

муниципальное общеобразовательное учреждение
МОУ «Гимназия г. Переславля-Залесского»

Рассмотрена на заседании
педагогического совета
протокол №
от «01» ноября 2023 г.

Утверждаю
Директор МОУ «Гимназия»
Л.М. Кольцова Л.М. Кольцова
от «02» ноября 2023 г.



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

естественнонаучной направленности
«Программа «Мастер Python с нуля до первых игр
и приложений начальный (2)»

Возраст обучающихся: 14-17 лет
Срок реализации - 1 год

Составитель: педагог
дополнительного образования
Тарасова В.В.

Городской округ город Переславль-Залесский
г. Переславль – Залесский, 2023 г.

**Общеобразовательная программа дополнительного образования
детей "Мастер Python с нуля до первых игр и приложений"
предназначена для занятий по дополнительному образованию.**

Данная программа составлена на основе программы дополнительного образования Международной школы программирования и математики «Алгоритмика»

Реализация программы обеспечивается нормативными документами различного уровня:

- Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации».
- Приказ Министерства просвещения Российской Федерации от 27.07.2022 № 629 "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам".

Направленность программы – техническая

Рабочая программа курса даёт представления о цели, задачах, общей стратегии обучения, воспитания и развития обучающихся средствами данного курса, устанавливает содержание курса, предусматривает его структурирование по разделам и темам; предлагает распределение учебных часов и последовательность их изучения с учётом межпредметных и внутрипредметных связей, логики учебного процесса, возрастных особенностей обучающихся, включает описание форм организации занятий и учебно-методического обеспечения образовательного процесса.

Программа курса определяет количественные и качественные характеристики учебного материала, в том числе планируемые результаты освоения обучающимися программы курса. Программа служит основой для составления поурочного тематического планирования курса.

Актуальность программы

Python - один из самых универсальных языков программирования. Python входит в топ-10 самых востребованных языков программирования. Самые прогрессивные IT-компании, такие как, Google, Pixar, Youtube, Instagram, NASA, Intel, Pinterest используют именно данный язык программирования.

Python — высокоуровневый язык общего назначения. Это значит, что такой язык быстр и удобен в работе, а программы, написанные на нём, просты для понимания программистами. Обычно его используют для веб-разработки, анализа данных, написания скриптов и игр. Сильные стороны языка, такие как, простота в освоении, низкий порог входа, читабельность, универсальность, большое и активное сообщество, позволяют начать изучать его с достаточно раннего возраста.

Такая популярность вызвана универсальностью и простотой изучения языка — он становится главным ориентиром новичков. Чтобы работать с ним, необязательно быть продвинутым программистом.

На рынке всё ещё чувствуется недостаток специалистов по разработке программного обеспечения. Поэтому количество открытых вакансий на должность разработчика — показатель востребованности специалистов. В 2022 году Java, Python, JavaScript и C++ входили в число популярных языков, требуемых работодателями.

Особая роль курса заключается в формировании алгоритмического мышления ученика. Задача школы популяризировать и актуализировать среди обучающихся необходимость знания программирования в жизни, а также формирование IT-культуры обучающихся, повышение их компьютерной грамотности.

Решение задач программирования способствует раскрытию и расширению прикладного аспекта изучаемого материала.

Обучающиеся, сдающие экзамен по информатике и ИКТ, планируют связать дальнейшую деятельность с IT-сферой, с программированием, разработкой различных продуктов. Поэтому знание такого универсального языка, как Python, необходимо для будущего выпускника школы и абитуриента университета.

Большинство задач единого государственного экзамена решается с использованием языка Python, поэтому изучение данного языка, значительно облегчит подготовку обучающихся к экзамену, и будет способствовать успешному результату сдачи экзамена.

Цели и задачи программы:

Цель:

Способствовать формированию и развитию у обучающихся навыков решения задач программирования, анализа и систематизации полученных ранее знаний, развитие алгоритмического мышления и культуры обучающихся, повышение интереса к изучению программирования, информатики и информационно-коммуникационных технологий, способности к личному самоопределению и самореализации.

