Администрация муниципального района «Мирнинский район»
Республики Саха (Якутия)
Муниципальное автономное учреждение дополнительного образования
"Центр дополнительного образования" г. Мирный
муниципального района "Мирнинский район"
Республики Саха (Якутия)

П Р И Н Я Т А
на заседании педагогического совета
МАУ ДО «ЦДО» г. Мирный
(протокол № 3 от 29.05.2026г.)

У Т В Е Р Ж Д А Ю
И.о. директора МАУ ДО «ЦДО» г. Мирный
_____ Ч.М. Базыр
Приказ № 146 от «05» августа 2026 г.

Дополнительная общеобразовательная общеразвивающая программа технической направленности «Программирование на Python»

Тип программы: модифицированная
Срок реализации: 3 года
Возраст обучающихся: 10-18 лет
Уровень: базовый и средний

Составитель:
Моякулова Екатерина Николаевна,
педагог дополнительного образования

Мирный, 2026 г.

Внутренняя экспертиза проведена. Программа рекомендована к рассмотрению на педагогическом (методическом) совете учреждения.

Заместитель директора по УВР_____ /Базыр Ч.М./

Заместитель директора по НМР _____/Рожина М.П./

«25» мая 2026 г.

СОДЕРЖАНИЕ

1.	**Раздел 1. Комплекс основных характеристик программы**	3
1.1	Пояснительная записка	3
1.2	Цель и задачи программы	5
1.3	Содержание программы	5
1.3.1	Учебно-тематическое планирование «Базового уровня» (1 год обучения**)**	5
1.3.2	Учебно-тематическое планирование «Среднего уровня» (2 год обучения**)**	10
1.3.3	Учебно-тематическое планирование «Среднего уровня» (3 год обучения**)**	17
1.4	Планируемые результаты	19
1.4.1	Планируемые результаты «Базового уровня» (1 год обучения)	19
1.4.2	Планируемые результаты «Среднего уровня» (2 год обучения)	20
1.4.3	Планируемые результаты «Среднего уровня» (3 год обучения)	20
2.	**Раздел 2. Комплекс организационно-педагогических условий**	21
2.1	Календарный учебный график	21
2.2	Условия реализации программы	21
2.3	Формы аттестации	22
2.4	Оценочный материал	23
2.5	Методические материалы	23
3.	**Раздел 3. Список литературы**	25
	Приложения	26
	Приложение 1 Вводная диагностика (базовый уровень)	26
	Приложение 2 Вводная диагностика (средний уровень)	27
	Приложение 3 Промежуточная аттестация (базовый уровень)	28
	Приложение 4 Промежуточная аттестация (2 год обучения)	29
	Приложение 5 Промежуточная аттестация (3 год обучения)	30
	Приложение 6 Итоговая аттестация (базовый уровень)	32
	Приложение 7 Итоговая аттестация (2 год обучения)	34
	Приложение 8 Итоговая аттестация (3 год обучения)	36
	Приложение 9 Диагностическая карта	38
	Приложение 10 Календарный учебный график	39
	Приложение 11 Рабочая программа воспитания и календарный план воспитательной работы студии «Программирование на Python» на 2025-2026 учебный год	65

РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» имеет техническую направленность.

Дополнительная общеобразовательная общеразвивающая программа составлена с учетом:

Федерального Закона Российской Федерации от 29.12.2012 273-ФЗ «Об образовании в Российской Федерации»;

Приказ Министерства просвещения РФ “Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам” от 27.07. 2022 г. N 629;

Постановление Государственного санитарного врача РФ от 28.09.2020г. "Об утверждении санитарных правил СП-2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи"" (СП-2.4.3648-20);

Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (Приложение к письму Департамента государственной политики в сфере воспитания детей и молодежи Министерства образования и науки РФ от 18.11.2015 № 09-3242).

Дополнительная общеобразовательная программа «Программирование на Python» в соответствии с утвержденным Федеральным оператором перечнем направлений, относится к обязательному направлению ЦЦОД «IT-куб».

Актуальность

Язык программирования Python на сегодняшний день является наиболее популярной высокоуровневой средой международного олимпиадного программирования. Python – мощный и простой для изучения язык программирования. В нем представлены проработанные высокоуровневые структуры данных и простой, но эффективный подход к объектно-ориентированному программированию. Данная программа направлена на расширение знаний обучающихся в направлении изучения языков программирования, в частности, языка программирования Python.

Изучение Python в школе откроет ученикам возможности дальнейшего развития в области IT и поможет профориентации в старших классах, пригодится в олимпиадах по программированию и решении заданий ЕГЭ.

Методы, применяемые в процессе обучения, такие как проблемное обучение, проектная деятельность, способствуют формированию мотивации учащихся к углубленному изучению программирования, как одной из компьютерных наук. У детей формируется познавательный интерес, самостоятельность мышления, стремление к самопознанию.

Отличительная особенность программы состоит в том, что она позволяет привлечь детей среднего и старшего школьного возраста к изучению программирования при помощи языка Python, так как он обладает следующими достоинствами:

- Python – это текстовый язык программирования. Он универсален, пригоден для создания самых разных программ, от текстовых процессоров до веб-браузеров;
- Python – простой и удобный язык. По сравнению со многими другими языками читать и составлять программы на Python совсем не сложно;
- В Python есть библиотеки готовых процедур для использования в своих программах. Это позволяет создавать сложные программы быстро;

Новизна заключается в том, что Python дает более широкие возможности в области программирования, чем другие языки программирования, которые входят в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры

и программировать роботов. Этот язык быстрее и легче усваивается. Многие мировые компании такие используют этот язык при реализации своих проектов. Многие крупнейшие интернет-ресурсы разработаны с помощью языка программирования Python.

Адресат дополнительной общеобразовательной программы

Программа составлена с учетом поло -возрастных и индивидуально-психологических, физических особенностей обучающихся и рассчитана на обучающихся 10-18 лет соответствует направлениям деятельности данного объединения и относится к базовому уровню сложности.

Студию могут посещать как мальчики, так и девочки. Количество обучающихся не более 8 человек в группе.

Программа рассчитана на 3 года:

1 год обучения – 2 занятия в неделю по 2 часа. Всего 144 часа.

2 год обучения – 2 занятия в неделю по 2 часа. Всего 144 часа.

3 год обучения – 2 занятия в неделю по 2 часа. Всего 144 часа.

Продолжительность одного академического часа - 45 мин. Перерыв между учебными занятиями – 10 минут. Общее количество часов в неделю – 4 часа.

Формы обучения: Занятия проводятся в очной и очно-заочной формах обучения без использования дистанционных технологий, без использования сетевой формы.

Уровень программы – базовый и средний.

Форма реализации образовательной программы - традиционная.

Организационные формы обучения: Занятия проводятся групповым составом. Группы формируются преимущественно одновозрастные, но также имеются разновозрастные группы. При формировании групп учитываются возраст детей, смены в школах.

Режим занятий: Занятия проводятся 2 раза в неделю по 2 часа. Общее количество часов в неделю – 4 часа. Продолжительность одного академического часа - 45 мин. Перерыв между учебными занятиями – 10 минут.

Дистанционное обучение

С учетом санитарно-эпидемиологической ситуации при необходимости возможна реализация программы в заочной форме обучения с использованием дистанционных технологий, без использования сетевой формы.

Опосредованное осуществление взаимодействия педагога с обучающимися может быть организовано при подготовке к участию в конкурсах на заключительных этапах; для обучающихся с высокой степенью успешности в освоении программ; для обучающихся, пропускающих учебные занятия по уважительной причине (болезнь и др.); в период отмены (приостановки) занятий в очной (контактной) форме. В обучении с применением ЭО и ДОТ могут использоваться следующие организационные формы учебной деятельности:

- лекция;
- презентация;
- онлайн-беседа;
- упражнения;
- практическое занятие;
- контрольная работа;
- самостоятельная работа.

