да

I ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная программа «Python: разработка игр с Pygame и GUI» является неотъемлемой частью образовательной программы Автономной некоммерческой организации дополнительного профессионального образования «Центральный многопрофильный институт профессиональной переподготовки и повышения квалификации» АНО ДПО «ЦМИ» и дает возможность каждому ребенку получать дополнительное образование исходя из его интересов, склонностей и способностей, образовательных потребностей, осуществляемых за пределами федеральных государственных образовательных стандартов и федеральных государственных требований.

По своему функциональному назначению программа является общеразвивающей и направлена на удовлетворение потребностей, обучающихся в интеллектуальном, нравственном совершенствовании, в организации их свободного времени.

Направленность общеобразовательной программы «Python: разработка игр с Pygame и GUI» техническая.

Актуальность и педагогическая целесообразность программы

Python является одним из наиболее популярных языков программирования, который используется в различных областях, включая разработку игр и создание графических пользовательских интерфейсов. Pygame и GUI являются библиотеками Python, которые предоставляют разработчикам инструменты для создания игр и графических приложений с более простым и интуитивно понятным интерфейсом для пользователей.

Изучение Python, Pygame и GUI имеет множество преимуществ. Во-первых, Python является относительно простым и легко читаемым языком программирования, который позволяет новичкам быстро освоить основы программирования. Во-вторых, Pygame предоставляет разработчикам множество инструментов для создания игр, включая работу с изображениями, звуком и управлением игровыми объектами. В-третьих, GUI предоставляет возможность создавать графические пользовательские интерфейсы с помощью визуальных инструментов, что делает процесс разработки более быстрым и эффективным.

Изучение Python, Pygame и GUI является актуальным и востребованным в современном мире, так как все больше компаний и организаций используют эти инструменты для создания игр, приложений и веб-сервисов. Кроме того, развитие и использование искусственного интеллекта, машинного обучения и анализа данных также требует знания Python и его библиотек.

Цель и основные задачи программы: формирование первичных компетенций в области информационных технологий, создания игр, навыков исследовательской и изобретательской деятельности, проектной и командной работы.

Основные задачи:

Образовательные:

- Овладение основными концепциями и принципами программирования на языке Python.
- Приобретение навыков работы с библиотекой Pygame для разработки компьютерных игр.

- Освоение принципов работы с графическим интерфейсом пользователя (GUI) и создание приложений с помощью соответствующих библиотек.
- Понимание методов и приемов работы с графикой, звуком и управлением игровым процессом.
- Разработка собственных игр и приложений с использованием полученных знаний и навыков.

Воспитательные:

- воспитать мотивацию учащихся к изобретательству, созданию собственных игр;
- привить стремление к получению качественного законченного результата в проектной деятельности;
- формировать правильное восприятие системы ценностей, принципов, правил информационного общества;
- формировать потребность в самостоятельном приобретении и применении знаний, потребность к постоянному саморазвитию;
- воспитывать социально-значимые качества личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность.

Развивающие:

- способствовать развитию творческих способностей учащихся, познавательных интересов, развитию индивидуальности и самореализации;
- расширять технологические навыки при подготовке различных информационных материалов;
- развивать познавательные способности ребенка, память, внимание, пространственное мышление, аккуратность и изобретательность при выполнении учебных проектов;
- формировать творческий подход к поставленной задаче;
- развивать навыки инженерного мышления, умения работать как по предложенным инструкциям, так и находить свои собственные пути решения поставленных задач;
- формировать навыки рефлексивной деятельности.

Педагогические принципы, определяющие теоретические подходы к построению образовательного процесса

Реализация программы «Python: разработка игр с Pygame и GUI» основывается на принципах научности, последовательности, системности, связи теории с практикой, доступности.

В целях раскрытия педагогического и развивающего потенциала учебно-воспитательного процесса по программе акцент в ней делается на следующих принципах:

Принцип прочности предполагает стремление к тому, чтобы приобретенные знания, умения и навыки обучающихся были прочно закреплены, в первую очередь, практически.

Для этого необходимо их участие в соревнованиях, где полученные умения и навыки можно проверить на практике, в выездных мероприятиях, а также постоянное поддержание интереса обучающихся к занятиям.

Принцип проектности предполагает последовательную ориентацию всей деятельности педагога на подготовку и выведение ребенка в проектное действие, совершаемое в логике замысел – реализация – рефлексия. В ходе проектирования перед

человеком всегда стоит задача представить себе еще не существующее, но то, что он хочет, чтобы появилось в результате его активности.

Принцип активности предполагает необходимость строить учебный процесс таким образом, чтобы учащиеся не просто воспринимали предлагаемый материал, но и стремились закрепить полученные знания, умения и навыки, анализировали ошибки и достижения, решали поставленные задачи.

Основные характеристики образовательного процесса

Уровень освоения программы базовый, что предполагает освоение обучающимися специализированных знаний, обеспечение трансляции общей и целостной картины тематического содержания программы.

Форма обучения: очная форма без применения дистанционных образовательных технологий

Режим работы: занятия проводятся 2-3 раза в неделю по 2 академических часа, длительность одного академического часа – 45 минут.

Продолжительность образовательного процесса: начало занятий 01 октября 2023 г., завершение 31 мая 2024 г.

Объем учебных часов по программе 144 ак. часа.

Ожидаемые результаты освоения программы

Личностные:

- развита способность переносить (выдерживать) известные нагрузки в течение определенного времени, преодолевать трудности;
- умеет контролировать свои поступки;
- умеет воспринимать общие дела как свои собственные.

Метапредметные:

- развито умение представлять результаты своей работы окружающим, аргументировать свою позицию;
- развита способность правильно организовывать рабочее место и время для достижения поставленных целей.
- сформировано у учащихся умение работать в команде.

Предметные:

- сформированы практические и теоретические знания в области устройства и функционирования современных платформ;

- изучены основы алгоритмизации, построения алгоритмов;
- научился формулировать и анализировать алгоритмы;
- изучить основы программирования на языке Python
- научиться разрабатывать игры на Python с помощью библиотеки Pygame и Tkinter.

Данная программа адаптирована для занятий с учащимися среднего и старшего возраста, с различной степенью подготовки, включает теоретические и практические задания.

Входные требования к обучающимся: тестирование, базовое владение персональным компьютером.

Для занятий -наличие ПК, камеры, микрофона, стабильного доступа в интернет.

Календарный учебный график формируется непосредственно при реализации программы общеобразовательной программы «**Python: разработка игр с Pygame и GUI**».

Первый модуль с 01.10.2023 - 30.11.2023

Второй модуль: с 01.12.2023 - 31.01.2024

Третий модуль: с 01.02.2024 - 31.03.2024

Четвертый модуль: с 01.04.2024 - 31.05.2024

Занятия проводятся 2-3 раза в неделю по 2 академических часа, длительность одного академического часа – 45 минут.

II. СОДЕРЖАНИЕ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ

СОДЕРЖАНИЕ УЧЕБНОГО (ТЕМАТИЧЕСКОГО) ПЛАНА «Python: разработка игр с Pygame и GUI»

№	п/п	Наименование модулей (курсов)	Общая трудоемкость, ч.	Всего контактных часов		Контактные часы		СРС, ч.	Формы контроля
				Общее количество Лекций и ПЗ	ДЗ	Лекции	Практические занятия		
I		Модуль 1. Введение в ООП	36	30	1	9	21	6	Зачет
1.1		Тема 1.1 Объектно-ориентированное программирование	1	1		1			
1.2		Тема 1.2 Конструктор, деструктор	1	1		1			
1.3		Тема 1.3 Инкапсуляция	1	1		1			
1.4		Тема 1.4 Наследование	7	6		1	5	1	
1.5		Тема 1.5 Переопределение	7	6		1	5	1	
1.6		Тема 1.6 Lambda функции	4	3		1	2	1	
1.7		Тема 1.7 Полиморфизм	5	4		1	3	1	
1.8		Тема 1.8 Стандартные библиотеки Python	5	4		1	3	1	
1.9		Тема 1.9 Работа с файловой системой. Формат JSON	5	4	1	1	3	1	Зачет

2	Модуль 2. Создание графических пользовательских интерфейсов (GUI)	36	30	1	9	21	6	Зачет
2.1	Тема 2.1 Введение в графические пользовательские интерфейсы	2	2		1	1		
2.2	Тема 2.2 Использование библиотеки Tkinter для создания GUI	3	3		1	2		
2.3	Тема 2.3 Использование библиотеки PyQt для создания GUI	4	3		1	2	1	
2.4	Тема 2.4 Создание интерфейсов с использованием дизайнера пользовательского интерфейса Qt Designer	4	3		1	2	1	
2.5	Тема 2.5 Использование графических элементов в PyQt	4	3		1	2	1	
2.6	Тема 2.6 Выпадающие меню — Tkinter	4	4		1	3		
2.7	Тема 2.7 Tkinter — виджет OptionMenu	3	3		1	2		
2.8	Тема 2.8 Как установить PyQt для Python в Windows?	4	3		1	2	1	
2.9	Тема 2.9 Python — калькулятор возраста с использованием PyQt	4	3		1	2	1	
2.10	Тема 2.10 Создание браузера с вкладками с использованием PyQt5	4	3	1		3	1	Зачет
3	Модуль 3. Создание игр на Python с	36	30	1	10	20	6	Зачет

	использованием библиотеки Pygame							
3.1	Тема 3.1 Введение в Pygame	4	3		1	2	1	
3.2	Тема 3.2 Работа со звуком и аудио в Pygame	4	3		1	2	1	
3.3	Тема 3.3 Работа с графикой в Pygame	4	3		1	2	1	
3.4	Тема 3.4 Создание анимации в Pygame	4	3	1	1	2	1	Зачет
3.5	Тема 3.5 Работа с клавиатурой и мышью в Pygame	3	3		1	2		
3.6	Тема 3.6 Создание игровых механик в Pygame	4	4		1	3		
3.7	Тема 3.7 Создаем змейку в Pygame	4	4		1	3		
3.8	Тема 3.8 Создаем тетрис в Pygame	4	3		1	2	1	
3.9	Тема 3.9 Разработка концепции игры	5	4		2	2	1	
4	Модуль 4. Разработка игр с GUI и Pygame	36	30	1	9	21	6	Зачет
4.1	Тема 4.1 Игровой цикл	5	4		1	3	1	
4.2	Тема 4.2 Создание пользовательского интерфейса для игры с помощью Tkinter	4	4		2	2		
4.3	Тема 4.3 Интеграция пользовательского интерфейса с Pygame	5	4		1	3	1	
4.4	Тема 4.4 Интеграция пользовательского	4	3		1	2	1	

	интерфейса с Pygame							
4.5	Тема 4.5 Создание современной игры с использованием GUI и Pygame	5	4		1	3	1	
4.6	Тема 4.6 Работа со спрайтами для разработки игры в Pygame	6	5		2	3	1	
4.7	Тема 4.7 Создание игры «Стрелялка»	5	4	1	1	3	1	Зачет
5	Консультация	2	2			2		
	Итого	144	120	4	37	83	24	

III. ОЦЕНКА КАЧЕСТВА ОСВОЕНИЯ ПРОГРАММЫ

Условия прохождения курса

В курсе будут оцениваться:

Текущая работа, включающая выполнение 4 практических задания.