Задачи:

Образовательные:

- 1) расширение и углубление представления обучающихся о приемах и методах решения алгоритмических задач;
- 2) формирование и развитие аналитического, логического мышления при проектировании решения задач у обучающихся;
- 3) развитие самостоятельно анализировать и решать задачи по образцу и в незнакомой ситуации;
- 4) развитие самостоятельно анализировать и решать алгоритмические задачи;
- 5) развитие математических способностей, повышение уровня математической грамотности;
- 6) развитие математической интуиции, нахождение оптимального способа решения задач;
- 7) формирование навыков соблюдения единых требований к правилам решения задач, включенных в основной государственный экзамен и единый государственный экзамен.
- 8) формирование навыков решения олимпиадных задач программирования.
- 9) формирование навыка работы с дополнительной литературой, использования различных Интернет-ресурсов.

Развивающие:

- 1) начало профессионального самоопределения, ознакомление с миром профессий, связанных с информатикой;
- 2) умение грамотно, ясно и точно излагать свои мысли в устной и письменной форме, понимать смысл поставленной задачи, выстраивать аргументацию, приводить;
- 3) умение грамотно структурировать данные, оформлять решение алгоритмической задачи;
- 4) развитие критического мышления, умение распознавать логически некорректные высказывания;
- 5) креативное мышление, инициатива, находчивость, активность при решении математических задач;
- 6) умение контролировать процесс и результат учебной математической и алгоритмической деятельности;
- 7) способность к восприятию математических и алгоритмических объектов, задач, решений, рассуждений.

Воспитательные

- представление о социальных нормах и правилах межличностных отношений в коллективе, в том числе в социальных сообществах;
- соблюдение правил безопасности, в том числе навыков безопасного поведения в интернет-среде;
- ориентация на совместную деятельность при выполнении учебных и познавательных задач, создании учебных проектов;
- стремление оценивать своё поведение и поступки своих товарищей с позиции нравственных и правовых норм с учётом осознания последствий поступков;
- создание условий для воспитания личности обладающей способностью и склонностью к творческой деятельности способной к самоопределению, самовоспитанию, самосовершенствованию умение работать в группе для нахождения общего согласованного решения.

Компетентности, приобретаемые ребёнком:

Социально-личностные:

- 1) начало профессионального самоопределения, ознакомление с миром профессий, связанных с информатикой;
- 2) умение грамотно, ясно и точно излагать свои мысли в устной и письменной форме, понимать смысл поставленной задачи, выстраивать аргументацию, приводить;
- 3) умение грамотно структурировать данные, оформлять решение алгоритмической задачи;
- 4) развитие критического мышления, умение распознавать логически некорректные высказывания;
- 5) креативное мышление, инициатива, находчивость, активность при решении математических задач;
- 6) умение контролировать процесс и результат учебной математической и алгоритмической деятельности;
- 7) способность к восприятию математических и алгоритмических объектов, задач, решений, рассуждений.

Проектные:

- 1) осмысленно читать и воспринимать на слух текст задания;
- 2) уметь извлекать и анализировать информацию, полученную из текста;
- 3) уметь критически оценивать данную информацию;
- 4) уметь читать таблицы, блок-схемы, условные обозначения, программу, написанную на языке программирования, а также на алгоритмическом языке.
- 5) иметь представления об идеях и о методах математики как об универсальном языке науки и техники, о средстве моделирования явлений и процессов;
- 6) иметь представления об идеях и о методах программирования, о средствах моделирования на нем;
- 7) уметь планировать свою деятельность, ставить задачи, находить пути их решения;
- 8) уметь видеть математическую задачу в контексте проблемной ситуации в задачах программирования;
- 9) уметь находить в различных источниках информацию, необходимую для решения проблем, и представлять ее в понятной форме; принимать решение в условиях неполной и избыточной, точной и вероятностной информации;
- 10) уметь выдвигать гипотезы при решении учебных задач и понимать необходимость их проверки; умение применять индуктивные и дедуктивные способы рассуждений, видеть различные стратегии решения задач;
- 11) уметь действовать в соответствии с предложенным алгоритмом;
- 12) уметь самостоятельно ставить цели, выбирать и создавать алгоритмы для решения учебных проблем;
- 13) уметь планировать и осуществлять деятельность, направленную на решение задач исследовательского характера.