Электронные образовательные ресурсы, используемые при применении дистанционных образовательных технологий:

- Stepik Class — эта платформа бесплатной организации учебных процессов, тестирования, проверки уровня усвояемости информации обучаемыми;
- Яндекс.Диск — облачный сервис, позволяющий пользователям хранить свои данные на серверах в «облаке» и передавать их другим пользователям в Интернете.

Педагогическая целесообразность: заключается в том, что данная программа позволит выявить заинтересованных обучающихся, проявивших интерес к знаниям, оказать

им помощь в формировании устойчивого интереса к программированию на языке Python. В процессе изучения Python обучающиеся научатся программировать на языке будущего, это повысит уровень логического мышления, а также аналитический склад ума.

1.2. Цель и задачи программы

Цель:

Сформировать знания обучающихся в области алгоритмического мышления и программирования на языке Python, подготовить к самостоятельной разработке программных решений, внедряя практические навыки в проектной деятельности через студию «Программирование на Python».

Задачи:

Образовательные:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- объяснить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);
- научить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

Воспитательные:

- формировать правильный методологический подход к познавательной и практической деятельности;
- формирование мотивации обучающихся к изобретательству, созданию собственных программных реализаций;
- способствовать стремлению к овладению техникой исследования;
- воспитывать трудолюбие, инициативность и настойчивость в преодолении трудностей;
- стремление к получению качественного законченного результата в проектной деятельности;
- развивать навыки сознательного и рационального использования компьютера в своей учебной, а впоследствии и в профессиональной деятельности

Развивающие:

- развить познавательные процессы (внимание, восприятие, логическое мышление, память),
- развить креативность,
- развить способности к самореализации.
- содействовать развитию креативности с помощью задач, требующих нестандартных решений.
- поддерживать способности к самореализации через выполнение индивидуальных и групповых проектов, позволяя студентам выразить свои идеи и находить уникальные подходы.

1.3. Содержание программы

1.3.1 Учебно-тематическое планирование «Базового уровня» (1 год обучения)

Основная форма работы теоретической части – лекционные занятия. Практические задания планируется выполнять индивидуально, в парах и в малых группах. Занятия проводятся в виде

бесед, семинаров, лекций: для наглядности изучаемого материала используется различный мультимедийный материал – презентации.

№ п/п	Наименование тем	Количество часов			Формы аттестации/ контроля
		Теория	Практика	Всего	
1	**Введение**	**1**	**1**	**2**	
1.1	Вводный урок. Инструктаж по технике безопасности в компьютерном классе. Знакомство с Python. IDLE.	1	1	2	Теория. Входящий контрольный тест
2	**Операторы ввода\вывода. Переменные.**	**3**	**3**	**6**	
2.1	Команды input() и print().	1	1	2	Решение практических задач
2.2	Параметры sep, end. Переменные. Комментарии. PEP 8	1	1	2	
2.3	Работа с целыми числами.	1	1	2	Тестирование, Решение практических задач
3	**Условный оператор.**	**3**	**9**	**12**	
3.1	Условный оператор. Логические операции and, or, not	2	2	4	Решение практических задач
3.2	Вложенный и каскадный условный оператор	1	1	2	
3.3	Повторение пройденных тем. Решение задач.	0	4	4	Тестирование, Решение практических задач
3.4	Проект Калькулятор	0	2	2	Решение практических задач
4	**Числа и строки**	**3**	**5**	**8**	
4.1	Типы данных int, float, str. Встроенные функции min(), max(), abs(). Оператор in.	1	1	2	Решение практических задач
4.2	Строковый тип данных: индексация и срезы	1	1	2	
4.3	Методы строк	1	1	2	
4.4	Повторение пройденных тем. Решение задач.	0	2	2	Тестирование, Решение практических задач
5	**Библиотеки**	**4**	**6**	**10**	
5.1	Введение в библиотеки. Библиотека Random. Math.	2	2	4	Решение практических задач
5.2	Мини проект игра с использованием библиотеки Random	0	2	2	

5.3	Библиотека Turtle. Черепашечья графика.	2	2	4	
6	**Цикл for.**	**4**	**6**	**10**	
6.1	Цикл for. Функция range().	2	2	4	Решение практических задач
6.2	Использование циклов в черепашечьей графике	2	2	4	
6.3	Повторение пройденных тем. Решение задач.	0	2	2	Тестирование, Решение практических задач
7	**Работа с Turtle**	**3**	**3**	**6**	
7.1	Вывод текста в Turtle, работа с окном ввода.	1	1	2	Решение практических задач
7.2	Анимация в Turtle	1	1	2	
7.3	Обработка событий в Turtle	1	1	2	
8	**Промежуточная аттестация**	0	2	2	**Промежуточная тестовая работа**
9	**Библиотека Random**	**0**	**2**	**2**	
9.1	Задачи с использованием библиотека Random в черепашечьей графике.	0	2	2	Решение практических задач
10	**Проектная работа**	**3**	**3**	**6**	
10.1	Мини проект игра с использованием библиотеки Turtle	2	2	4	Решение практических задач
10.2	Итоговый урок по Turtle, Random. Подведение итогов.	1	1	2	
11	**Цикл while**	**6**	**8**	**14**	
11.1	Урок повторение. Логический тип данных.	1	1	2	Решение практических задач
11.2	Цикл с предусловием while	1	1	2	
11.3	Операторы break, continue, else.	1	1	2	
11.4	Вложенные циклы	1	1	2	
11.5	Частые сценарии при написании циклов. Расширенные операторы присваивания.	2	2	4	
11.6	Повторение пройденных тем. Решение задач.	0	2	2	Тестирование, Решение практических задач
12	**Списки. Массивы.**	**6**	**14**	**20**	
12.1	Введение в списки.	1	1	2	Решение практических задач
12.2	Основы работы со списками. Методы списков	1	1	2	
12.3	Методы списков. Списочные выражения	2	2	4	
12.4	Введение в Массивы. Решение задач с массивами.	2	4	6	

12.5	Повторение пройденных тем. Решение задач.	0	2	2	Тестирование, Решение практических задач
12.6	Мини проект на массивы и списки	0	4	4	
13	**Pygame**	**11**	**11**	**22**	
13.1	Введение в библиотеку Pygame. Простые фигуры.	2	2	4	Решение практических задач
13.2	Pygame. Обзор методов.	2	2	4	
13.3	Pygame. Обработчик событий на клик кнопок клавиатуры. Движение фигур.	2	2	4	
13.4	Картинки в pygame. Загрузка картинок.	1	1	2	
13.5	Музыка и звуки в pygame	1	1	2	
13.6	Pygame. Обработчик событий на клик мыши. Движение фигур.	2	2	4	
13.7	Создание игрового цикла	1	1	2	
14	**Работа над проектом**	**6**	**6**	**12**	
14.1	Начало работы с проектом. Выбор, разбор, анализ проекта	1	1	2	Решение практических задач
14.2	Теория цветов. Цветовая схема.	1	1	2	
14.3	Знакомство с Figma. Создание дизайн макета будущего мини приложения.	2	2	4	
14.4	Реализация дизайна проекта Pygame на основе дизайн макета Figma.	2	2	4	
15	**Итоговая самостоятельная работа работа**	**0**	**2**	**2**	**Итоговая тестовая работа**
16	**Реализация функционала проекта**	**3**	**5**	**8**	
16.1	Реализация функционала проекта. Обработка коллизий.	1	1	2	Решение практических задач
16.2	Реализация функционала проекта. Создание уровней и управление состояниями	1	3	4	
16.3	Сдача проекта. Создание исполняемого файла exe.	1	1	2	
17	**Заключительный урок. Подведение итогов . Ознакомление с планом на следующий год.**	**1**	**1**	**2**	
	ИТОГО:	**57**	**87**	**144**	

Содержание учебного плана «Базового уровня» (1 год обучения)

Раздел 1. Введение - 2 часа

Теория: Учащиеся познакомятся с основами работы в компьютерном классе и изучат технику безопасности. Вводный урок будет посвящен языку программирования Python, а также среде разработки IDLE.