Задания будут оцениваться по пятибалльной шкале:

Шкала оценивания:

Нижнее значение 2

Верхнее значение 5

Минимальный проходной балл для успешной сдачи 3

При оценке надо руководствоваться следующими критериями:

- 1) полнота и правильность выполнения заданий;
- 2) степень осознанности, понимания изученного;
- 3) оформление ответа.

Оценка «5» ставится, если:

- 1) практическое задание выполнено без ошибок, полно изложен изученный материал;
- 2) учащийся понимает материал, может обосновать свои суждения и действия, хорошо применяет знания на практике, приводит необходимые примеры не только по учебному материалу, но и самостоятельно составленные.

Оценка «4» ставится, если ученик выполнил практическое задание, удовлетворяющее тем же требованиям, что и для оценки «5», но допускает 1—2 ошибки, которые сам же исправляет, и 1—2 недочета в последовательности излагаемого материала.

Оценка «3» ставится, если ученик освоил знания и понимает основы практического задания, но:

- 1) подготовил материал неполно и допускает неточности в последовательности выполнения заданий;
- 2) не может обосновать свои суждения и привести свои примеры;
- 3) допускает ошибки в оформлении.

Оценка «2» ставится, если ученик обнаруживает незнание большей части изучаемого материала, допускает ошибки в написании кода, беспорядочно и неуверенно излагает материал.

Оценка «2» отмечает такие недостатки в подготовке ученика, которые являются серьезным препятствием к успешному овладению последующим материалом.

Для успешного завершения обучения по курсу необходимо, чтобы 50 % заданий в рамках текущей работы были зачтены (оценены преподавателем на 3, 4 или 5 баллов).

№ п/п	Мероприятия промежуточного контроля	Количество баллов
1.	Практическое задание № 1. Введение в язык программирования Python	3-5
2.	Практическое задание № 2. Создание графических пользовательских интерфейсов (GUI)	3-5
3.	Практическое задание № 3. Создание игр на Python с использованием Pygame	3-5
4.	Практическое задание № 4. Разработка игр с GUI и Pygame	3-5

В таблице представлен перечень заданий, которые необходимо выполнить в период обучения на курсе.

Формы проведения аттестации: зачетная работа.

Оценочные материалы для текущего контроля успеваемости:

Модуль 1. Практическое задание № 1. Введение в язык программирования Python

Шаг 1: Установка Python

Первым шагом необходимо установить интерпретатор языка Python на ваш компьютер. Это можно сделать, загрузив установочный файл Python с официального сайта Python (<https://www.python.org/downloads/>) и следуя инструкциям по установке.

Шаг 2: Интерпретатор Python

После установки Python на вашем компьютере, вы можете запустить интерпретатор Python. Это можно сделать, открыв терминал (командную строку) и вводя команду "python". После этого появится приглашение для ввода команд.

Шаг 3: Изучение синтаксиса Python

Синтаксис Python относительно простой и понятный, что делает язык очень доступным для новичков. Однако перед тем, как начать писать код на Python, важно изучить основные элементы синтаксиса, такие как комментарии, переменные, операторы, условные операторы и циклы.

Шаг 4: Первая программа на Python

После того, как вы изучили основы синтаксиса Python, вы можете написать свою первую программу. Обычно, первая программа на Python, это программа, которая выводит на экран сообщение "Hello, World!".

Шаг 6: Создание функций. В Python вы можете создавать функции, чтобы организовывать повторяющийся код в отдельные блоки. Функция начинается с ключевого слова "def", за которым следует имя функции и ее параметры. Затем следует блок кода, который будет выполнен при вызове функции. Выведите надпись "Hello, World!" с помощью функции.

Шаг 7: Использование условных операторов

Условные операторы позволяют вам создавать условия в вашей программе, и выполнять различный код, в зависимости от выполнения условия. В Python условный оператор представлен ключевым словом "if". Напишите пример, который проверяет, является ли число четным.

Шаг 8. Сохраните файлы и выложите их в Odin

Модуль 2. Практическое задание № 2. Создание графических пользовательских интерфейсов (GUI)

Шаг 1. Открыть среду разработки Idle

Шаг 2. Создать новый проект

Шаг 3. Создание простого калькулятора: создайте GUI-приложение, которое имеет поле для ввода чисел и кнопки для выполнения математических операций, таких как сложение, вычитание, умножение и деление. После ввода чисел пользователь должен иметь возможность нажать на кнопку, чтобы выполнить выбранную операцию и увидеть результат в приложении.

Шаг 4. Сохраните файл и выложите его в Odin

Модуль 3. Практическое задание № 3. Создание игр на Python с использованием Pygame

Шаг 1. Открыть среду разработки Idle

Шаг 2. Создать новый проект

Шаг 3. «Арканоид»: создайте игру, где пользователь управляет платформой, которая ловит мяч и отбивает его, чтобы разрушить блоки на верхней части экрана. Игра заканчивается, когда все блоки разрушены или мяч падает вниз за пределы платформы.

Шаг 4. Сохраните файл и выложите его в Odin/

Модуль 4. Практическое задание № 4. Разработка плагинов и механизмов движения

Шаг 1. Открыть среду разработки Idle

Шаг 2. Создать новый проект

Шаг 3. «Космический корабль»: создайте игру, где пользователь управляет космическим кораблем, который может летать в космосе и сражаться с врагами. Игрок должен управлять кораблем, чтобы избегать астероидов и атак врагов, стрелять в них и собирать ресурсы.

Шаг 4. Сохраните файл и выложите его в Odin

Для успешного завершения обучения по курсу необходимо, чтобы 50 % заданий в рамках текущей работы были зачтены (оценены преподавателем на 3, 4 или 5 баллов).

Руководитель программы:

Сергеев Олег Владимирович, Российский государственный социальный университет, Москва Факультет: Информационные технологии (бакалавриат) Специальность: Педагогическое образование (информатика) Сентябрь 2011 - Июль 2015 Российский государственный социальный университет, Москва Факультет: Информационные технологии (магистратура) Специальность: Педагогическое образование (информатика) Сентябрь 2016 - Июль 2019

Стаж работы – более 7 лет.

Сертификаты/ курсы:

Разработка веб-приложений с использованием Javascript ФГАОУ ВО "Национальный

исследовательский университет "Высшая школа экономики". Московский институт электроники и математики им. А. Н. Тихонова 2019 год.

Программирование веб-сайтов и веб-приложений на HTML и CSS ФГАОУ ВО "Национальный исследовательский университет "Высшая школа экономики". Московский институт электроники и математики им. А. Н. Тихонова 2018 год.

Методика и использование 3d-прототипирования и моделирования на учебных занятиях в инженерном классе Академия «Просвещение» 2017 год.

«Подготовка технических специалистов, оказывающих информационно-техническую помощь руководителю и организаторам пункта проведения экзамена при проведении государственной итоговой аттестации по образовательным программам основного общего образования» ГАОУ ДПО МЦКО 2017 год.

"ИК технологии при обучении робототехнике" Городской методический центр Департамента образования города Москвы 2016 год Дизайнер-верстальщик РГСУ 2013 год

«Методика обучения технологии программирования объектов дополненной реальности на Unity 3D». Городской Методический Центр 2023 год.

IV. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ МОДУЛЯ (организационно-педагогические)

Материально-технические условия реализации программы: Материально-технические условия реализации программы

Обучение по программе реализовано в режиме синхронно-асинхронной работы слушателей в электронной среде. Для проведения синхронных занятий применяется формат вебинаров. Асинхронная работа слушателей реализуется на базе электронного курса на платформе электронного обучения LMS Odin.

Требования к оборудованию

Сеть: скорость соединения от 2 Мб/с.

Оборудование для синхронных занятий: персональный компьютер (рекомендуется) / мобильный телефон / планшет; наушники, микрофон и камера (обязательно).

Для работы на платформе электронного обучения LMS Odin рекомендуется использовать персональный компьютер.

Для успешной реализации Программы необходимо, чтобы рабочее место обучающегося и преподавателя включали в себя:

- компьютеры, обеспечивающие возможность работы с мультимедийным контентом: воспроизведение видеоизображений, качественный стереозвук в наушниках, речевой ввод с микрофона и др.

периферийное оборудование:

- принтер (черно/белой печати, формата А4);
- устройства для ввода визуальной информации (сканер, цифровой фотоаппарат, web-камера и пр.);
- устройства создания графической информации (графический планшет), используемые для создания и редактирования графических объектов, ввода рукописного текста;

- акустические колонки;
- оборудование, обеспечивающее подключение к сети Интернет (комплект оборудования для подключения к сети Интернет, сервер).

программное обеспечение компьютера:

- операционная система семейства MacOS или Windows;
- программа Python,
- файловый менеджер (в составе операционной системы или др.);
- почтовый клиент (в составе операционных систем или др.);
- браузер (в составе операционных систем или др.);
- мультимедиа проигрыватель (в составе операционной системы или др.);
- антивирусная программа;
- программа-архиватор;
- программа-переводчик;
- программа интерактивного общения;
- текстовый редактор;
- растровый графический редактор;
- звуковой редактор;
- редактор Web-страниц.

Учебно-методическое и информационное обеспечение программы:

Методические рекомендации и пособия по изучению курса.

Программа реализуется в формате смешанного обучения. Синхронные занятия включают интерактивные лекции и практические занятия, предусматривающие групповую и индивидуальную работу слушателей. Для организации асинхронной работы слушателей используются записи синхронных занятий, презентации, конспекты лекций, практические задания и тестовые вопросы, размещаемые в электронном курсе на платформе электронного обучения LMS Odin.

Содержание комплекта учебно-методических материалов.

Для изучения программы используется электронный курс в LMS Odin. Электронный курс включает информационно-организационные материалы по программе (учебно-тематический план, календарный график работы по программе, информацию о преподавателях программы, чат для объявлений и вопросов, сведения о результатах обучения), набор видеолекций, презентации к лекциям, набор ссылок на внешние ресурсы, систему заданий с инструкциями, списки основной и дополнительной литературы.