Коммуникативные:

- выслушивать и принимать во внимание взгляды и мнения других людей;
- выступать на публике - владение способами презентации себя и своей деятельности.

Информационные: способствовать развитию информационной компетентности учащихся через овладение системой дополнительных знаний в области современных ИКТ; формирование у них алгоритмического стиля мышления; развитие познавательной исследовательской деятельности, что будет способствовать подготовке учащихся к жизни в информационном обществе.

Обучающийся должен знать:

- этапы решения задачи на компьютере;
- типы данных;
- базовые конструкции изучаемых языков программирования;
- принципы структурного и модульного программирования;
- принципы объектно-ориентированного программирования;
- основы создания приложений в Python;
- инструментальные средства Python.

Обучающийся должен уметь:

- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программы на конкретном языке

- программирования;
- настраивать рабочую среду Python.

Основными критериями оценки эффективности реализации дополнительной образовательной программы являются:

- мотивационно-ценностный критерий;
- информационный критерий;
- инструментальный критерий;
- деятельностный критерий.

Сроки реализации программы (продолжительность образовательного, процесса, этапы)

Дети занимаются 2 раз в неделю по 1 часу. Количество учебных часов в год – 68 ч. Занятия групповые. Возраст детей – 14-17 лет. Наряду с теоретическими знаниями большое внимание уделяется практическим, проектным работам.

Обучение осуществляется при поддержке Центра образования информационно-технологической направленности «Точка роста», который создан в муниципальном общеобразовательном учреждении «Гимназия г. Переславля-Залесского» для развития у обучающихся технологической грамотности, формирования критического и креативного мышления, совершенствования навыков информационной деятельности, а также для практической отработки учебного материала.

Календарный учебный график

Дата начала занятий	Дата окончания занятий	Кол-во учебных недель	Кол-во часов в год	Место проведения	Режим занятий
01.10	04.06	34	68	МОУ «Гимназия»	2 раз в неделю

Учебно-тематический план программы

Тема	Всего часов	В том числе		
		теоретич. занятия	практич. занятия	Контроль
Модуль 1. Основы языка, управляющие конструкции	17	2	13	2
Модуль 2. Функции, модуль Turtle	17	2	13	2
Модуль 3. Объектно-ориентированное программирование	17	2	13	2
Модуль 4. Разработка игр на Pygame	17	2	13	2
ИТОГО:	68	8	52	8

Содержание программы

Модуль 1. Основы языка, управляющие конструкции

Изучение понятий: «язык программирования», «алгоритм», «программа». Знакомство с синтаксисом языка. Функция ввода `input()`. Функция вывода `print()`. Необходимость вывода данных.

1. Введение в язык Python

- Изучение основ языка Python: базовые функции, типы данных, правила оформления кода, условные операторы, циклы
- Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программы, осуществляющей ввод-вывод данных, ввод команд с клавиатуры.
- Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.

2. Переменные

- Изучение понятий: данные, типы данных, переменная, числовой тип данных, оператор присваивания, приоритет операций.
- Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку переменных числового типа. Изменение типов данных с помощью функций `int()` и `str()`

3. Строки

- Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку строковых переменных (сумма строк, повтор строки `n`-раз, вычисление длины строки, получение символа строки по номеру, замена подстроки на другую подстроку, разбиение строки по разделителю).

4. Вложенные конструкции

- Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих вложенные конструкции.
- Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.

5. Условный оператор

- Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих условный оператор

6. Прохождение аттестации по модулю

Модуль 2. Функции, модуль Turtle

Изучение правил создания собственных функций и модулей на языке Python. Изучение модуля Turtle для создания графических объектов.

1. Функции

- Изучение понятий: «функция», «локальная и глобальная функции», «оператор `def`», «параметры функции», «значение функции». Изучение синтаксиса создания и вызова функции. Использование одной функции внутри другой.
- Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью функций и содержащих функцию внутри другой функции.
- Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.

2. Модули

- Изучение понятий: «модуль», «встроенные модули стандартной библиотеки». Использование модулей `random` и `time`. Синтаксис создания и подключения модуля.
- Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка

программ, осуществляющих обработку данных с помощью модулей, встроенных в стандартную библиотеку, и модулей, не входящих в стандартную библиотеку.

- Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.
3. Хакатон
 - Выполнение заданий на платформе. Создание консольного приложения на заданную тему, используя инструменты, изученные за модуль.
 4. Turtle. Линейные алгоритмы
 5. Turtle. Циклы
 6. Turtle. Вложенные структуры данных
 7. Прохождение аттестации по модулю.

Модуль 3. Объектно-ориентированное программирование

1. Проект «Городская среда»
 - Выполнение заданий на платформе по планированию и реализации проекта. Презентация реализованного проекта.
 - Кастомизация реализованного проекта. Повторение материала, изученного в рамках модуля.
2. ООП. Объекты и методы
 - Изучение понятий: «объекты», «поля и методы объектов». Синтаксис создания объектов.
 - Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью передачи объекта в функцию.
 - Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.
3. Практическое занятие по доработке проекта "Черепашьи бега"
4. ООП. События
 - Изучение типов событий в программе. Обработка взаимодействия пользователя и программы. Обработка событий мыши.
 - Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку событий.
 - Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.
5. Практическое занятие по доработке проекта "Поймай черепашку"
6. ООП. Проект Simple Paint 2035
 - Выполнение заданий на платформе по планированию и реализации проекта. Презентация реализованного проекта.
 - Кастомизация реализованного проекта. Изучение теории для следующего занятия.
7. Классы
8. Наследование
9. Прохождение аттестации по модулю.

Модуль 4. Разработка игр на Pygame

Изучение библиотеки Pygame для создания 2D-игр. Реализация и презентация итоговых проектов. Подведение итогов обучения на командном хакатоне

1. Знакомство с Pygame. События
 - Библиотека Pygame. Спрайт. Игровой цикл: обработка событий, обновление игры, прорисовка, подсчет времени. Расположение спрайта. Обработка списков.
 - Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ для создания и расположения спрайтов. Разработка программ, осуществляющих

- обработку списков.
- Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.
2. Практическое занятие по доработке проекта "Черепашьи бега"
 3. Игра Fast Clicker
 - Разбор технического задания. Планирование проекта. Игровая сцена, игровой таймер, частота кадров, отображение текста, создание спрайта-карточки. Элементы интерфейса для счетчиков статистики. Условия победы и проигрыша.
 - Разработка игры Fast Clicker. Обработка событий мыши. Работа с модулем time: отображение надписи на случайном спрайте ограниченное количество времени.
 - Самостоятельная доработка проекта разработанного в рамках урока. Изучение теории для следующего занятия.
 4. Игра «Арканоид»
 - Разработка игры «Арканоид». Работа с большим количеством спрайтов. Автоматическое движение спрайта. Обработка нажатий на клавиши. Презентация проекта.
 - Самостоятельная доработка проекта, разработанного в рамках урока.
 5. Прохождение аттестации по модулю.

Тематическое планирование

Модуль 1. Основы языка, управляющие конструкции

Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Введение в язык Python	Изучение понятий: «язык программирования», «алгоритм», «программа». Знакомство с синтаксисом языка. Функция ввода input(). Функция вывода print(). Необходимость вывода данных.	теоретические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программы, осуществляющей ввод-вывод данных, ввод команд с клавиатуры.	практические занятия	1
	Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.		
Переменные	Изучение понятий: данные, типы данных, переменная, числовой тип данных, оператор присваивания, приоритет операций.	теоретические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку переменных числового типа. Изменение типов данных с помощью функций int() и str()	практические занятия	1
Строки	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку строковых переменных (сумма строк, повтор строки n-раз, вычисление длины строки, получение символа строки по номеру, замена подстроки на другую подстроку, разбиение строки по разделителю).	практические занятия	1
Вложенные конструкции	Выполнение автопроверяемых заданий на	практические	1