Практика: Выполнение вводного практического задания, в ходе которого учащиеся создадут

свои первые программы на Python. Входящий контрольный тест поможет оценить уровень знаний учащихся.

Раздел 2. Операторы ввода/вывода. Переменные - 6 часов

Теория: Изучение команд `input()` и `print()`, а также параметров `sep` и `end`. Учащиеся ознакомятся с переменными и правилами их использования, следуя стандартам PEP 8.

Практика: Решение практических задач, связанных с вводом и выводом данных, а также работа с целыми числами.

Раздел 3. Условный оператор - 12 часов

Теория: Учащиеся изучат условный оператор и логические операции `and`, `or`, `not`. Также обсудим концепцию вложенных и каскадных условных операторов.

Практика: Решение задач, применение условных операторов в проектах, таких как создание калькулятора.

Раздел 4. Числа и строки - 8 часов

Теория: Изучение типов данных `int`, `float`, `str`, а также встроенных функций `min()`, `max()`, `abs()` и оператора `in`.

Практика: Работа с строками, включая индексацию, срезы и методы строк, решение практических задач.

Раздел 5. Библиотеки - 10 часов

Теория: Введение в библиотеки Python, изучение библиотек `Random` и `Math`.

Практика: Мини проект на основе библиотеки `Random`, работа с библиотекой `Turtle` для создания графики.

Раздел 6. Цикл for - 10 часов

Теория: Изучение цикла `for`, функции `range()`, и его применения в программировании.

Практика: Исп ользование циклов в графике `Turtle`, решение задач с использованием цикла `for`.

Раздел 7. Работа с Turtle - 6 часов

Теория: Учащиеся узнают, как выводить текст в `Turtle`, а также основы анимации и обработки событий.

Практика: Создание графических проектов с использованием библиотеки `Turtle`.

Раздел 8. Промежуточная аттестация - 2 часа

Практика: Учащиеся выполнят промежуточную тестовую работу для оценки усвоенного материала.

Раздел 9. Библиотека Random - 2 часа

Теория: Обсуждение задач, связанных с использованием библиотеки `Random` в графике `Turtle`.

Практика: Решение практических задач с использованием этой библиотеки.

Раздел 10. Проектная работа - 6 часов

Теория: Обсуждение мини проекта на основе библиотеки `Turtle`, подведение итогов изученного материала.

Практика: Создание и защита мини проекта.

Раздел 11. Цикл while - 14 часов

Теория: Изучение цикла `while`, операторов `break`, `continue`, и концепции вложенных циклов.

Практика: Решение задач с использованием цикла `while`, тестирование усвоенного материала.

Раздел 12. Списки. Массивы - 20 часов

Теория: Введение в списки и массивы, методы работы с ними.

Практика: Решение задач с использованием массивов и списков, мини проект на основе изученного материала.

Раздел 13. Pygame - 22 часа

Теория: Учащиеся познакомятся с библиотекой `Pygame`, научатся создавать простые фигуры, обрабатывать события и добавлять звуки.

Практика: Создание простых игр с использованием `Pygame`.
Раздел 14. Работа над проектом - 12 часов
Теория: Обсуждение выбора проекта, анализ и дизайн.
Практика: Реализация дизайна проекта на основе макета, созданного в `Figma`.
Раздел 15. Итоговая самостоятельная работа - 2 часа
Практика: Проведение итоговой тестовой работы для оценки усвоенного материала.
Раздел 16. Реализация функционала проекта - 8 часов
Теория: Учащиеся научатся реализовывать функционал проекта, включая обработку коллизий и создание уровней.
Практика: Создание исполняемого файла проекта и его сдача.
Раздел 17. Заключительный урок. Подведение итогов. Ознакомление с планом на следующий год. - 2 часа
Теория: Подведение итогов, обсуждение успехов и трудностей, а также ознакомление с планом на следующий год.
Практика: Обсуждение результатов обучения и перспектив на будущее.

1.3.2 Учебно-тематическое планирование «Среднего уровня» (2 год обучения)

Основная форма работы теоретической части – лекционные занятия. Практические задания планируется выполнять индивидуально, в парах и в малых группах. Занятия проводятся в виде бесед, семинаров, лекций: для наглядности изучаемого материала используется различный мультимедийный материал – презентации.

№ п/п	Наименование тем	Количество часов			Формы аттестации/ контроля
		Теория	Практика	Всего	
1	**Вводный урок**	**2**	**0**	**2**	
1.1	Инструктаж по технике безопасности в компьютерном классе. Основные понятия и терминология. Инструменты и среды разработки.	2	0	2	Опрос, входное тестирование
2	**Повторение основных типов, циклов, условий**	**2**	**2**	**4**	
2.1	Синтаксис и структура кода. Переменные и типы данных. Условные операторы и циклы. Решение задач на логику	1	1	2	Устный опрос, решение задач.
2.2	Понятие алгоритмов и блок-схем. Решение задач популярных алгоритмов	1	1	2	
3	**Библиотека Tkinter**	**2**	**4**	**6**	
3.1	Основы работы в tkinter.	1	1	2	Решение задач
3.2	Реализация калькулятора на Tkinter	1	3	4	
4	**Введение в библиотеку Turtle**	**2**	**2**	**4**	
4.1	Основные методы библиотеки. Рисование простых фигур. Ввод вывод текста.	1	1	2	Решение задач
4.2	Рисование простой картины	1	1	2	
5	**Решение задач на логику**	**1**	**1**	**2**	Решение задач

6	**Списки**	**2**	**4**	**6**	
6.1	Основы работы со списками. Итерация списков. Индексы и срезы	1	1	2	Устный опрос, решение задач.
6.2	Методы работы со списками	1	1	2	Решение задач
6.3	Вложенные списки. Задачи на списки.	0	2	2	
7	**Аналог игры "Виселица" на Turtle**	**2**	**4**	**6**	
7.1	Проектирование игры. Например, вместо виселицы звезда или снежинка.	1	1	2	Решение задач
7.2	Реализация игры	1	3	4	
8	**Кортежи**	**2**	**2**	**4**	
8.1	Понятие кортежей. Методы кортежей и применение кортежей.	1	1	2	Решение задач
8.2	Примеры задач на кортежи	1	1	2	
9	**Решение задач на логику**	**1**	**1**	**2**	Решение задач
10	**Словари**	**2**	**2**	**4**	
10.1	Основы словарей, методы.	1	1	2	Решение задач
10.2	Примеры задач на словари	1	1	2	
11	**Повторение пройденного материала**	**1**	**1**	**2**	тест
12	**Понятие JSON**	**1**	**1**	**2**	
12.1	Структура JSON. Работа с JSON в Python (модуль `json`). Примеры использования JSON для передачи данных	1	1	2	Решение задач
13	**Pygame: Продвинутые возможности**	**2**	**4**	**6**	
13.1	Работа с изображениями и звуками.	1	1	2	Решение задач
13.2	Спрайты, анимация	1	3	4	
14	**Разработка игры "Змейка"**	**2**	**6**	**8**	
14.1	Проектирование игры.	1	1	2	Решение задач
14.2	Реализация игрового процесса.	1	3	4	
14.3	Добавление уровней и сложности.	0	2	2	
15	**Промежуточная аттестация**	**0**	**2**	**2**	контрольная работа
16	**Решение задач на логику**	**1**	**1**	**2**	Решение задач
17	**Повторение пройденного материала**	**1**	**1**	**2**	Устный опрос, решение задач.
18	**Генератор списков, словарей**	**1**	**1**	**2**	
18.1	Определение генераторов. Применение. Преимущества использования генераторов. Практические примеры.	1	1	2	Решение задач
19	**Множества**	**2**	**2**	**4**	
19.1	Теория множест в математике и в пайтон.	1	1	2	Решение задач