Список литературы

Литература для педагога

1. "Making Games with Python and Pygame" - автор: Al Sweigart
2. "Python GUI Programming with Tkinter" - автор: Alan D. Moore
3. "Python Programming Blueprints" - автор: Daniel Furtado, Marcus Pennington
4. "Pygame Tutorials" - автор: Paul Vincent Craven

5. "Beginning Game Development with Python and Pygame" - автор: Will McGugan
6. "Python and Pygame for Beginners" - автор: Paul Craven

Компетенции

Наименование компетенции	Тип компетенции	Знания, соответствующие компетенции	Умения, соответствующие компетенции	Владение инструментами, соответствующие компетенции
строка, не менее 10 символов	строка, значение из «общекультурные», «общепрофессиональные», «профессиональные»	строка не менее 50 символов, перечень знаний	строка не менее 50 символов, перечень умений	строка не менее 50 символов, перечень инструментов
ПК-1 Создание программ на языке Python	общепрофессиональные	<ul style="list-style-type: none"> -Принципы работы информационных технологий -Принципы работы алгоритмов -Существующие структуры данных -Принципы написания программ -Работа с командной строчкой -Подключение библиотек 	<ul style="list-style-type: none"> -Устанавливать Python IDLE, Pycharm на ПК -Писать циклы, условия, функции на Python -Подключать сторонние библиотеки 	<ul style="list-style-type: none"> - Средствами разработки на Python Командной строкой (cmd) Методами машинного обучения предварительной обработкой данных подходом к решению задач машинного обучения

Модули

Наименование поля	Допустимые значения полей	Значение полей	Значение полей	Значение полей	Значение полей
Порядковый номер модуля	целое число	1	2	3	4
Название модуля	1.	Модуль 1. Введение в ООП	Модуль 2. Создание графических пользовательских интерфейсов (GUI)	Модуль 3. Создание игр на Python с использованием библиотеки Pygame	Модуль 4. Разработка игр с GUI и Pygame
Описание модуля	2.	Модуль 1. В рамках прохождения данного модуля будут рассмотрены темы: Тема 1.1 Объектно-ориентированное программирование Тема 1.2 Конструктор, деструктор Тема 1.3 Инкапсуляция Тема 1.4 Наследование Тема 1.5 Переопределение Тема 1.6 Lambda функции Тема 1.7 Полиморфизм Тема 1.8 Стандартные библиотеки Python Тема 1.9 Работа с файловой системой. Формат JSON	Модуль 2. В рамках прохождения данного модуля будут рассмотрены темы: Тема 2.1 Введение в графические пользовательские интерфейсы Тема 2.2 Использование библиотеки Tkinter для создания GUI Тема 2.3 Использование библиотеки PyQt для создания GUI Тема 2.4 Создание интерфейсов с использованием	Модуль 3. В рамках прохождения данного модуля будут рассмотрены темы: Тема 3.1 Введение в Pygame Тема 3.2 Работа со звуком и аудио в Pygame Тема 3.3 Работа с графикой в Pygame Тема 3.4 Создание анимации в Pygame Тема 3.5 Работа с клавиатурой и мышью в Pygame Тема 3.6 Создание игровых механик в Pygame	Модуль 4. В рамках прохождения данного модуля будут рассмотрены темы: Тема 4.1 Игровой цикл Тема 4.2 Создание пользовательского интерфейса для игры с помощью Tkinter Тема 4.3 Интеграция пользовательского интерфейса с Pygame Тема 4.4 Интеграция пользовательского интерфейса с Pygame Тема 4.5 Создание современной игры с

			<p>дизайнера пользовательского интерфейса Qt Designer</p> <p>Тема 2.5 Использование графических элементов в PyQt</p> <p>Тема 2.6 Выпадающие меню — Tkinter</p> <p>Тема 2.7 Tkinter — виджет OptionMenu</p> <p>Тема 2.8 Как установить PyQt для Python в Windows?</p> <p>Тема 2.9 Python — калькулятор возраста с использованием PyQt</p> <p>Тема 2.10 Создание браузера с вкладками с использованием PyQt5</p>	<p>Тема 3.7 Создаем змейку в Pygame</p> <p>Тема 3.8 Создаем тетрис в Pygame</p> <p>Тема 3.9 Разработка концепции игры</p>	<p>использованием GUI и Pygame</p> <p>Тема 4.6 Работа со спрайтами для разработки игры в Pygame</p> <p>Тема 4.7 Создание игры «Стрелялка»</p>
Аттестация по итогам модуля. Количество ак. часов	целое число	1	1	1	1
Аттестация по итогам модуля.	строка не менее 4 символов	Зачет	Зачет	Зачет	Зачет

Формы контроля					
Аттестация по итогам модуля. Диагностические инструменты	строка не менее 10 символов	Практическое задание № 1. Введение в язык программирования Python	Практическое задание № 2. Создание графических пользовательских интерфейсов (GUI)	Практическое задание № 3. Создание игр на Python с использованием Pygame	Практическое задание № 4. Разработка игр с GUI и Pygame
Аттестация по итогам модуля. Показатели и критерии оценивания	строка не менее 50 символов	<p>В ходе реализации программы предусмотрены следующие виды оценочных мероприятий:</p> <ul style="list-style-type: none"> - практические задания и упражнения в период синхронной работы; - индивидуальные задания, в том числе со взаимной проверкой (комментированием, рецензированием). 			
Аттестация по итогам модуля. Шкала оценивания, нижнее значение	целое число	2-5	2-5	2-5	2-5
Аттестация по итогам модуля. Шкала оценивания, верхнее значение	целое число	2-5	2-5	2-5	2-5

Аттестация по итогам модуля. Шкала оценивания, минимальный проходной балл для успешной сдачи	целое число в диапазоне шкалы	3	3	3	3
---	-------------------------------	----------	----------	----------	----------

РАБОЧАЯ ПРОГРАММА

«Python: разработка игр с Pygame и GUI»

I. АННОТАЦИЯ

Программа «Python: разработка игр с Pygame и GUI» Программа направлена на формирование профессиональных практических навыков в области цифровых технологий актуальных в 2023-2024 году.

Программа направлена на формирование компетенций в соответствии с трудовыми функциями программиста.

В результате обучения выпускник программы будет способен:

- Создавать свои собственные проекты на Python с помощью GUI и Pygame.
- Программировать на языке Python.
- Строить логические схемы с использованием скриптов.
- Разбираться в коде и исправлять ошибки.
- Работать с анимацией.
- Разбираться в синтаксисе языка программирования Python.
- Создать портфолио из созданных моделей и разработанных игр.
- Продолжить обучение на курсах более сложного уровня.

Рабочая программа с описанием каждого модуля

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Модуль 1. Введение в ООП Введение в язык программирования	Тема 1.1 Объектно- ориентирован ное программиров ание	Знакомство с объектно-ориентированным подходом в программировании.	теоретические занятия	(1 ч.)

Python, это модуль обучения, предназначенный для знакомства с основами языка программирования Python. В ходе этого модуля обучающиеся изучают синтаксис Python, основные типы данных, операторы, условные выражения, циклы и функции. Они также будут знакомиться с базовыми концепциями программирования, включая переменные, массивы, ввод/вывод данных и работу с файлами. Обучающиеся будут практиковаться в	Тема 1.2 Конструктор, деструктор	Введение в магические методы класса: конструктор, деструктор.	теоретические занятия	(1 ч.)
	Тема 1.3 Инкапсуляция	Соккрытие методов класса через инкапсуляцию. Геттер и сеттер	теоретические занятия	(1 ч.)
	Тема 1.4 Наследование	Наследование родительских методов и атрибутов в дочерних классах.	теоретические занятия	(1 ч.)
		Отработка полученных знаний на практике. <i>Задание к теме 1.4</i> Наследование	практические занятия	(5 ч.)
		Изучить дополнительный материал по теме 1.4 Наследование https://metanit.com/python/tutorial/7.3.php	самостоятельная работа	(1 ч.)
	Тема 1.5 Переопределе ние	Переопределение функционала базового класса	теоретические занятия	(1 ч.)
		<i>Отработка полученных знаний на практике</i> <i>Задание к теме 1.5</i> Переопределение	практические занятия	(5 ч.)
		Изучить дополнительный материал по теме 1.5. Переопределение https://metanit.com/python/tutorial/7.4.php	самостоятельная работа	(1 ч.)
Тема 1.6 Lambda функции	Лямбда-выражения в Python определяемые с помощью оператора lambda	теоретические занятия	(1 ч.)	

<p>написании простых программ на Python, решении задач и создании небольших проектов. По окончании модуля обучающиеся получают базовые навыки программирования на Python и будут готовы применять их в дальнейших модулях или проектах, требующих использования этого языка.</p>				
		<p><i>Отработка на практике полученных знаний.</i> <i>Задание к теме 1.6</i> Работа с функциями и модулями в Python</p>	практические занятия	(2 ч.)
		<p>Изучить дополнительный материал по теме 1.6. Lambda функции https://metanit.com/python/tutorial/2.18.php</p>	самостоятельная работа	(1 ч.)
	Тема 1.7 Полиморфизм	<p>Рассмотрение разных типов полиморфизма на примерах и способа реализации полиморфизма в Python</p>	теоретические занятия	(1 ч.)
		<p><i>Задание к теме 1.7</i> Полиморфизм</p>	практические занятия	(3 ч.)
		<p>Изучить дополнительный материал по теме 1.7 Полиморфизм https://habr.com/ru/articles/552922/</p>	самостоятельная работа	(1 ч.)
	Тема 1.8 Стандартные библиотеки Python	<p>Изучение стандартных библиотек: datetime, collections, сору</p>	теоретические занятия	(1 ч.)
		<p><i>Задание к теме 1.8</i> Стандартные библиотеки Python</p>	практические занятия	(3 ч.)
		<p>Изучить дополнительный материал по теме 1.8. Стандартные библиотеки Python https://pythonru.com/primery/kak-ispolzovat-modul-datetime-v-python</p>	самостоятельная работа	(1 ч.)

	Тема 1.9 Работа с файловой системой. Формат JSON	Создание файлов. Запись и чтение файлов JSON. Сериализация и десериализация JSON	теоретические занятия	(1 ч.)
		Отработка полученных знаний на практике по теме 1.9 Работа с файловой системой. Формат JSON	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 1.9. Работа с файловой системой. Формат JSON https://sky.pro/media/modul-json-v-python/	самостоятельная работа	(1 ч.)
Промежуточная аттестация		Практическое задание № 1. Введение в ООП. Создать игру “Палочки”. Шаг 1. Создать игровой цикл, где игрок и компьютер по очереди берут палочки от 1 до 3 из набора палочек. Шаг 2. Реализовать поведение компьютера следующим алгоритмом: Создать переменную = генерируем случайное число. Шаг 3. Проверяем если количество палочек 4 * Переменной присваиваем 3 Если количество палочек 3 * Переменной присваиваем 2 Если количество палочек 2 * Переменной присваиваем 1 Общее количество палочек – переменная.	практические занятия	(1 ч.)