	платформе по пройденной теме. Разработка программ, содержащих вложенные конструкции.	занятия	
	Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.		
Практическое занятие по темам: "Переменные", "Строки", "Вложенные конструкции"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	1
	Изучение теории для следующего занятия.		
Условный оператор	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих условный оператор.	практические занятия	1
	Изучение теории для следующего занятия.		
Вложенный условный оператор	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих вложенный условный оператор.	практические занятия	1
	Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.		
Практическое занятие по темам: "Условный оператор", "Вложенный условный оператор"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	1
	Изучение теории для следующего занятия.		
Циклы	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих циклы различных типов.	практические занятия	2
	Изучение теории для следующего занятия.		
Вложенные управляющие конструкции	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью вложенных алгоритмических конструкций.	практические занятия	1
	Подготовка к аттестации по модулю.		
Прохождение аттестации по модулю.			2
		Объем в ак.ч.	Объем в %
ИТОГО:	теоретические занятия	2	12%
	практические занятия	13	76%
	аттестация	2	12%
Всего:		17	

Модуль 2. Функции, модуль Turtle

Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Практикум по решению задач	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих материалы всего модуля.	практические занятия	2
	Изучение теории для следующего занятия.		
Функции	Изучение понятий: «функция», «локальная и глобальная функции», «оператор def», «параметры функции», «значение функции». Изучение синтаксиса создания и вызова функции. Использование одной функции внутри другой.	теоретические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью функций и содержащих функцию внутри другой функции.	практические занятия	2
	Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.		
Практическое занятие по теме "Функции"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	1
	Изучение теории для следующего занятия.		
Модули	Изучение понятий: «модуль», «встроенные модули стандартной библиотеки». Использование модулей random и time. Синтаксис создания и подключения модуля.	теоретические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью модулей, встроенных в стандартную библиотеку, и модулей, не входящих в стандартную библиотеку.	практические занятия	2
	Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.		
Практическое занятие по теме "Модули"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2
	Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме. Изучение теории для следующего занятия.		
Хакатон	Выполнение заданий на платформе. Создание консольного приложения на заданную тему, используя инструменты изученные за модуль.	практические занятия	1
	Изучение теории для следующего занятия.		

Turtle. Линейные алгоритмы	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих рисование геометрических фигур с использованием условного оператора.	практические занятия	1
	Изучение теории для следующего занятия.		
Turtle. Циклы	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих рисование многоугольников.	практические занятия	1
	Изучение теории для следующего занятия.		
Turtle. Вложенные структуры данных	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих рисование геометрических фигур с использованием условного оператора.	практические занятия	1
	Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.		
Прохождение аттестации по модулю.			2
		Объем в ак.ч.	Объем в %
ИТОГО:	теоретические занятия	2	12%
	практические занятия	13	76%
	аттестация	2	12%
		Всего:	17

Модуль 3. Объектно-ориентированное программирование

Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Проект «Городская среда»	Выполнение заданий на платформе по планированию и реализации проекта. Презентация реализованного проекта.	практические занятия	2
	Кастомизация реализованного проекта. Повторение материала, изученного в рамках модуля.		
ООП. Объекты и методы	Изучение понятий: «объекты», «поля и методы объектов». Синтаксис создания объектов.	теоретические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью передачи объекта в функцию.	практические занятия	1
	Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.		

Практическое занятие по доработке проекта "Черепашьи бега"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме.		
ООП. События	Изучение типов событий в программе. Обработка взаимодействия пользователя и программы. Обработка событий мыши.	теоретические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку событий.	практические занятия	1
	Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.		
Практическое занятие по доработке проекта "Поймай черепашку"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	1
	Изучение теории для следующего занятия.		
ООП. Проект Simple Paint 2035	Выполнение заданий на платформе по планированию и реализации проекта. Презентация реализованного проекта.	практические занятия	2
	Кастомизация реализованного проекта. Изучение теории для следующего занятия.		
Классы	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку классов.	практические занятия	1
	Изучение теории для следующего занятия.		
Наследование	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, использующих наследование.	практические занятия	2
	Изучение теоретического материала на платформе. Решение задач, которые включают в себя тестовые задания и задание на написание кода.		
Практическое занятие по темам "Классы", "Наследование"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2
	Изучение теории для следующего занятия.		
Прохождение аттестации по модулю.			2
		Объем в ак.ч.	Объем в %
ИТОГО:	теоретические занятия	2	12%
	практические занятия	13	76%
	аттестация	2	12%
	Всего:	17	