19.2	Методы множеств. Задачи на множества	1	1	2	
20	**Функции**	**2**	**4**	**6**	
20.1	Определение и вызов функций. Возврат значений	1	1	2	Решение задач
20.2	Аргументы и параметры.	1	1	2	
20.3	Решение задач с использованием функций	0	2	2	
21	**Лямбда функции**	**1**	**3**	**4**	
21.1	Определение и использование.	1	1	2	Решение задач
21.2	Примеры применения лямбда функций в Python	0	2	2	
22	**Библиотека Turtle**	**4**	**4**	**8**	
22.1	Установка и настройка. Основные команды Turtle.	1	1	2	Решение задач
22.2	Создание графических картинок из простых фигур	1	1	2	
22.3	Создание сложных рисунков, используя циклы, функции. Выгрузка картинки. Импорт в Фигму.	2	2	4	
23	**Решение задач на логику**	**1**	**1**	**2**	
24	**Библиотека Matplotlib**	**2**	**2**	**4**	
24.1	Установка и основы работы. Построение графиков и визуализация данных. Примеры создания различных типов графиков	1	1	2	Решение задач
24.2	Мини приложение, рисующее график нескольких функций	1	1	2	
25	**Повторение пройденного материала**	**1**	**1**	**2**	тест
26	**Классы и объекты**	**2**	**4**	**6**	
26.1	Основы объектно-ориентированного программирования. Создание классов и объектов.	1	1	2	Решение задач
26.2	Наследование и полиморфизм	1	3	4	
27	**Решение задач на логику**	**1**	**1**	**2**	Решение задач
28	**Разработка телеграмм бота**	**2**	**6**	**8**	
28.1	Основы работы с API Telegram. Использование библиотеки `python-telegram-bot`. Примеры создания простых ботов	1	1	2	Решение задач
28.2	Разработка собственного бота. Добавление дополнительных функций в телеграмм бот.	1	1	2	
28.3	Расширение функционала бота		2	2	
28.4	Использование меню в боте	0	2	2	
29	**Решение задач на логику**	**1**	**1**	**2**	Решение задач
30	**Компьютерное зрение**	**2**	**2**	**4**	

30.1	Введение в компьютерное зрение. Использование библиотеки OpenCV. Примеры обработки изображений	1	1	2	Решение задач
30.2	Обнаружение лиц. Подключение модели. Примеры обработки видео.	1	1	2	
31	**OpenAI и ChatGPT**	**2**	**2**	**4**	
31.1	Основы работы с API OpenAI. Примеры взаимодействия с ChatGPT. Применение в проектах и разработках	1	1	2	Решение задач
31.2	Подключение в телеграм бот	1	1	2	
32	**Повторение пройденного материала**	**1**	**1**	**2**	Устный опрос, решение задач.
33	**Решение задач на логику**	**1**	**1**	**2**	Решение задач
34	**Фигма. Плагины.**	**2**	**2**	**4**	
34.1	Теория цвета. Плагины. Наложение эффектов.	1	1	2	Практическая работа
34.2	Дизайн игры Крестики-Нолики	1	1	2	
35	**Разработка игры "Крестики-нолики"**	**2**	**2**	**4**	
35.1	Проектирование игры.	1	1	2	Решение задач
35.2	Реализация логики и интерфейса	1	1	2	
36	**Итоговая аттестация**	**0**	**2**	**2**	контрольная работа
37	**Разработка игры "Крестики-нолики"**	**0**	**4**	**4**	
37.1	Возможные улучшения и расширения	0	2	2	Решение задач
37.2	Защита проекта "Крестики-нолики"	0	2	2	
38	**Решение задач на логику**	**1**	**1**	**2**	Решение задач
39	**Заключительный урок**	**2**	**0**	**2**	
39.1	Обзор пройденного материала Рекомендации по дальнейшему обучению	2	0	2	Устный опрос
	Итого:	59	85	144	

Содержание учебного плана «Среднего уровня» (2 год обучения)

Раздел 1. Вводный урок – 2 часа

Теория: В данном разделе Учащиеся познакомятся с образовательной программой и ее целями. Будет представлен краткий обзор содержания курса, ключевых тем и ожидаемых результатов. Также проведем инструктаж по технике безопасности, чтобы обеспечить безопасную и продуктивную учебную среду.

Практика: Выполнение входной диагностики для определения начального уровня знаний учащихся.

Раздел 2. Повторение основных типов, циклов, условий - 4 часа

Теория: Учащиеся повторят основные типы данных в Python, включая целые числа, строки,

списки и словари. Также будет рассмотрено использование циклов `for` и `while`, а также условные операторы `if`, `elif` и `else`. Упражнения помогут закрепить эти концепции на практике.
Практика: Решение задач на использование типов данных, циклов и условий.

Раздел 3. Библиотека Tkinter – 6 часов

Теория: Учащиеся познакомятся с библиотекой Tkinter, предназначенной для создания графических интерфейсов. Будут рассмотрены основные компоненты интерфейса, такие как окна, кнопки, метки и текстовые поля. Также обсудим обработку событий и взаимодействие с пользователем.
Практика: Создание простых графических приложений с использованием Tkinter.

Раздел 4. Введение в библиотеку Turtle - 4 часа

Теория: Учащиеся изучат библиотеку Turtle, предназначенную для рисования графики с помощью программирования. Будут рассмотрены основные команды для управления черепашкой, такие как движение, поворот и изменение цвета. Упражнения помогут развить творческие навыки программирования.
Практика: Создание простых рисунков и фигур с использованием библиотеки Turtle.

Раздел 5. Решение задач на логику - 2 часа

Теория: Учащиеся разовьют логическое мышление через решение задач, требующих анализа и критического подхода. Обсудим стратегии решения логических задач и методы работы с различными типами задач.
Практика: Решение логических задач в группах.

Раздел 6. Списки - 6 часов

Теория: Учащиеся изучат списки в Python, их создание, модификацию и использование. Обсудим методы работы со списками, такие как добавление, удаление и сортировка элементов. Также рассмотрим применение списков в реальных задачах.
Практика: Решение задач на использование списков.

Раздел 7. Аналог игры "Виселица" на Turtle - 6 часов

Теория: Учащиеся создадут аналог игры "Виселица" с использованием библиотеки Turtle. Обсудим логику игры, управление вводом пользователя и графическое оформление.
Практика: Разработка игры "Виселица" в группах.

Раздел 8. Кортежи - 4 часа

Теория: Учащиеся познакомятся с кортежами в Python, их особенностями и преимуществами по сравнению со списками. Обсудим, когда целесообразно использовать кортежи и как их создавать и изменять.
Практика: Решение задач на использование кортежей.