	<p>Шаг 4. Проверяем сколько палочек осталось. Если палочек не осталось, объявить проигравшим того, кто взял последнюю палочку</p> <p>Критерии создания программы: Игрок и Компьютер должны объектами соответствующих классов.</p> <p>Шаг 5. Сохраните файлы и выложите их в Odin</p>			
			Объем в ак.ч.	Объем в %
	ИТОГО:	теоретические занятия	9	
		практические занятия	20	58%
		самостоятельная работа	6	17%
		аттестация	1	
		Всего:	36	

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.	
Модуль 2. Создание графических пользовательских интерфейсов (GUI) Модуль 2: Создание графических пользовательских интерфейсов (GUI), это модуль обучения, направленный на изучение разработки графических пользовательских интерфейсов с использованием языка программирования Python. В ходе этого модуля обучающиеся узнают о	Тема 2.1 Введение в графические пользовательские интерфейсы	Графические пользовательские интерфейсы. Для чего нужен GUI. Какие компоненты есть в GUI. Каким может быть GUI.	теоретические занятия	(1 ч.)	
		<i>Отработка на практике полученных знаний. Задание к теме 2.1</i> Графические пользовательские интерфейсы	практические занятия	(1 ч.)	
	Тема 2.2 Использование библиотеки Tkinter для создания GUI	Использование библиотеки Tkinter для создания GUI	Использование библиотеки Tkinter для создания GUI. Для чего нужна библиотека tkinter в Python	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний. Задание к теме 2.2</i> Использование библиотеки Tkinter для создания GUI	практические занятия	(2 ч.)	
	Тема 2.3 Использование библиотеки PyQt для создания GUI	Использование библиотеки PyQt для создания GUI	Использование библиотеки PyQt для создания GUI. Для чего нужна библиотека PyQt	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний. Задание к теме 2.3</i>	практические занятия	(2 ч.)	

<p>различных библиотеках и фреймворках Python, которые позволяют создавать интерактивные и привлекательные GUI для своих приложений. Они изучат основные элементы интерфейса, такие как кнопки, текстовые поля, выпадающие списки, таблицы и другие виджеты, а также научатся работать с событиями и обработкой пользовательского ввода. Обучающиеся также будут практиковаться в создании пользовательских</p>		Использование библиотеки PyQt для создания GUI		
		Изучить дополнительный материал по теме 2.3 Использование библиотеки PyQt для создания GUI https://www.youtube.com/watch?v=HuszbTeETUo	самостоятельная работа	(1 ч.)
	<p>Тема 2.4 Создание интерфейсов с использованием дизайнера пользовательского интерфейса Qt Designer</p>	Как создавать интерфейсы с использованием дизайнера пользовательского интерфейса Qt Designer	теоретические занятия	(1 ч.)
		<p><i>Отработка на практике полученных знаний.</i> <i>Задание к теме 2.4</i> Создание интерфейсов с использованием дизайнера пользовательского интерфейса Qt Designer</p>	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 2.4 Создание интерфейсов с использованием дизайнера пользовательского интерфейса Qt Designer https://tproger.ru/translations/python-gui-pyqt/	самостоятельная работа	(1 ч.)
Тема 2.5	Применение графических элементов в PyQt	теоретические занятия	(1 ч.)	

интерфейсов для различных приложений, включая формы ввода данных, панели управления и многое другое. По завершении модуля обучающиеся будут обладать навыками разработки графических пользовательских интерфейсов на языке Python и смогут применять их в своих проектах.	Использование графических элементов в PyQt	<i>Отработка на практике полученных знаний.</i> Задание к теме 2.5 Использование графических элементов в PyQt	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 2.5 Использование графических элементов в PyQt https://is42-2018.susu.ru/safonovan/2020/12/11/python-i-pyqt-sozdanie-menju-panelej-instrumentov-i-strok-sostoyaniya/	самостоятельная работа	(1 ч.)
	Тема 2.6 Выпадающие меню — Tkinter	Как сделать выпадающие меню — Tkinter в Python	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> Задание к теме 2.6 Выпадающее меню — Tkinter	практические занятия	(3 ч.)
	Тема 2.7 Tkinter — виджет OptionMenu	Tkinter — виджет OptionMenu в Python	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> Задание к теме 2.7 Tkinter — виджет OptionMenu	практические занятия	(2 ч.)

Тема 2.8 Как установить PyQt для Python в Windows?	Установка PyQt для Python в Windows?	теоретические занятия	(1 ч.)
	<i>Отработка на практике полученных знаний.</i> Задание к теме 2.8 Как установить PyQt для Python в Windows?	практические занятия	(2 ч.)
	Изучить дополнительный материал по теме 2.8 Как установить PyQt для Python в Windows? https://www.youtube.com/watch?v=eTL25yiNpG4	самостоятельная работа	(1 ч.)
Тема 2.9 Python — калькулятор возраста с использованием PyQt	Создание простого калькулятора в Python — калькулятор возраста с использованием PyQt	теоретические занятия	(1 ч.)
	<i>Отработка на практике полученных знаний.</i> Задание к теме 2.9 Python — калькулятор возраста с использованием PyQt	практические занятия	(2 ч.)
	Изучить дополнительный материал по теме 2.9 Python — калькулятор возраста с использованием PyQt https://python-scripts.com/pyqt5-calculator	самостоятельная работа	(1 ч.)

	Тема 2.10 Создание браузера с вкладками с использовани ем PyQt5	Отработка полученных знаний на практике по теме 2.10 Создание браузера с вкладками с использованием PyQt5	практические занятия	(2 ч.)	
		Изучить дополнительный материал по теме 2.10 Система частиц. Осколки https://www.youtube.com/watch?v=GAffSuQZtdU	самостоятельная работа	(1 ч.)	
Промежуточная аттестация		Практическое задание № 2. Roblox Studio Шаг 1. Открыть среду разработки Idle Шаг 2. Создать новый проект Шаг 3. Создание простого калькулятора: создайте GUI-приложение, которое имеет поле для ввода чисел и кнопки для выполнения математических операций, таких как сложение, вычитание, умножение и деление. После ввода чисел пользователь должен иметь возможность нажать на кнопку, чтобы выполнить выбранную операцию и увидеть результат в приложении. Шаг 4. Сохраните файл и выложите его в Odin	практические занятия	(1 ч.)	
				Объем в ак.ч.	Объем в %

ИТОГО:	теоретические занятия	9	
	практические занятия	20	55%
	самостоятельная работа	6	17%
	аттестация	1	
	Всего:	36	

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Модуль 3. Создание игр на Python с использованием библиотеки Pygame Модуль 3: Создание игр на	Тема 3.1 Введение в Pygame	Введение в Pygame. Установка библиотеки. Pygame. Введение в разработку игр на Python" знакомит с базовыми особенностями программирования игр и основными возможностями библиотеки Pygame.	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний. Задание к теме 3.1</i>	практические занятия	(2 ч.)

Python с использованием библиотеки Pygame, это модуль обучения, направленный на изучение создания игр с использованием языка программирования Python и библиотеки Pygame. В ходе этого модуля обучающиеся узнают о основных принципах разработки игр, включая обработку пользовательского ввода, управление графикой и анимацией, обработку столкновений, управление звуком		Введение в Pygame.		
		Изучить дополнительный материал по теме 3.1 Введение в Pygame https://younglinux.info/pygame/	самостоятельная работа	(1 ч.)
	Тема 3.2 Работа со звуком и аудио в Pygame	Как работать со звуком и аудио в Pygame. Класс Sound и модуль pygame.mixer.music	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> <i>Задание к теме 3.2</i> Работа со звуком и аудио в Pygame	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 3.2 Работа со звуком и аудио в Pygame https://younglinux.info/pygame/mixer	самостоятельная работа	(1 ч.)
	Тема 3.3 Работа с графикой в Pygame	Как работать с графикой в Pygame. Основной цикл. Вывод текста. Обработка событий. Графические примитивы и маски. Работа с растровыми файлами. Частота обновления.	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> <i>Задание к теме 3.3</i> Работа с графикой в Pygame	практические занятия	(2 ч.)

и другие аспекты игрового процесса. Они будут изучать возможности, предоставляемые библиотекой Pygame, и практиковаться в создании различных типов игр, таких как аркадные игры, головоломки, платформеры и другие. Обучающиеся будут разрабатывать игровую логику, создавать игровые сцены, добавлять графические элементы, звуки и эффекты, а также оптимизировать производительность		Изучить дополнительный материал по теме 3.3 Работа с графикой в Pygame https://nsu-programming.github.io/textbook/python/graphics	самостоятельная работа	(1 ч.)
	Тема 3.4 Создание анимации в Pygame	Создание анимации с Python и Pygame Фреймворк Pygame. Метода subsurface	теоретические занятия	(1 ч.)
		Отработка полученных знаний на практике по теме 3.4 Создание анимации в Pygame	практические занятия	(1 ч.)
		Изучить дополнительный материал по теме 3.4 Создание анимации в Pygame https://myrusakov.ru/pygame-animation-loop.html	самостоятельная работа	(1 ч.)
	Тема 3.5 Работа с клавиатурой и мышью в Pygame	Как работать с клавиатурой и мышью в Pygame. События мыши. Свойство события – button. Свойство pos. Атрибут – rel	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> <i>Задание к теме 3.5</i> Работа с клавиатурой и мышью в Pygame	практические занятия	(2 ч.)
	Тема 3.6 Создание	Как создавать игровые механики в Pygame	теоретические занятия	(1 ч.)

в своих игр. По завершении модуля обучающиеся будут обладать навыками создания игр на языке Python с использованием библиотеки Pygame и смогут применять их для разработки собственных игровых проектов.	игровых механик в Pygame	<i>Отработка на практике полученных знаний.</i> Задание к теме 3.6 Создание игровых механик в Pygame	практические занятия	(3 ч.)
	Тема 3.7 Создаем змейку в Pygame	Как создать змейку в Pygame. Создание экрана. Создание змейки. Движение змейки.	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> Задание к теме 3.7 Создаем змейку в Pygame	практические занятия	(3 ч.)
	Тема 3.8 Создаем тетрис в Pygame	Как создать тетрис в Pygame. Основные параметры игры. Шаблоны и цвет фигур. Пауза, экран паузы и прозрачность. Основной код Тетриса.	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> Задание к Теме 3.8 Создаем тетрис в Pygame	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 3.8 Создание анимации в Pygame https://www.youtube.com/watch?v=1MNSoLP45yE	самостоятельная работа	(1 ч.)
	Тема 3.9 Разработка	Разработка концепции игры в Pygame. Рисование. Анимация и обработка событий. Покадровая анимация. Управление	теоретические занятия	(2 ч.)

	концепции игры	движением объекта.		
		Отработка на практике полученных знаний. Задание к Теме 3.9 Разработка концепции игры	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 3.9 Разработка концепции игры https://proglib.io/p/kak-napisat-igru-na-python-5-igrovyyh-dvizhkov-2022-11-07	самостоятельная работа	(1 ч.)
Промежуточная аттестация		Практическое задание № 3. Создание анимации в Pygame Шаг 1. Открыть среду разработки Idle Шаг 2. Создать новый проект Шаг 3. «Арканоид»: создайте игру, где пользователь управляет платформой, которая ловит мяч и отбивает его, чтобы разрушить блоки на верхней части экрана. Игра заканчивается, когда все блоки разрушены или мяч падает вниз за пределы платформы. Шаг 4. Сохраните файл и выложите его в Odin/	практические занятия	(1 ч.)
				Объем в ак.ч.
				Объем в %

ИТОГО:	теоретические занятия	10	
	практические занятия	19	53%
	самостоятельная работа	6	17%
	аттестация	1	
	Всего:	36	

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Модуль 4. Разработка игр с GUI и Pygame Модуль 4: Разработка игр с графическими пользовательскими интерфейсами	Тема 4.1 Игровой цикл	Что такое игровой цикл. Как писать игры на Python с Pygame. Выход из игрового цикла. Игровые события. Игровой экран. Создание игры.	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> <i>Задание к теме 4.1</i> Игровой цикл	практические занятия	(3 ч.)