Модуль 4. Разработка игр на Pygame

Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Знакомство с Pygame. События	Библиотека Pygame. Спрайт. Игровой цикл: обработка событий, обновление игры, прорисовка, подсчет времени. Расположение спрайта. Обработка списков.	теоретические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ для создания и расположения спрайтов. Разработка программ, осуществляющих обработку списков.	практические занятия	1
	Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода. Изучение теории для следующего занятия.		
Практическое занятие по доработке проекта "Черепашьи бега"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	1
	Выполнение автопроверяемых заданий на платформе по пройденной теме.		
Игра Fast Clicker	Разбор технического задания. Планирование проекта. Игровая сцена, игровой таймер, частота кадров, отображение текста, создание спрайта-карточки. Элементы интерфейса для счетчиков статистики. Условия победы и проигрыша.	теоретические занятия	1
	Разработка игры Fast Clicker. Обработка событий мыши. Работа с модулем time: отображение надписи на случайном спрайте ограниченное количество времени.	практические занятия	5
	Самостоятельная доработка проекта разработанного в рамках урока. Изучение теории для следующего занятия.		
Игра «Арканоид»	Разработка игры «Арканоид». Работа с большим количеством спрайтов. Автоматическое движение спрайта. Обработка нажатий на клавиши. Презентация проекта.	практические занятия	6
	Самостоятельная доработка проекта, разработанного в рамках урока.		
Прохождение аттестации по модулю.			2
		Объем в ак.ч.	Объем в %
ИТОГО:	теоретические занятия	2	12%
	практические занятия	13	76%
	аттестация	2	12%
Всего:		17	

Календарно-тематическое планирование

№	№ модуля и тема	Тема занятия	Кол-во занятий	Дата
1	Модуль 1. Основы языка, управляющие конструкции	Цели и задачи курса. ТБ, программное обеспечение.	1	03.10.2023
2		Введение в язык Python	1	07.10.2023
3		Переменные	1	10.10.2023
4		Типы переменных	1	14.10.2023
5		Строки	1	17.10.2023
6		Вложенные конструкции	1	21.10.2023
7		Практическое занятие по темам: "Переменные", "Строки"	1	24.10.2023
8		Практическое занятие по теме: "Вложенные конструкции"	1	28.10.2023
9		Условный оператор	1	31.10.2023
10		Вложенный условный оператор	1	04.11.2023
11		Практическое занятие по темам: "Условный оператор", "Вложенный условный оператор"	1	07.11.2023
12		Циклы с условием	1	11.11.2023
13		Циклы с параметром	1	14.11.2023
14		Вложенные управляющие конструкции	1	18.11.2023
15		Вложенные управляющие конструкции	1	21.11.2023
16	Аттестация		1	25.11.2023
17	Заключительное занятие по модулю		1	28.11.2023
18	Модуль 2. Функции, модуль Turtle	Практикум по решению задач	1	02.12.2023
19		Функции	1	05.12.2023
20		Функции	1	09.12.2023
21		Практическое занятие по теме "Функции"	1	12.12.2023
22		Практическое занятие по теме "Функции"	1	16.12.2023
23		Модули	1	19.12.2023
24		Модули	1	23.12.2023
25		Практическое занятие по теме "модули"	1	26.12.2023
26		Практическое занятие по теме "модули"	1	30.12.2023
27		Хакатон	1	09.01.2023
28		Turtle. Линейные алгоритмы	1	13.01.2024
29		Turtle. Циклы	1	16.01.2024
30		Практическое занятие по теме "Turtle. Линейные и циклические алгоритмы"	1	20.01.2024
31		Turtle. Вложенные структуры данных	1	23.01.2024
32		Практическое занятие по теме "Turtle. Вложенные структуры"	1	27.01.2024