Раздел 9. Решение задач на логику – 2 часа

Теория: Продолжение работы над логическими задачами для углубления понимания и применения логического мышления.
Практика: Решение новых логических задач.

Раздел 10. Словари - 4 часа

Теория: Учащиеся изучат словари в Python, их структуру и методы работы с ними. Обсудим, как словари могут использоваться для хранения данных в формате "ключ-значение".
Практика: Решение задач на использование словарей.

Раздел 11. Повторение пройденного материала - 2 часа

Теория: Учащиеся повторят ключевые темы курса, включая типы данных, списки, кортежи и словари. Обсудим важные концепции и ответим на вопросы.
Практика: Тестирование на знание пройденного материала.

Раздел 12. Понятие JSON - 2 часа

Теория: Учащиеся познакомятся с форматом JSON, его структурой и применением для обмена данными между клиентом и сервером. Обсудим, как работать с JSON в Python.
Практика: Преобразование Python-объектов в JSON и обратно.

Раздел 13. Pygame: Продвинутые возможности - 6 часов

Теория: Учащиеся изучат продвинутые функции библиотеки Pygame, включая работу со

звуком, анимацией и событиями. Обсудим создание сложных игровых механик.
Практика: Разработка небольшой игры с использованием Pygame.

Раздел 14. Разработка игры "Змейка" - 8 часов

Теория: Учащиеся создадут игру "Змейка" с использованием Pygame. Обсудим правила игры, управление и графическое оформление.
Практика: Работа в командах для разработки игры.

Раздел 15. Промежуточная аттестация - 2 часа

Теория: Учащиеся пройдут промежуточную аттестацию для оценки своих знаний и навыков, полученных за прошедшие разделы.
Практика: Проведение тестирования и обсуждение результатов.

Раздел 16. Решение задач на логику - 2 часа

Теория: Продолжение работы над логическими задачами, с акцентом на сложные и нестандартные задачи.
Практика: Групповое решение задач.

Раздел 17. Повторение пройденного материала - 2 часа

Теория: Учащиеся повторят основные темы прошедших разделов.
Практика: Тестирование на знание пройденного материала.

Раздел 18. Генератор списков, словарей – 2 часа

Теория: Учащиеся изучат генераторы списков и словарей, их синтаксис и преимущества. Обсудим, как использовать генераторы для упрощения кода.
Практика: Решение задач с использованием генераторов.

Раздел 19. Множества – 4 часа

Теория: Учащиеся познакомятся с множествами в Python, их свойствами и методами работы. Обсудим, как множества могут быть использованы для устранения дубликатов и выполнения математических операций.
Практика: Решение задач на использование множеств.

Раздел 20. Функции – 6 часов

Теория: Учащиеся изучат функции в Python, включая создание, параметры и возврат значений. Обсудим, как функции помогают структурировать код и повышают его читаемость.
Практика: Написание функций и использование их в проектах.

Раздел 21. Лямбда функции – 4 часа

Теория: Учащиеся познакомятся с лямбда-функциями, их синтаксисом и применением для создания анонимных функций. Обсудим, когда использовать лямбда-функции вместо обычных.
Практика: Применение лямбда-функций в коде.

Раздел 22. Библиотека Turtle - 8 часов

Теория: Углубленное изучение библиотеки Turtle, включая создание анимаций и сложных графических элементов. Учащиеся научатся использовать Turtle для создания визуальных представлений данных.
Практика: Разработка проекта с использованием библиотеки Turtle.

Раздел 23. Решение задач на логику – 2 часа

Теория: Учащиеся продолжат развивать свои логические навыки через решение новых задач.
Практика: Групповое решение логических задач.

Раздел 24. Библиотека Matplotlib – 4 часа

Теория: Учащиеся познакомятся с библиотекой Matplotlib для визуализации данных. Обсудим создание различных типов графиков и диаграмм.
Практика: Построение графиков на основе предоставленных данных.

Раздел 25. Повторение пройденного материала – 2 часа

Теория: Учащиеся повторят ключевые темы, изученные в предыдущих разделах.
Практика: Тестирование на знание пройденного материала.

Раздел 26. Классы и объекты - 6 часов

Теория: Учащиеся изучат основы объектно-ориентированного программирования, включая

создание классов, объектов и использование методов. Обсудим принципы инкапсуляции и наследования.
Практика: Создание простых классов и объектов.
Раздел 27. Решение задач на логику – 2 часа
Теория: Продолжение работы над логическими задачами для развития навыков.
Практика: Решение новых логических задач.
Раздел 28. Разработка телеграмм бота – 8 часов
Теория: Учащиеся узнают, как создавать телеграмм ботов с использованием API Telegram. Обсудим основные команды и методы взаимодействия с пользователем.
Практика: Разработка собственного телеграмм бота.
Раздел 29. Решение задач на логику – 2 часа
Теория: Углубление в решение логических задач.
Практика: Групповое решение задач.
Раздел 30. Компьютерное зрение – 4 часа
Теория: Учащиеся познакомятся с основами компьютерного зрения и библиотекой OpenCV. Обсудим, как обрабатывать изображения и видео, а также основные алгоритмы компьютерного зрения.
Практика: Простые проекты с использованием OpenCV.
Раздел 31. OpenAI и ChatGPT – 4 часа
Теория: Учащиеся узнают о технологиях OpenAI и ChatGPT, их применении и возможностях. Обсудим, как использовать эти инструменты для разработки интеллектуальных приложений.
Практика: Создание простого приложения с использованием ChatGPT.
Раздел 32. Повторение пройденного материала – 2 часа
Теория: Учащиеся повторят ключевые темы, изученные в предыдущих разделах.
Практика: Тестирование на знание пройденного материала.
Раздел 33. Решение задач на логику – 2 часа
Теория: Углубление в решение логических задач.
Практика: Групповое решение задач.
Раздел 34. Фигма. Плагины. – 4 часа
Теория: Учащиеся познакомятся с инструментами Figma и ее плагинами для дизайна интерфейсов. Обсудим, как использовать Figma для создания прототипов и дизайнов.
Практика: Создание простого проекта в Figma.
Раздел 35. Разработка игры "Крестики-нолики" – 4 часа
Теория: Учащиеся создадут игру "Крестики-нолики", изучая логику игрового процесса и создание интерфейса. Обсудим, как реализовать игру для двух игроков.
Практика: Разработка игры "Крестики-нолики".
Раздел 36. Итоговая аттестация – 2 часа
Теория: Учащиеся пройдут итоговую аттестацию для проверки своих знаний и навыков, полученных за весь курс.
Практика: Проведение тестирования и обсуждение результатов.
Раздел 37. Разработка игры "Крестики-нолики" – 4 часа
Теория: Продолжение работы над игрой "Крестики-нолики", с акцентом на улучшение интерфейса и добавление новых функций.
Практика: Завершение разработки игры.
Раздел 38. Решение задач на логику – 2 часа
Теория: Углубление в решение логических задач.
Практика: Групповое решение задач.
Раздел 39. Заключительный урок – 2 часа
Теория: Подведение итогов курса, обсуждение полученных знаний и навыков. Учащиеся смогут поделиться своим опытом и задать вопросы.
Практика: Обсуждение проектов и результатов курса.

1.3.3 Учебно-тематическое планирование «Средний уровень» (3 год обучения)

Основная форма работы теоретической части – лекционные занятия. Практические задания планируется выполнять индивидуально, в парах и в малых группах. Занятия проводятся в виде бесед, семинаров, лекций: для наглядности изучаемого материала используется различный мультимедийный материал – презентации, тесты.