<p>(GUI) и библиотекой Pygame, это модуль обучения, целью которого является изучение разработки игр с использованием графических пользовательских интерфейсов и библиотеки Pygame на языке программирования Python. В ходе этого модуля обучающиеся углубятся в создание игровых проектов, где основное внимание будет уделено разработке интерфейсов, включая графические</p>		Изучить дополнительный материал по теме 4.1 Игровой цикл https://timeweb.cloud/tutorials/python/kak-pisat-igry-na-python-s-pygame	самостоятельная работа	(1 ч.)
	Тема 4.2 Создание пользовательского интерфейса для игры с помощью Tkinter	Как создать пользовательский интерфейс для игры с помощью Tkinter	теоретические занятия	(2 ч.)
		<i>Отработка на практике полученных знаний.</i> Задание к теме 4.2 Создание пользовательского интерфейса для игры с помощью Tkinter	практические занятия	(2 ч.)
	Тема 4.3 Создание движущихся механизмов в Pygame	Как создать движущиеся механизмы в Pygame	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> Задание к теме 4.3 Создание движущихся механизмов в Pygame	практические занятия	(3 ч.)
		Изучить дополнительный материал по теме 4.3 Создание движущихся механизмов Pygame https://www.youtube.com/watch?v=mJCbnWR4jmc	самостоятельная работа	(1 ч.)

элементы, кнопки, окна и другие интерактивные элементы. Они изучат способы интеграции графического пользовательского интерфейса с игровым движком Pygame и научатся управлять игровым процессом с помощью GUI. Обучающиеся будут практиковаться в создании игровых проектов с использованием графического интерфейса, разрабатывать пользовательские меню, системы	Тема 4.4 Интеграция пользовательского интерфейса с Pygame	Интеграция пользовательского интерфейса с Pygame. Как разрабатывать на Python под Android.	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> Задание к теме 4.4 Интеграция пользовательского интерфейса с Pygame	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 4.4 Интеграция пользовательского интерфейса с Pygame https://tproger.ru/translations/python-development-on-android/	самостоятельная работа	(1 ч.)
	Тема 4.5 Создание современной игры с использованием GUI и Pygame	Как создавать современные игры с использованием GUI и Pygame. Основной цикл. Обработка событий. Отрисовка. Воспроизведение звука	теоретические занятия	(1 ч.)
		<i>Отработка на практике полученных знаний.</i> Задание к теме 4.5 Создание современной игры с использованием GUI и Pygame	практические занятия	(3 ч.)

настроек и другие элементы, повышающие удобство и привлекательность игры для пользователей. По завершении модуля обучающиеся будут обладать навыками разработки игр с GUI и использования библиотеки Pygame для создания увлекательных и интерактивных игровых проектов.		Изучить дополнительный материал по теме 4.5 Создание современной игры с использованием GUI и Pygame https://www.youtube.com/watch?v=wDgZdYRQ4gU	самостоятельная работа	(1 ч.)
	Тема 4.6 Работа со спрайтами для разработки игры в Pygame	Спрайты для разработки игры в Pygame. Классы Sprite и Group	теоретические занятия	(2 ч.)
		<i>Отработка на практике полученных знаний.</i> <i>Задание к теме 4.6</i> Работа со спрайтами для разработки игры в Pygame	практические занятия	(3 ч.)
		Изучить дополнительный материал по теме 4.6 Работа со спрайтами для разработки игры в Pygame https://younglinux.info/pygame/sprite	самостоятельная работа	(1 ч.)
	Тема 4.7 Создание игры «Стрелялка»	Как создать игру «Стрелялка»	теоретические занятия	(1 ч.)
		Отработка полученных знаний на практике по теме 4.7 Создание игры «Стрелялка»	практические занятия	(2 ч.)
		Изучить дополнительный материал по теме 4.7 Создание игры «Стрелялка»	самостоятельная работа	(1 ч.)

		https://itvdn.com/ru/channel/video/fps-python-22		
Консультация				(2 ч.)
Промежуточная аттестация		<p>Практическое задание № 4. Разработка игр с GUI и Pygame</p> <p>Шаг 1. Открыть среду разработки Idle</p> <p>Шаг 2. Создать новый проект</p> <p>Шаг 3. «Космический корабль»: создайте игру, где пользователь управляет космическим кораблем, который может летать в космосе и сражаться с врагами. Игрок должен управлять кораблем, чтобы избегать астероидов и атак врагов, стрелять в них и собирать ресурсы.</p> <p>Шаг 4. Сохраните файл и выложите его в Odin</p>	практические занятия	(1 ч.)
				Объем
				м
				в
				ак.ч.
				Объем в %
ИТОГО:			теоретические занятия	9
			практические занятия	20
			самостоятельная работа	6
				55%
				17%

	промежуточная аттестация	1
	Всего:	36

Задание к теме 1.4

Наследование (5 ч)

1. Откройте пустой проект в IDLE
2. Напишите программу на Python. Задание включает в себя следующие критерии:
 - Опишите на ООП взаимодействие студента, ментора, код-ревьюера и куратора.
 - Все эти люди — люди, поэтому создадим базовый класс Human, со свойством name (у каждого человека должно быть имя) и методом answer_question() для ответов на вопросы.
 - По умолчанию объект Human будет отвечать на любой вопрос так: «Очень интересный вопрос! Не знаю.»
 - От класса Human унаследуем классы Student, Mentor, CodeReviewer и Curator. Student должен уметь задавать вопросы.
 - Реализуйте в классе Student метод ask_question (Human, question). При вызове этот метод должен:
 - Напечатать на экране вопрос в формате <имя человека, которому задаём вопрос>, <текст вопроса>
 - Задать вопрос question человеку, объекту класса Human. Имя объекта, которому адресован вопрос, передаётся при вызове метода ask_question().
 - Объекты классов Mentor, CodeReviewer и Curator должны уметь отвечать на вопросы при вызове метода answer_question().
 - Задан непредусмотренный вопрос — для него подойдет ответ по умолчанию.

Задание к теме 1.5

Переопределение (5 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу на Python. Задание включает в себя следующие критерии:
 - создать класс Line с метод write, который в консоль выводит информацию о форме объекта;
 - создать класс Rect наследник класса Line;
 - добавить при инициализации класса параметр color (по умолчанию red);
 - переопределить метод write.

Задание к теме 1.6

Lambda функции (2 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу на Python. Напишите функцию, которая будет сортировать список авторов книг по их фамилиям:
 - В списке могут содержаться как полные имена, так и инициалы.
 - Если указывается не только фамилия, то она идет последней (т.е. Александр Пушкин, а не Пушкин Александр).
 - Имена, фамилии, инициалы могут писаться буквами в разных регистрах.

- Если две фамилии начинаются с одной буквы, верните их в том порядке, в котором они стоят в списке.
- Обратите внимание на пробелы между инициалами.

Задание к теме 1.7

Полиморфизм (3 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу на Python. Перегрузить оператор сложения:
 - Для его перегрузки используется метод `__add__`. Он вызывается, когда объекты класса, имеющего данный метод, фигурируют в операции сложения, причем с левой стороны. Это значит, что в выражении `a + b` у объекта `a` должен быть метод `__add__`.
 - Объект `b` может быть чем угодно, но чаще всего он бывает объектом того же класса. Объект `b` будет автоматически передаваться в метод `__add (self, b)` в качестве второго аргумента.
 - В вашей программе метод перегрузки сложения будет возвращать новый объект того же класса.

Задание к теме 1.8

Стандартные библиотеки Python (3 ч.)

1. Откройте пустой проект в IDLE
2. Задание: Написать программу, которая позволяет пользователям рассчитать стоимость содержания своего автомобиля на основе различных параметров. Программа должна запрашивать у пользователя следующую информацию:

- Марка и модель автомобиля
- Год выпуска автомобиля
- Пробег автомобиля в километрах
- Средний расход топлива на 100 км
- Цена топлива за литр
- Стоимость страховки в год
- Стоимость ремонта в год (включая запасные части и техобслуживание)
- Прочие расходы в год (например, на парковку, штрафы и т.д.)

На основе этой информации программа должна рассчитывать общую стоимость содержания автомобиля за год и выводить ее на экран. Программа должна также позволять пользователю изменять любой из параметров и перерасчитывать стоимость содержания автомобиля. Примечание: для расчета среднего расхода топлива на 100 км необходимо запросить от пользователя расход топлива на 1 км и перевести его в литры на 100 км. Для удобства пользователя можно также предложить ввести информацию о стоимости автомобиля и рассчитать ежегодную депрециацию на основе средней цены рынка для автомобилей такой же марки и модели.

Задание к теме 2.1

Введение в графические пользовательские интерфейсы (1 ч.)

1. Откройте пустой проект в IDLE
2. Цель задания: показать, как создать простое приложение с помощью графического пользовательского интерфейса на Python.

Задачи:

- Создать окно с помощью Tkinter.
- Добавить на окно элементы управления, такие как метки, кнопки и текстовые поля.
- Создать функции для обработки событий, например, щелчков на кнопках или изменения текста в текстовых полях.
- Подключить функции к соответствующим элементам управления.
- Создать логику приложения, например, программу для расчета простых математических операций или список задач с возможностью добавления и удаления задач.
- Обеспечить графический интерфейс для взаимодействия с пользователем, например, вывод результатов расчетов или отображение списка задач.
- Обеспечить возможность сохранения данных, если это применимо для приложения.

Задание к теме 2.2

Использование библиотеки Tkinter для создания GUI (2 ч.)

1. Откройте пустой проект в IDLE
2. Цель задания: научить студентов создавать приложения с помощью библиотеки Tkinter и освоить основные элементы управления.

Задачи:

- Создать окно приложения с помощью класса Tk.
- Добавить на окно метку с описанием приложения.
- Добавить текстовое поле и кнопку, которая будет выводить содержимое поля в консоль.
- Создать функцию, которая будет вызываться при нажатии на кнопку и выводить содержимое текстового поля в консоль.
- Добавить элементы управления, такие как флажки, переключатели и список, чтобы показать, как их использовать в приложении.
- Создать функции для обработки событий, например, изменения состояния флажков или выбора элемента из списка.
- Обеспечить возможность сохранения данных, если это применимо для приложения.
- Провести тестирование приложения, убедиться, что все элементы управления работают правильно.

Задание к теме 2.3

Использование библиотеки PyQt для создания GUI (2 ч.)

1. Откройте пустой проект в IDLE

2. Напишите программу, которая создает графический интерфейс пользователя для простого калькулятора. Калькулятор должен иметь функциональность для выполнения операций сложения, вычитания, умножения и деления двух чисел.

Интерфейс должен содержать следующие элементы:

- Два поля для ввода чисел.
- Четыре кнопки для выполнения операций сложения, вычитания, умножения и деления.
- Поле для вывода результата операции.
- Кнопка для очистки всех полей.
- Программа должна использовать библиотеку PyQt для создания графического интерфейса пользователя. Кроме того, программа должна реализовывать функциональность для выполнения операций и вывода результатов на экран.