33	Аттестация		1	30.01.2024
34	Заключительное занятие по модулю		1	03.02.2024

35	Модуль 3. Объектно-ориентированное программирование	Проект «Городская среда»	1	06.02.2024
36		Практическое занятие по теме "Городская среда"	1	10.02.2024
37		ООП. Объекты и методы	1	13.02.2024
38		Практическое занятие по теме "ООП. Объекты и методы"	1	17.02.2024
39		Практическое занятие по доработке проекта "Черепашьи бега"	1	20.02.2024
40		ООП. События	1	24.02.2024
41		Практическое занятие по теме "ООП. События"	1	27.02.2024
42		Практическое занятие по доработке проекта "Поймай черепашку"	1	02.03.2024
43		ООП. Проект Simple Paint	1	05.03.2024
44		ООП. Проект Simple Paint	1	12.03.2024
45		Классы	1	16.03.2024
46		Практическое занятие по теме "Классы"	1	19.03.2024
47		Наследование	1	23.03.2024
48		Наследование	1	26.03.2024
49		Практическое занятие по теме "Наследование"	1	30.03.2024
50	Аттестация		1	02.04.2024
51	Заключительное занятие по модулю		1	06.04.2024

52	Модуль 4. Разработка игр на Pygame	Знакомство с Pygame.	1	09.04.2024
53		События	1	13.04.2024
54		Практическое занятие по доработке проекта "Черепашьи бега"	1	16.04.2024
55		Игра Fast Clicker	6	20.04.2024
56				23.04.2023
57				27.04.2023
58				30.04.2023
59				04.05.2023
60				07.05.2023
61		Игра «Арканоид»	6	11.05.2023
62				14.05.2023
63				18.05.2023
64				21.05.2023
65				25.05.2023
66				28.05.2023
67	Аттестация		1	01.06.2023
68	Заключительное занятие по модулю		1	04.06.2023

Учебно-методические материалы

Методы, формы и технологии	<p>Методы и формы обучения На протяжении всего курса используются смешанные формы и методы обучения.</p> <ul style="list-style-type: none"> ● Лекционный формат. ● Устный опрос. ● Тестирование. ● Практические работы. ● Дискуссия. ● Презентация учениками результатов своей работы. ● Индуктивный и дедуктивный методы. ● Соревнования и турниры. <p>Педагогические технологии</p> <ul style="list-style-type: none"> ● Проектная деятельность: курс включает несколько индивидуальных мини-проектов с явной координацией со стороны учителя. ● Проблемное обучение: каждый урок построен на решении конкретной проблемы, которая может возникнуть в процессе разработки игры. ● Групповое обучение: курс включает работу в парах, во время которой один ученик даёт обратную связь по проекту другого. ● Индивидуализация обучения: задания и теоретический материал доступны ученикам в любое время. ● Модульное обучение. ● Игровые технологии.
Методические разработки	Размещенные на образовательной платформе поурочные методические рекомендации к занятиям. Тематические презентации. Учебные задачи для отработки содержания изучаемой темы.
Материалы модуля	Каждый из представленных на платформе модулей содержит несколько уроков. Каждый урок содержит в себе несколько блоков заданий, объединённых одной из изученных на уроке тем. Большая часть заданий проверяется автоматически. Во время урока преподаватель объясняет ученикам тему урока, подкрепляя объяснения графическими материалами и примерами кода, обсуждает тему с учениками, вовлекая их в диалог и отвечает на появившиеся у ученика вопросы. После этого ученик переходит на платформу и решает задания из соответствующего блока заданий при поддержке преподавателя. В любой момент времени (на уроке, во время практической части или самостоятельно работы, вне урока) ученик имеет доступ ко всем представленным в пройденных уроках заданиям. Также ученик имеет доступ к теоретическому материалу урока и в любой момент времени может самостоятельно повторить пройденные темы.
Учебная литература	<ol style="list-style-type: none"> 1. Брайсон Пэйн. Python для детей и родителей. — ООО «Эксмо», 2017. — 352 с. 2. Вордерман Кэрол, Джон Вудкок, Шон Макаманус. Программирование для детей. Иллюстрированное руководство для детей; пер. с англ. С. Ломакина. — 4-е изд. М.: Манн, Иванов и Фербер, 2018, — 224 с. 3. Джейсон Бриггс. Python для детей. Самоучитель по программированию. пер. с англ. С. Ломакина. — Манн, Иванов и Фербер, 2022, — 320 с. 4. Лапчик М.П., Семакин И.Г., Хеннер Е.К., Самылкина Н.Н. Теория и методика обучения информатике. М.: Академия, 2008. — 592 с. 5. Фридман Л.М. Как научиться решать задачи. — М.: Просвещение, 1989.