№	Наименование тем	Количество часов			Формы аттестации/ контроля
		Теория	Практика	Всего	
1	**Вводный урок. Инструктаж.**	1	1	2	
2	**Повторение пройденного материала**	8	12	20	Тест, практическая работа
2.1	Типы данных: int, float, str, bool	2	2	4	
2.2	Структуры данных: списки, кортежи	2	2	4	
2.3	Словари и множества	2	4	6	
2.4	Условные конструкции и циклы	2	4	6	
3	**Объектно-ориентированное программирование (ООП)**	8	18	26	Проект: игра/приложение
3.1	Классы, объекты, наследование	4	6	10	
3.2	Полиморфизм, инкапсуляция	2	6	8	
3.3	Практика: создание игры с ООП	2	6	8	
4	**Основы HTML и веб-разработки**	4	4	8	Веб-страница
4.1	Структура HTML, CSS (базово)	2	2	4	
4.2	Адаптивная верстка (основы)	1	1	2	
4.3	Понятие деплоя, тонкости и методы	1	1	2	
5	**Промежуточная аттестация**	0	2	2	Тест, практическая работа
6	**Основы веб-приложений**	7	13	20	Проект: Веб-приложение
6.1	Основы веб-приложений	1	1	2	
6.2	Flask/Django: простое веб-приложение	2	6	8	
6.3	Валидация, семантические теги	2	2	4	
6.4	Проект на веб-приложение	2	4	6	
7	**Парсинг данных**	6	10	16	Парсер новостей/сайта
7.1	Библиотеки Requests, BeautifulSoup	4	6	10	
7.2	Работа с API (JSON)	2	4	6	
8	**Анализ данных**	10	16	26	Анализ датасета
8.1	Pandas: обработка таблиц	6	8	14	
8.2	Визуализация (Matplotlib/Seaborn)	2	4	6	
8.3	Подготовка отчета по данным	2	4	6	
9	**Работа с файлами и базами данных**	4	8	12	Мини-БД

9.1	CSV, JSON, SQLite	2	4	6	
9.2	ORM (SQLAlchemy)	2	4	6	
10	**Итоговая аттестация**	0	2	2	Тест + практика
11	**Итоговый проект**	1	7	8	Защита проекта
11.1	Определение идеи, создание дизайна	1	1	2	
11.2	Реализация проекта	0	2	2	
11.3	Оформление проекта и защита	0	4	4	
12	**Заключительный урок. Подведение итогов года, планы на следующий год**	2	0	2	
	Итого:	51	93	144	

Содержание учебного плана «Средний уровень» (3 год обучения)

Раздел 1. Вводный урок. Инструктаж. – 2 часа
Теория: Учащиеся познакомятся с правилами поведения и техникой безопасности в компьютерном классе. Будет проведён инструктаж по работе с оборудованием. Также будут рассмотрены возможности сред разработки (PyCharm, VS Code), их настройка и основные функции.
Практика: Выполнение практического задания для оценки базовых навыков программирования. Проведение входного тестирования для выявления уровня подготовки учащихся.
Раздел 2. Повторение пройденного материала – 20 часов
Теория: Повторение ключевых тем базового уровня: типы данных, структуры данных (списки, кортежи, словари, множества), условные конструкции и циклы. Особое внимание будет уделено применению этих знаний в более сложных задачах.
Практика: Решение комплексных задач, требующих комбинированного использования различных структур данных и логических конструкций. Выполнение практической работы и тестовой проверки знаний.
Раздел 3. Объектно-ориентированное программирование (ООП) – 26 часов
Теория: Изучение принципов ООП: классы и объекты, наследование, полиморфизм, инкапсуляция. Рассмотрение примеров и реальных кейсов применения ООП в разработке программ.
Практика: Реализация собственных классов, создание проектов с использованием ООП. Разработка игры или приложения по аналогии с реальными приложениями.
Раздел 4. Основы HTML и веб-разработки – 8 часов
Теория: Изучение структуры HTML-документа, базовых тегов и понятий CSS. Знакомство с адаптивной версткой и понятием деплоя — как опубликовать сайт.
Практика: Создание простой веб-страницы с использованием HTML и CSS. Добавление базового оформления и проверка отображения на разных устройствах.
Раздел 5. Промежуточная аттестация – 2 часа
Теория: Краткий теоретический разбор ключевых моментов во избежание ошибок.
Практика: Выполнение теста и практического задания для проверки знаний по пройденным разделам.
Раздел 6. Основы веб-приложений – 20 часов
Теория: Знакомство с фреймворками Flask и Django. Изучение жизненного цикла веб-приложения. Основы валидации и семантической разметки.
Практика: Создание простого веб-приложения на Flask/Django. Реализация маршрутов, шаблонов и форм. Тестирование и запуск готового проекта.
Раздел 7. Парсинг данных – 16 часов

Теория: Работа с библиотеками Requests и BeautifulSoup. Основы HTTP-запросов, анализ HTML-структуры сайта. Работа с API, формат JSON.
Практика: Создание парсера новостей или данных с популярного ресурса. Получение информации через API.

Раздел 8. Анализ данных – 26 часов

Теория: Использование библиотек Pandas, Matplotlib и Seaborn. Формирование представления о сборе, обработке и визуализации данных. Подготовка отчетов и выводов.
Практика: Анализ реального датасета. Построение графиков, диаграмм, составление итогового отчета по результатам анализа.

Раздел 9. Работа с файлами и базами данных – 12 часов

Теория: Форматы хранения данных: CSV, JSON. Работа с SQLite, использование ORM (SQLAlchemy).
Практика: Создание и работа с базами данных, сохранение и чтение данных из файлов. Создание мини-БД для хранения пользовательских данных.

Раздел 10. Итоговая аттестация – 2 часа

Теория: Подготовка к итоговой проверке знаний.
Практика: Выполнение теста и практического задания, объединяющего несколько тем курса.

Раздел 11. Итоговый проект – 8 часов

Теория: Определение идеи проекта, выбор технологий, планирование этапов реализации.
Практика: Создание законченного проекта (Telegram-бот, сервис анализа данных, веб-приложение и т.п.). Оформление презентации и материалов проекта. Защита перед преподавателем и одногруппниками.

Раздел 12. Заключительный урок. Подведение итогов года, планы на следующий год – 2 часа

Теория: Обсуждение успехов и достижений за год. Разбор ошибок, рекомендации по дальнейшему обучению.
Практика: Ответы на вопросы. Анкетирование и обратная связь. Планирование следующего года обучения.

1.4. Планируемые результаты

Личностные результаты:

-готовность и способность к образованию и самообразованию на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности;
-способность оценить собственные возможности с точки зрения дальнейшей образовательной перспективы в области выбранного профиля;
- осознанный выбор будущей профессии на основе понимания ее ценностного содержания и возможностей реализации собственных жизненных планов;

Метапредметные результаты:

-владение навыками познавательной, учебно-исследовательской деятельности, навыками разрешения проблем; способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания для изучения различных сторон окружающей действительности;
-умение понимать проблему, умение понимать алгоритмический язык;
-умение работать с разными источниками информации, находить ее, анализировать, использовать в самостоятельной деятельности;
-умение правильно осмыслить цель, наметить пути ее достижения, сформировать организаторские умения и лидерские качества.