Задание к теме 2.4

Создание интерфейсов с использованием дизайнера пользовательского интерфейса Qt Designer (2 ч.)

1. Откройте пустой проект в IDLE
2. Напишем задание на создание интерфейса с использованием дизайнера пользовательского интерфейса Qt Designer:

Напишите программу, которая создает графический интерфейс пользователя для приложения "Блокнот". Интерфейс должен содержать следующие элементы:

- Текстовое поле для ввода текста.
- Меню "Файл" с опцией "Сохранить" и "Выход".
- Меню "Правка" с опциями "Вырезать", "Копировать" и "Вставить".
- Меню "Помощь" с опцией "О программе".

Программа должна использовать дизайнер пользовательского интерфейса Qt Designer для создания графического интерфейса пользователя. Кроме того, программа должна реализовывать функциональность для выполнения операций вырезания, копирования и вставки текста, а также сохранения и выхода из приложения.

Задание к теме 2.5

Использование графических элементов в PyQt (2 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу, которая создает графический интерфейс пользователя с использованием библиотеки PyQt. Интерфейс должен содержать следующие элементы:

- Окно с заголовком "Пример использования графических элементов в PyQt".
- Кнопка "Нажми меня!".
- Текстовое поле, в которое будет выводиться текст после нажатия на кнопку.
- Поле для ввода текста.

При нажатии на кнопку "Нажми меня!" в текстовое поле должно выводиться содержимое поля для ввода текста. Также необходимо использовать другие

графические элементы (например, метки, флажки, радиокнопки и т.д.) для демонстрации возможностей библиотеки PyQt.

Задание к теме 2.6

Выпадающие меню — Tkinter (3 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу, которая создает графический интерфейс пользователя с использованием библиотеки Tkinter. Интерфейс должен содержать следующие элементы:
 - Окно с заголовком "Пример использования выпадающих меню в Tkinter".
 - Выпадающее меню "Файл" с опцией "Открыть" и "Выход".
 - Выпадающее меню "Правка" с опцией "Вырезать", "Копировать" и "Вставить".
 - Текстовое поле для отображения выбранных опций меню.

При выборе опций "Открыть", "Выход", "Вырезать", "Копировать" и "Вставить" в текстовое поле должна выводиться соответствующая информация. Также необходимо использовать другие графические элементы (например, кнопки, метки и т.д.) для демонстрации возможностей библиотеки Tkinter.

Задание к теме 2.7

Tkinter — виджет OptionMenu (2 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу, которая создает графический интерфейс пользователя с использованием библиотеки Tkinter. Интерфейс должен содержать следующие элементы:
 - Окно с заголовком "Пример использования виджета OptionMenu в Tkinter".
 - Надпись "Выберите язык программирования:".
 - Выпадающий список с вариантами выбора языка программирования: "Python", "Java", "C++", "JavaScript".
 - Кнопка "Выбрать".
 - Текстовое поле для отображения выбранного языка программирования.

При выборе опции из выпадающего списка и нажатии на кнопку "Выбрать" в текстовом поле должен отобразиться выбранный язык программирования.

Задание к теме 2.8

Как установить PyQt для Python в Windows (2 ч.)

1. Скачайте установщик PyQt с официального сайта Riverbank Computing: <https://www.riverbankcomputing.com/software/pyqt/download5>
2. Запустите установщик и следуйте инструкциям на экране.
3. Выберите нужные вам компоненты для установки. Если вы планируете использовать Qt Designer для создания графического интерфейса, убедитесь, что вы выбрали опцию "Qt Designer Components".
4. Укажите путь установки PyQt. По умолчанию, он будет установлен в папку "C:\PythonXX\Lib\site-packages\PyQt5", где XX - номер версии Python.

5. Дождитесь завершения установки.
6. После установки вы можете начать использовать PyQt в своих проектах Python. Обратите внимание, что для работы с PyQt вам также может понадобиться установить библиотеку Qt. Вы можете скачать ее с официального сайта Qt: <https://www.qt.io/download>.

Задание к теме 2.9

Python — калькулятор возраста с использованием PyQt (2 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу-калькулятор, который вычисляет возраст на основе введенной даты рождения. Интерфейс программы должен быть создан с использованием PyQt. Программа должна иметь следующие элементы интерфейса:
 - Текстовое поле для ввода даты рождения в формате "дд.мм.гггг".
 - Кнопку "Вычислить", при нажатии на которую будет производиться расчет возраста.
 - Поле для вывода результата в формате "Ваш возраст: X лет, Y месяцев, Z дней".

При выполнении задания необходимо использовать следующие элементы PyQt:

- QLabel - для создания текстовых меток;
- QLineEdit - для создания поля ввода;
- QPushButton - для создания кнопки.

Кроме того, программа должна реализовывать следующую функциональность:

- При нажатии на кнопку "Вычислить" должна происходить проверка корректности введенных данных.
- В случае ошибочного ввода (например, неправильный формат даты) должно выводиться сообщение об ошибке.
- Расчет возраста должен быть произведен в соответствии с текущей датой.

Задание к теме 3.1

Введение в Pygame (2 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу, используя библиотеку Pygame, которая будет отображать окно с фоновым цветом и простой графикой в виде квадрата или круга.

Программа должна иметь следующую функциональность:

- Окно программы должно иметь размер 600x400 пикселей и белый фоновый цвет.
- На экране должна быть нарисована графика в виде квадрата или круга произвольного цвета, размера и расположения на экране.
- Графика должна быть отрисована каждые 20 миллисекунд с использованием функции `pygame.time.Clock()`.

При выполнении задания необходимо использовать следующие элементы Pygame:

- `pygame.init()` - для инициализации Pygame.
- `pygame.display.set_mode()` - для создания окна.

- `pygame.draw.rect()` или `pygame.draw.circle()` - для отрисовки графики.
- `pygame.time.Clock()` - для обновления экрана.

Дополнительно можно реализовать следующие функции:

- Изменение цвета графики при клике мышью на окно программы.
- Изменение размера и/или формы графики при клике мышью на окно программы.
- Движение графики по экрану с использованием клавиатуры.

При выполнении задания следует руководствоваться официальной документацией Pygame: <https://www.pygame.org/docs/>

Задание к теме 3.2

Работа со звуком и аудио в Pygame (1 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу, которая загружает звуковой файл в формате WAV с помощью Pygame и проигрывает его при нажатии на кнопку "Play". Программа должна также иметь возможность управлять громкостью звука с помощью слайдера.

Интерфейс программы должен состоять из следующих элементов:

- Кнопка "Play", при нажатии на которую будет проигрываться звуковой файл.
- Слайдер для управления громкостью звука.
- Метка, указывающая текущее значение громкости.

При выполнении задания необходимо использовать следующие элементы Pygame:

- `pygame.mixer.Sound` - для загрузки звукового файла в память.
- `sound.play()` - для воспроизведения звукового файла.
- `sound.set_volume()` - для установки громкости звука.

Кроме того, программа должна реализовывать следующую функциональность:

- При изменении значения слайдера громкость звука должна меняться соответственно.
- Метка, указывающая текущее значение громкости, должна обновляться при изменении значения слайдера.

Задание к теме 3.3

Работа с графикой в Pygame (2 ч.)

1. Откройте пустой проект в IDLE
2. Цель задания: Изучение базовых возможностей работы со звуком и аудио в Pygame.

Описание задания: Написать игру на Pygame с использованием звуковых эффектов.

Игра должна иметь следующие элементы:

- Игровое поле, на котором появляются объекты (например, мячи).
- Графический интерфейс с кнопками управления (например, "Старт", "Стоп", "Выход").
- Звуковые эффекты при столкновении объектов или при нажатии на кнопки управления.

Кроме того, необходимо выполнить следующие дополнительные требования:

- Реализовать загрузку звуковых эффектов в формате WAV или MP3.
- Добавить возможность настройки громкости звуковых эффектов.
- Использовать объектно-ориентированный подход при написании кода.

Для выполнения задания необходимо использовать следующие элементы Pygame:

- `pygame.mixer` - для работы со звуковыми эффектами.
- `pygame.mixer.Sound` - для загрузки звуковых файлов.
- `pygame.mixer.Channel` - для управления каналами воспроизведения звуковых эффектов.
- `pygame.mixer.set_volume` - для настройки громкости звуковых эффектов.

Задание к теме 3.5

Работа с клавиатурой и мышью в Pygame (2 ч.)

1. Откройте пустой проект в IDLE
2. Напишите программу на Pygame, которая будет реагировать на пользовательский ввод с клавиатуры и мыши.
 - Создайте окно Pygame и настройте его размер и заголовок.
 - Настройте игровой цикл Pygame, чтобы он работал бесконечно.
 - Создайте объекты для отслеживания состояния клавиатуры и мыши с помощью функции `pygame.key` и `pygame.mouse`.
 - Напишите цикл, который будет проверять состояние клавиатуры и мыши на каждой итерации игрового цикла.
 - Если определенная клавиша была нажата, обработайте это событие соответствующим образом. Например, если нажата клавиша "W", двигайте объект вверх.
 - Если произошло событие мыши, такое как щелчок или движение, обработайте его соответствующим образом. Например, если пользователь щелкнул левой кнопкой мыши, запустите функцию стрельбы.
 - Обновите экран на каждой итерации игрового цикла, чтобы показать все изменения, произошедшие в игре.

Задание к теме 3.6

Создание игровых механик в Pygame (3 ч.)

1. Откройте пустой проект в IDLE
2. Задание: Создание игровых механик в Pygame
3. Цель задания: Научиться создавать игровые механики с помощью Pygame, использовать различные функции и модули Pygame для создания интерактивных игровых приложений.

Шаги выполнения задания:

- Изучите документацию Pygame и ознакомьтесь с основными функциями и модулями.

- Напишите простую игру, используя Pygame. Например, игру "Змейка" или "Тетрис".
- Реализуйте следующие игровые механики:
- Управление игровым объектом с помощью клавиатуры или мыши
- Отслеживание столкновений игровых объектов
- Генерация и удаление игровых объектов в процессе игры
- Создание и использование анимаций
- Работа с звуком и музыкой
- Добавьте в игру графический интерфейс пользователя, содержащий информацию о текущем состоянии игры, счете, уровне сложности и т.д.
- Оптимизируйте игру, чтобы она работала быстро и плавно.
- Протестируйте игру и убедитесь, что все механики работают правильно.
- Документируйте вашу работу, включая комментарии к коду, описание игровых механик и использованных функций и модулей Pygame.

Задание к теме 3.7

Создаем змейку в Pygame (3 ч.)

1. Откройте пустой проект в IDLE
2. Задание: Создание змейки в Pygame

Цель: Написать игру "Змейка" с использованием библиотеки Pygame.

Описание задания:

- Создайте окно игры, установите размер окна и заголовок.
- Создайте змейку и еду.
- Определите движение змейки, используя клавиши управления.
- Если змейка съедает еду, добавьте очки и увеличьте ее размер.
- Если змейка сталкивается с самой собой или стеной, игра заканчивается.
- Создайте систему очков и отображайте их на экране.
- Добавьте звуки и музыку.
- Реализуйте возможность перезапуска игры.
- Добавьте возможность сохранения и загрузки результатов.
- Оформите игру и сделайте ее презентабельной.

Подсказки:

- Для создания змейки можно использовать список координат ее элементов.
- Для отслеживания столкновений можно использовать функцию `pygame.Rect.colliderect()`.
- Для воспроизведения звуков и музыки можно использовать библиотеку `pygame.mixer`.
- Для сохранения и загрузки результатов можно использовать модуль `pickle`.

Задание к теме 3.8

Создаем тетрис в Pygame (2ч.)

1. Откройте пустой проект в IDLE

2. Создайте игру "Тетрис" с помощью библиотеки Pygame. Ваша игра должна иметь следующие функциональные возможности:
- Главное меню: создайте главное меню для игры, которое позволит игроку начать игру или выйти из нее.
 - Игровое поле: создайте игровое поле для тетриса. Поле должно быть 10 клеток в ширину и 20 клеток в высоту. Каждая клетка должна иметь размер 20 пикселей.
 - Фигуры: реализуйте все семь фигур тетриса (I, J, L, O, S, T и Z). Фигуры должны падать сверху вниз и поворачиваться вокруг своей оси.
 - Управление: реализуйте управление фигурами. Игрок должен иметь возможность перемещать фигуры вправо и влево, поворачивать их по часовой стрелке и падать быстрее, нажимая клавиши на клавиатуре.
 - Очки: добавьте систему очков, чтобы игрок мог зарабатывать очки за сбор полных рядов.
 - Конец игры: когда игрок наберет 10 000 очков или соберет 20 полных рядов, игра должна закончиться.
 - Звук: добавьте звуковые эффекты для каждого действия игрока (падение фигуры, удаление полного ряда, конец игры и т.д.).

Задание к теме 3.9

Разработка концепции игры (2ч)

1. Задание:

Ваша задача - разработать концепцию игры. Эта игра может быть любого жанра (например, экшен, стратегия, платформер, головоломка и т.д.), но ее концепция должна быть новой и уникальной. В концепции должны быть определены следующие элементы:

- Цель игры и механика: опишите основную цель игры и механику игрового процесса. Как игрок будет достигать цели? Какие будут основные задачи и миссии?
- Мир игры: опишите мир игры. Какой он будет? Какие у него будут особенности и характеристики? Будут ли в нём различные локации, мирные и военные фракции, персонажи и т.д.?
- Главный герой: опишите главного героя игры. Как он будет выглядеть? Какие у него будут особенности и способности? Какие задачи ему предстоит решать?
- Оппоненты: опишите противников и их особенности. Какие у них будут способности и слабости? Какие будут у них задачи и цели?
- Аудитория игры: определите, для какой аудитории предназначена ваша игра. Каковы будут ее возрастные ограничения? Какие аудитории вы хотите привлечь?
- Ключевые особенности: опишите ключевые особенности игры, которые будут отличать ее от других игр на рынке. Что делает вашу игру уникальной и интересной?
- Монетизация: определите, как вы будете зарабатывать на игре. Какой будет бизнес-модель? Будет ли это платная или бесплатная игра? Будут ли в игре платные элементы?

Ожидаемый результат:

Ваша работа должна содержать подробное описание концепции игры, включающее в себя

все перечисленные элементы. Для удобства вы можете использовать таблицы, графики, диаграммы и другие визуальные средства. Также вы можете приводить примеры и аналоги игр, которые могут быть похожи на вашу концепцию.

Задание к теме 4.1

Игровой цикл (3 ч.)

1. Задание:

Напишите игровой цикл, который будет имитировать основную логику игры.

Требования:

- Игровой цикл должен содержать бесконечный цикл, который будет выполняться до тех пор, пока игра не будет завершена.
- Игровой цикл должен обрабатывать ввод игрока и выполнять необходимые действия в соответствии с этим вводом.
- Игровой цикл должен обновлять состояние игры после каждого действия игрока.
- Игровой цикл должен реализовывать логику искусственного интеллекта, если таковая присутствует в игре.
- Игровой цикл должен проверять условия победы или поражения и завершать игру соответствующим образом.

Задание к теме 4.2

Создание пользовательского интерфейса для игры с помощью Tkinter (2 ч.)

1. Откройте пустой проект в IDLE

2. Создайте игру "Крестики-нолики" с использованием модуля Tkinter для создания пользовательского интерфейса. Игра должна содержать следующие функции:

- Окно с инструкцией и возможностью начать игру.
- Игровое поле размером 3x3, разделенное на клетки.
- Иконки "крестика" и "нолика" для двух игроков.
- Счетчик ходов для определения очередности игроков.
- Вывод сообщений о победе, поражении или ничьей.
- Возможность начать новую игру.

Задание к теме 4.3

Создание движущихся механизмов в Pygame (3 ч.)

1. Откройте пустой проект в IDLE

2. Задание: Создание движущихся механизмов в Pygame

Цель задания: Научиться создавать движущиеся механизмы с использованием библиотеки Pygame.

Шаги выполнения задания:

- Установите библиотеку Pygame, если ее нет на вашем компьютере.
- Создайте новый файл в вашей среде разработки и импортируйте библиотеку Pygame.
- Создайте окно приложения, задайте его размер и заголовок.

- Создайте класс для механизма, который вы хотите создать. Например, это может быть механический робот или простой автомобиль.
- Определите методы класса для перемещения механизма и отрисовки его на экране. Например, вы можете определить методы `move()` и `draw()`.
- Создайте экземпляр класса и вызовите его методы для отрисовки механизма на экране.
- Добавьте обработку событий, чтобы механизм мог перемещаться по экрану в ответ на действия пользователя. Например, вы можете обрабатывать события клавиатуры или мыши.
- Запустите приложение и проверьте, что механизм отображается на экране и перемещается в соответствии с действиями пользователя.
- Добавьте дополнительные функции и опции для вашего механизма, такие как звуковые эффекты или анимации

Задание к теме 4.4

Интеграция пользовательского интерфейса с Pygame (2 ч.)

1. Откройте пустой проект в IDLE
2. Задание: Интеграция пользовательского интерфейса с Pygame Цель: Создать игровой пользовательский интерфейс (UI) на базе библиотеки Pygame.

Задачи:

- Создать окно игры, используя Pygame.
- Отображать изображения элементов пользовательского интерфейса, таких как кнопки, текстовые поля, фоны, и т.д.
- Обработка событий, таких как нажатия клавиш, клики мыши и т.д.
- Реализация логики игры, связанной с пользовательским интерфейсом, такой как изменение состояний элементов интерфейса, обработка ввода пользователя и т.д.
- Реализация анимации и эффектов взаимодействия пользователя с элементами интерфейса.
- Тестирование и отладка пользовательского интерфейса и его взаимодействия с игрой.

Дополнительные требования:

- Игровой пользовательский интерфейс должен быть интуитивно понятным и удобным для использования.
- Игровой пользовательский интерфейс должен быть стильным и эстетически привлекательным.
- Игровой пользовательский интерфейс должен корректно отображаться на различных разрешениях экранов и устройствах.

Ресурсы:

- Официальная документация Pygame (<https://www.pygame.org/docs/>)
- Уроки и примеры на сайте Pygame (<https://www.pygame.org/wiki/tutorials>)
- Форум Pygame (<https://www.pygame.org/tags/forum>)
- Игровые ресурсы, такие как изображения и звуки.

Задание к теме 4.5

Создание современной игры с использованием GUI и Pygame (3 ч.)

1. Откройте пустой проект в IDLE
2. Задание: Создание современной игры с использованием GUI и Pygame.

Описание проекта: В этом проекте вы создадите современную игру, используя Pygame и GUI (графический пользовательский интерфейс). Игра должна быть интересной и иметь высокое качество графики и звука.

Шаги проекта:

- Изучите Pygame и GUI, чтобы понимать, как они работают.
- Определите жанр игры и ее сюжет. Можете выбрать любой жанр, но игра должна быть современной и интересной.
- Создайте прототип игры, который будет содержать минимальный функционал. Например, создайте главное меню, игровое поле и несколько объектов на нем. Проверьте, что все работает корректно.
- Добавьте в игру необходимые элементы, такие как графика, звуковые эффекты, музыку и управление игроком.
- Оптимизируйте игру, чтобы она работала быстро и плавно на любом компьютере.
- Протестируйте игру, чтобы убедиться, что все работает корректно.
- Опубликуйте игру и поделитесь ею со своими друзьями и семьей.

Требования к проекту:

- Игра должна быть современной и интересной;
- Игра должна иметь качественную графику и звуковое сопровождение;
- Игра должна содержать главное меню, игровое поле и элементы управления;
- Игра должна быть оптимизирована, чтобы работать быстро и плавно на любом компьютере.

Дополнительные задачи:

- Добавление разных уровней сложности;
- Добавление мультиплеера;
- Добавление возможности сохранения прогресса игры;
- Добавление локализации игры.

Задание к теме 4.6

Работа со спрайтами для разработки игры в Pygame (3 ч.)

1. Откройте пустой проект в IDLE
2. Задание: Работа со спрайтами для разработки игры в Pygame

Цель задания: Научиться работать со спрайтами в Pygame для создания игровых объектов и их анимации.

Шаги выполнения:

- Создайте новый проект в Pygame.
- Создайте изображения для спрайтов вашей игры в графическом редакторе. Изображения должны быть в формате .png или .jpg.

- Создайте класс для вашего спрайта, который будет унаследован от класса `pygame.sprite.Sprite`.
- В классе спрайта определите конструктор (`init`) и методы `update()` и `draw()`. Конструктор должен инициализировать положение спрайта и его изображение. Метод `update()` будет обновлять положение спрайта в каждом кадре, а метод `draw()` будет рисовать спрайт на экране.
- Создайте группу спрайтов, используя класс `pygame.sprite.Group()`.
- Добавьте созданные спрайты в группу, используя метод `add()`.
- В основном цикле игры вызывайте методы `update()` и `draw()` для каждого спрайта в группе.
- Добавьте анимацию для ваших спрайтов, используя различные изображения для каждого кадра анимации.
- Добавьте обработку столкновений между спрайтами, используя метод `collide_rect()` для определения пересечения спрайтов.
- Добавьте возможность управления спрайтами с клавиатуры или мыши, используя методы `pygame.key.get_pressed()` и `pygame.mouse.get_pos()`.
- Ожидаемый результат: В результате выполнения задания вы должны получить игру с использованием спрайтов, которые двигаются по экрану, имеют анимацию и могут взаимодействовать друг с другом и с игроком.

Календарный план-график реализации образовательной программы

Первый модуль с 01.10.2023 - 30.11.2023

Второй модуль: с 01.12.2023 - 31.01.2024

Третий модуль: с 01.02.2024 - 31.03.2024

Четвертый модуль: с 01.04.2024 - 31.05.2024

Занятия проводятся 2-3 раза в неделю по 2 академических часа, длительность одного академического часа – 45 минут.