Предметные результаты:

1.4.1 Планируемые результаты «Базового уровня» (1 год обучения)

В рамках курса «Программирования на Python» «Базовый уровень» учащиеся овладевают следующими знаниями, умениями и способами деятельности:

– составлять алгоритмы для решения учебных задач различных типов;
– выражать алгоритм решения задачи различными способами (словесным, графическим, в том числе и в виде блок-схемы, с помощью формальных языков и др.);
– определять наиболее оптимальный способ выражения алгоритма для решения конкретных задач (словесный, графический, с помощью формальных языков);
– определять результат выполнения заданного алгоритма (программы) или его фрагмента;
– использовать термины «исполнитель», «алгоритм», «программа», а также понимать разницу между употреблением этих терминов в обыденной речи и в информатике;
– выполнять без использования компьютера («вручную») несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных, записанные на конкретном язык программирования с использованием основных управляющих конструкций последовательного программирования (линейная программа, ветвление, повторение, вспомогательные алгоритмы);
– составлять несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных с использованием основных управляющих конструкций последовательного программирования и записывать их в виде программ на выбранном языке программирования;
– выполнять эти программы на компьютере;
– использовать величины (переменные) различных типов, табличные величины (массивы), а также выражения, составленные из этих величин;
– использовать оператор присваивания;
– анализировать предложенный алгоритм, например, определять какие результаты возможны при заданном множестве исходных значений;
– использовать логические значения, операции и выражения с ними;
– записывать на выбранном языке программирования арифметические и логические выражения и вычислять их значения.

1.4.2 Планируемые результаты «Среднего уровня» (2 год обучения)

В рамках курса «Программирования на Python» «Средний уровень» (2 год обучения) учащиеся овладевают следующими знаниями, умениями и способами деятельности:
–знание базового синтаксиса и инструментария языка программирования Python, умение применять язык программирования Python на практике;
– иметь навык работы в Черепашечьей графике;
– иметь представление о компьютерном зрении, и уметь применять компьютерное зрение;
– выполнять без использования компьютера («вручную») несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных, записанные на конкретном язык программирования с использованием основных управляющих конструкций последовательного программирования (линейная программа, ветвление, повторение, вспомогательные алгоритмы);
– иметь навык работы с библиотекой matplotlib, строить разные графики и диаграммы;
– использовать величины (переменные) различных типов, в том числе списки, кортежи, словари, множества;
– понимать различия списков, множеств, словарей, кортежей;
– анализировать предложенный алгоритм, например, определять какие результаты возможны при заданном множестве исходных значений;
– умение применять объектно-ориентированную парадигму в программировании;
– навык разработки эффективных алгоритмов и программ на основе изучения языка программирования Python.

1.4.3 Планируемые результаты «Среднего уровня» (3 год обучения)

В рамках курса «Программирования на Python. Вторая ступень» учащиеся овладевают следующими знаниями, умениями и способами деятельности:
– знать основные принципы объектно-ориентированного программирования (ООП):

классы, объекты, наследование, полиморфизм, инкапсуляция; методы анализа и визуализации данных;

- разбираться в структуре и синтаксисе HTML и CSS для создания веб-страниц;
- овладеть базовыми понятиями и технологиями веб-разработки (Flask/Django), принципами работы с API и основы парсинга веб-данных;
- уметь проектировать и реализовывать программы с использованием ООП;
- создавать и отлаживать веб-приложения на основе фреймворков Flask или Django;
- разрабатывать парсеры веб-страниц и взаимодействовать с внешними API;
- обрабатывать и анализировать данные с помощью библиотек Pandas и NumPy;
- строить графики, диаграммы и визуализировать данные с помощью Matplotlib и Seaborn;
- сохранять, извлекать и обрабатывать данные из файлов и баз данных (SQLite, ORM);
- применять полученные знания для решения практических задач, включая проектную деятельность;
- участвовать в командной разработке программного обеспечения;
- планировать и организовывать этапы выполнения проектов;
- анализировать и выбирать оптимальные алгоритмы и структуры данных;
- применять навыки критического мышления и логического анализа при решении задач.

РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

2.1 Календарный учебный график

Календарный учебный график (общий)

Год обучения (уровень)	Дата начала занятий	Дата окончания занятий	Количество учебных недель	Количество учебных дней	Количество учебных часов	Режим занятий	Каникулы зимние	Каникулы летние
1 год обучения (базовый)	8 сентября 2025г.	31 мая 2026г.	36	72	144, 4 часа в нед.	2 раза в нед. по 2 часа	-	01 июня – 31 августа
2 год обучения (средний)	8 сентября 2025 г.	31 мая 2026г.	36	72	144, 4 часа в нед.	2 раза в нед. по 2 часа	-	01 июня – 31 августа
3 год обучения (средний)	8 сентября 2025 г.	31 мая 2026г.	36	72	144, 4 часа в нед.	2 раза в нед. по 2 часа	-	01 июня – 31 августа

Календарный учебный график по группам для «Базового уровня» и «Среднего уровня» см в *Приложении 10*

2.2 Условия реализации программы

Материально-техническое обеспечение:

Учебное помещение соответствует требованиям санитарных норм и правил, установленных Санитарными правилами (Постановление Главного государственного санитарного врача РФ от 28.09.2020г. №28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучениями, отдыха и оздоровления

детей и молодежи»).

Необходимое материальное оснащение кабинета:

- столы для компьютера;
- компьютерные стулья;
- шкафы для дидактических материалов, пособий;
- специальная и научно-популярная литература для педагога и учащихся;
- канцтовары;
- персональный компьютер (на каждого обучающегося);
- мультимедийный проектор;

Информационное обеспечение:

- видеоматериалы разной тематики по программе;
- оргтехника;
- выход в сеть Internet;

Аппаратное обеспечение:

- Процессор не ниже Core2 Duo;
- Объем оперативной памяти не ниже 4 Гб DDR3;
- Дисковое пространство на менее 128 Гб;
- Монитор диагональю на мене 19’;

Программное обеспечение:

- Операционная система Windows 7 Профессиональная или выше;
- Интерпретатор Python версии 3.7 и выше;
- IDE JatBrains PyCharm;
- Foxit Reader или другой просмоторщик PDF файлов;
- WinRAR;
- Пакет офисных программ;
- Adobe Photoshop или другой растровый графический редактор;
- Любой браузер для интернет-серфинга.

Кадровое обеспечение

Реализовывать программу может педагог дополнительного образования, обладающий достаточными знаниями в области педагогики, психологии и методологии, знающий особенности технологии обучения основам программирования на языке Python.

2.3 Формы аттестации

Входящий контроль осуществляется при комплектовании группы в начале учебного года. Вводная диагностика определения уровня умений, навыков, развития детей и их творческих способностей проводится в начале обучения согласно предложенной форме (Приложение 1 и Приложение 2).

Цель − определить исходный уровень знаний учащихся, определить формы и методы работы с учащимися.

Форма контроля: тестирование.

Текущий контроль осуществляется после изучения отдельных тем, раздела программы. В практической деятельности результативность оценивается качеством выполнения практических работ, поиску и отбору необходимого материала, умению работать с различными источниками информации. Анализируются положительные и отрицательные стороны работы, корректируются недостатки. Контроль знаний осуществляется с помощью заданий педагога (решение практических задач средствами языка программирования); взаимоконтроля, самоконтроля и др. Они активизируют, стимулируют работу учащихся, позволяют более полно проявлять полученные знания, умения, навыки.

Промежуточный контроль осуществляется в конце I полугодия учебного года.

Форма контроля: тест, решение практических задач средствами языка программирования.

Итоговый контроль осуществляется в конце учебного года.

Форма контроля: тест, решение практических задач средствами языка программирования.

2.4 Оценочный материал

Отслеживание личностного развития учащихся осуществляется методом наблюдения, анкетирования. По итогам первого полугодия и по итогам года заполняется «Диагностическая карта», в которой проставляется уровень усвоения программы каждым учащимся объединения.

Система промежуточной и итоговой аттестации знаний и умений обучающихся представляется в виде учёта индивидуального результата по каждому контрольному мероприятию и подведения в итоге суммарного балла для каждого обучающегося.