1 занятие 2 академических часа

Календарно-тематическое планирование

<i>№</i>	<i>Тема и № модуля</i>	<i>Тема занятия</i>	<i>Кол-во занятий*</i>	<i>Кол-во часов</i>	<i>Дата</i>
1	Модуль 1. Введение в язык программирования Python 01.10.2023 - 30.11.2023	Тема 1.1 Объектно-ориентированное программирование	1	1	2.10.23
2		Тема 1.2 Конструктор, деструктор	1	1	2.10.23
3		Тема 1.3 Инкапсуляция	1	1	5.10.23
4		Тема 1.4 Наследование	3	6	5.10.23 9.10.23 14.10.23

5		Тема 1.5 Переопределение	3	6	16.10.23 19.10.23 21.10.23
6		Тема 1.6 Lambda функции	2	3	25.10.23 27.10.23
7		Тема 1.7 Полиморфизм	2	4	27.10.23 9.11.23 13.11.23
8		Тема 1.8 Стандартные библиотеки Python	2	4	16.11.23 20.11.23
9		Тема 1.9 Работа с файловой системой. Формат JSON	2	4	20.11.23 23.11.23
10		Аттестация	1	1	25.11.23
11	Модуль 2. Создание графических пользовательских	Тема 2.1 Введение в графические пользовательские интерфейсы	1	2	01.12.2023
12		Тема 2.2 Использование библиотеки Tkinter для создания GUI	2	3	04.12.2023 06.12.2023

13	интерфейсов (GUI) <i>01.12 2023 - 31.01.2024</i>	Тема 2.3 Использование библиотеки PyQt для создания GUI	2	3	06.12.2023 8.12.2023
14		Тема 2.4 Создание интерфейсов с использованием дизайнера пользовательского интерфейса Qt Designer	2	3	14.12-2023 18.12.2023
15		Тема 2.5 Использование графических элементов в PyQt	2	3	18.12.2023 21.12.2023
16		Тема 2.6 Выпадающие меню — Tkinter	2	4	25.12.2023 27.12.2023
17		Тема 2.7 Tkinter — виджет OptionMenu	2	3	11.01.2024 15.01.12.2024
18		Тема 2.8 Как установить PyQt для Python в Windows?	2	3	15.01.2024 17.01.2024
19		Тема 2.9 Python — калькулятор возраста с использованием PyQt	2	3	22.01.2024 24.01.2024
20		Тема 2.10 Создание браузера с вкладками с использованием PyQt5	2	3	24.01.2024 29.01.2024

21		Аттестация	1	1	30.01.24
22	Модуль 3. Создание игр на Python с использованием библиотеки Pygame <i>01.02.2024 - 31.03.2024</i>	Тема 3.1 Введение в Pygame	2	3	01.02.2024 03.02.2024
23		Тема 3.2 Работа со звуком и аудио в Pygame	2	3	03.02.2024 07.02.2024
24		Тема 3.3 Работа с графикой в Pygame	2	3	15.02.2024 17.02.2024
25		Тема 3.4 Создание анимации в Pygame	2	3	17.02.2024 21.02.2024
26		Тема 3.5 Работа с клавиатурой и мышью в Pygame	2	3	23.02.2024 26.02.2024
27		Тема 3.6 Создание игровых механик в Pygame	2	4	28.02.2024 01.03.2024
28		Тема 3.7 Создаем змейку в Pygame	2	4	04.03.2024 06.03.2024

29		Тема 3.8 Создаем тетрис в Pygame	2	3	11.03.2024 14.2024
30		Тема 3.9 Разработка концепции игры	2	4	18.03.2024 21.03.2024
31		Аттестация	1	1	23.03.24
32	Модуль 4. Разработка игр с GUI и Pygame <i>01.02.2024 - 31.03.2024</i>	Тема 4.1 Игровой цикл	2	4	03.04.2024 06.04.2024
33		Тема 4.2 Создание пользовательского интерфейса для игры с помощью Tkinter	2	4	08.04.2024 10.04.2024
34		Тема 4.3 Интеграция пользовательского интерфейса с Pygame	2	4	12.04.2024 17.04.2024
35		Тема 4.4 Интеграция пользовательского интерфейса с Pygame	2	3	19.04.2024 22.04.2024
36		Тема 4.5 Создание современной игры с использованием GUI и Pygame	2	4	22.04.2024 26.04.2024 29.04.2024

37	Тема 4.6 Работа со спрайтами для разработки игры в Pygame	3	5	29.04.2024 06.05.2024 13.05.2024
38	Тема 4.7 Создание игры «Стрелялка»	2	4	15.05.2024 20.05.2024
39	Консультация	1	2	21.05.2024
40	Аттестация	1	1	23.05.24

**количество занятий не включает часы, отведенные на самостоятельное изучение, и часы, отведенные на прохождение аттестации*

Учебно-методические материалы

<i>Наименование поля</i>	<i>Допустимые значения полей</i>	<i>Значение полей</i>	<i>Значение полей</i>	<i>Значение полей</i>	<i>Значение полей</i>
<i>Порядковый номер модуля</i>	<i>строка не менее 10 символов</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>

Методы, формы и технологии	строка не менее 10 символов	Программа реализуется в формате смешанного обучения. Синхронные занятия включают интерактивные лекции и практические занятия, предусматривающие групповую и индивидуальную работу слушателей. Для организации асинхронной работы слушателей используются записи синхронных занятий, презентации, конспекты лекций, практические задания и тестовые вопросы, размещаемые в электронном курсе на платформе электронного обучения LMS Odin.
Методические разработки	строка не менее 10 символов	Для изучения программы используется электронный курс в LMS Odin.
Материалы модуля	строка не менее 10 символов	Для изучения программы используется электронный курс в LMS Odin. Электронный курс включает информационно-организационные материалы по программе (учебно-тематический план, календарный график работы по программе, информацию о преподавателях программы, чат для объявлений и вопросов, сведения о результатах обучения), набор видеолекций, презентации к лекциям, набор ссылок на внешние ресурсы, систему заданий с инструкциями, списки основной и дополнительной литературы.
Учебная литература	строка не менее 10 символов	Литература: <ol style="list-style-type: none"> 1. "Making Games with Python and Pygame" - автор: Al Sweigart 2. "Python GUI Programming with Tkinter" - автор: Alan D. Moore 3. "Python Programming Blueprints" - автор: Daniel Furtado, Marcus Pennington

- | | | |
|--|--|---|
| | | <ol style="list-style-type: none">4. "Pygame Tutorials" - автор: Paul Vincent Craven5. "Beginning Game Development with Python and Pygame" - автор: Will McGugan6. "Python and Pygame for Beginners" - автор: Paul Craven |
|--|--|---|

Материально-технические условия реализации программы

<i>Наименование поля</i>	<i>Допустимые значения полей</i>	<i>Значение полей</i>	<i>Значение полей</i>	<i>Значение полей</i>	<i>Значение полей</i>
<i>Порядковый номер модуля</i>	<i>строка не менее 10 символов</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>
Наименование требуемого оборудования	<i>строка не менее 2 символов</i>	<p>Оборудование для синхронных занятий: персональный компьютер (рекомендуется) / мобильный телефон / планшет; наушники, микрофон и камера (обязательно).</p> <p>Для работы на платформе электронного обучения LMS Odin рекомендуется использовать персональный компьютер.</p> <p>Для успешной реализации Программы необходимо, чтобы рабочее место обучающегося и преподавателя включали в себя:</p> <ul style="list-style-type: none"> • компьютеры, обеспечивающие возможность работы с мультимедийным контентом: воспроизведение видеоизображений, качественный стереозвук в наушниках, речевой ввод с микрофона и др. <p>периферийное оборудование:</p> <ul style="list-style-type: none"> • принтер (черно/белой печати, формата А4); • устройства для ввода визуальной информации (сканер, цифровой фотоаппарат, web-камера и пр.); • устройства создания графической информации (графический планшет), использующиеся для создания и редактирования графических объектов, ввода рукописного текста; 			

		<ul style="list-style-type: none"> ● акустические колонки; ● оборудование, обеспечивающее подключение к сети Интернет (комплект оборудования для подключения к сети Интернет, сервер). Сеть: скорость соединения от 2 Мб/с.
<i>Наименование требуемого программного обеспечения</i>	<i>строка не менее 2 символов</i>	<p>Программное обеспечение компьютера:</p> <ul style="list-style-type: none"> ● операционная система семейства MacOS или Windows; ● программа Python, ● файловый менеджер (в составе операционной системы или др.); ● почтовый клиент (в составе операционных систем или др.); ● браузер (в составе операционных систем или др.); ● мультимедиа проигрыватель (в составе операционной системы или др.); ● антивирусная программа; ● программа-архиватор; ● программа-переводчик; ● программа интерактивного общения; ● текстовый редактор; ● растровый графический редактор; ● звуковой редактор; ● редактор Web-страниц.
<i>Электронные информационные ресурсы</i>	<i>строка не менее 10 символов</i>	<p>1. Информационный ресурс по основам БПЛА в свободном доступе от компании ГК Геоскан: https://docs.geoscan.aero/ru/master/database/base-module/base-main.html</p>

		<p>2. Информационный ресурс по основам программирования на Python: https://habr.com/ru/articles/31180/</p> <p>3. Информационный ресурс по бортовой аппаратуре БПЛА в свободном доступе от компании ГК Геоскан: https://docs.geoscan.aero/ru/master/module/module_main.html</p> <p>4. Информационный ресурс по программированию БПЛА в свободном доступе от компании ГК Геоскан: https://docs.geoscan.aero/ru/master/programming/programming.html</p>
<p><i>Электронные образовательные ресурсы</i></p>	<p><i>строка не менее 10 символов</i></p>	<p>1. Электронное свободно распространяемое пособие по основам БПЛА: https://infourok.ru/bespilotnye-letatelnye-apparaty-bpla-kniga-1-teoriya-4559907.html?ysclid=lj4k4sl1mt287611123</p> <p>2. Электронное свободно распространяемое пособие по основам Python: https://elar.urfu.ru/bitstream/10995/28769/1/978-5-7996-1198-9_2014.pdf?ysclid=lj4k928vd8295170590</p> <p>3. Электронное свободно распространяемое пособие по основам БПЛА: https://infourok.ru/bespilotnye-letatelnye-apparaty-bpla-kniga-1-teoriya-4559907.html?ysclid=lj4k4sl1mt287611123</p> <p>4. Электронное свободно распространяемое пособие по основам Python: https://elar.urfu.ru/bitstream/10995/28769/1/978-5-7996-1198-9_2014.pdf?ysclid=lj4k928vd8295170590</p>

Адреса и координаты

№ п/п	Название адрес	Адрес	Код адреса	Долгота	Широта
целое число	строка	строка	Целое число	вещественное число	вещественное число
1	Филиал ФГБОУ ВО Российский государственный социальный университет в г. Пятигорске Колледж, ВУЗ	ул. Карла Маркса, 22, г.Пятигорск, Ставропольский край	07000000000	44.041746	43.078647