Промежуточная аттестация реализуется посредством оценки решения задач и тестирования (Приложение 3 и Приложение 4). Максимальное количество баллов, которое возможно получить по результатам промежуточной аттестации – 50 баллов.

Итоговая аттестация обучающихся реализуется посредством оценки решения задач и тестирования (Приложение 5 и Приложение 6). Максимальное количество баллов, которое возможно получить по результатам итоговой аттестации – 50 баллов.

Сумма баллов результатов промежуточной аттестации, итоговой аттестации и защиты итогового проекта переводится в один из уровней освоения образовательной программы согласно таблице:

Уровень освоения программы по окончании обучения

Баллы, набранные учащимся.	**Уровень освоения**
0-39	Низкий
40-79	Средний
80-100	Высокий

Входящий контроль: *Тестирование (Приложение 1 и Приложение 2)*

Промежуточный контроль: *Тестирование, решение задач (Приложение 3 - Приложение 5)*

Итоговый контроль: *Тестирование, решение задач (Приложение 6 - Приложение 8)*

Диагностическая карта: *Приложение 9*

2.5 Методические материалы

Форма организации учебного занятия

С целью достижения качественных результатов учебный процесс оснащен современными техническими средствами. Процесс обучения проходит в компьютерном классе. Объясняемый материал демонстрируется на проекторе, все действия производятся преподавателем непосредственно. При обучении используется метод «Делай как я». К каждой теме занятия разработаны презентации теоретического материала. С помощью мультимедийных элементов занятие визуализируется, вызывая положительные эмоции у обучающихся и создавая условия для успешной деятельности каждого ребенка.

Формы работы:

- Занятия

Программа «Программирование на Phyton» рассчитана на 1 учебный год и построена на преемственности занятий. Знания, полученные на предыдущих занятиях, обучающиеся будут применять на следующих. Программа разработана с учётом возрастных особенностей детей 11-14 лет, основное время отводится на выполнение практических работ под руководством педагога. Самостоятельные действия обучающих контролируются преподавателем напрямую, и при необходимости вносятся поправки в работу, но самостоятельная работа ученика должна

преобладать. Все созданные файлы обязательно сохраняются в процессе работы или в конце урока.

- Урок-игра (игровая деятельность) Использование игрового метода на уроках призвано способствовать созданию благоприятной психологической атмосферы общения. Игра способствует развитию у детей произвольного внимания, повышает мотивацию к деятельности. Урок-проект (проектная деятельность) Метод проектов направлен на то, чтобы развить активное самостоятельное мышление ребенка и научить его не просто запоминать и воспроизводить знания, которые дает ему педагог, а уметь применять их на практике. Проектная методика отличается коллективным характером деятельности, которая при этом является творческой и ориентированной на личность обучающегося. Она предполагает высокий уровень индивидуальной и коллективной ответственности за выполнение каждого задания по разработке проекта. Совместная работа группы учащихся над проектом неотделима от коммуникативного взаимодействия обучающихся. Проект является одной из форм организации исследовательской познавательной деятельности, в которой учащиеся занимают активную позицию. При подборе темы проекта педагог должен ориентироваться на интересы и потребности учащихся, их возможности и личную значимость предстоящей работы, практическую значимость результата работы над проектом. Выполненный проект может быть представлен в самых разных формах. Разнообразны и формы презентации проекта: доклад, дискуссия, демонстрация готового векторного рисунка или обработанной фотографии и др. Главным результатом работы над проектом будут актуализация имеющихся и приобретение новых знаний, навыков и умений и их творческое применение в новых условиях. Работа над проектом требует от учащихся самостоятельной деятельности, координации действий, активного исследовательского, исполнительского и коммуникативного взаимодействия. Роль преподавателя заключается в подготовке учащихся к работе над 20 проектом, выборе темы, в оказании помощи учащимся при планировании работы, в текущем контроле и консультировании учащихся по ходу выполнения проекта.
- Открытое занятие, презентация.

Педагогические технологии и приемы

Технология группового обучения, технология игровой деятельности, технология развивающего обучения, технология проектной деятельности, технология коллективной творческой деятельности, технология сотрудничества.

Для привлечения внимания ребенка необходимо не только разнообразие деятельности во время учебного процесса, но также и личностно-ориентированный подход к каждому ребенку. Именно такой подход позволит ребенку раскрыть и развить свои личностные качества.

РАЗДЕЛ 3: СПИСОК ЛИТЕРАТУРЫ

Литература для педагога

1. Бизли, Дэвид М. Python. Подробный справочник. – М.–СПб.: Символ-Плюс, 2010.
2. Лутц, Марк Python. Справочник. – М.: Вильямс, 2015.
3. Дронов В. Django 2.1 Практика создания веб-сайтов на Python. – СПб.: БХВ-Петербург, 2019
4. Бессонов Р. 100 программ в одну строчку кода Python. [Электронный ресурс] Режим доступа: платный
5. Официальный сайт программы [Электронный ресурс] − Режим доступа: https://docs.python.org/, свободный.
6. Сайт, среда разработки для языка Python. [Электронный ресурс] − Режим доступа: https://www.jetbrains.com/pycharm/?fromMenu, свободный.

Литература для родителей и учащихся

1. Сайт / справочные материалы [Электронный ресурс] − Режим доступа: https://metanit.com/python/, свободный.
2. Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] − Режим доступа: http://pythontutor.ru/, свободный.
3. Сайт / Адаптивный тренажер Python [Электронный ресурс] − Режим доступа: https://stepik.org/course/431 , свободный.
4. Сайт / среда разработки для языка Python [Электронный ресурс] − Режим доступа: https://www.jetbrains.com/pycharm/?fromMenu , свободный.

Сайт проекта Open Book Project. Практические примеры на Python Криса Мейерса [Электронный ресурс] − Режим доступа: openbookproject.net , свободны

Приложение 1

Вводная диагностика (базовый уровень)

(максимум-20 баллов)

Дата_____

ФИО _____

Группа_____

1. К расширениям аудиофайлов относятся: (1 балл)

1) exe, txt, msi
2) flac, aac, mpa
3) iso, odt, ai

2. Опишите своими словами, что такое язык программирования: (2 балла)

_____

3. Сколько килобайт в гигабайте? (2 балл)

_____

4. Какую среду разработки программного обеспечения вы знаете? (2 балл)

_____

5. Элемент компьютера, способный хранить информацию только при включенном компьютере? (1 балла)

1) Процессор
2) Оперативная память
3) Жесткий диск
4) SSD

6. Какой язык программирования вы знаете (если знаете)? (2 балла)

_____

7. Опишите, как вы можете организовать последовательность действий для приготовления простого блюда: бутерброда. (10 баллов)

_____

_____

_____

_____

_____

_____

_____

_____

Приложение 2

Вводная диагностика (средний уровень)

(максимум-20 баллов)

Дата_____

ФИО _____

Группа_____

1. Что такое Python?
   - A) Математическая формула
   - B) Язык программирования
   - C) Компьютерная игра
   - D) Тип книги
2. Какой символ используется для начала комментария в Python?
   - A) #
   - B) /
   - C) *
   - D) //
3. Какой из следующих вариантов правильно выводит текст "Привет, мир!" на экран?
   - A) print(Привет, мир!)
   - B) print("Привет, мир!")
   - C) echo("Привет, мир!")
   - D) write("Привет, мир!")
4. Какой тип данных используется для хранения целых чисел в Python?
   - A) str
   - B) int
   - C) float
   - D) list
5. Какой из следующих вариантов проверяет, равны ли переменные a и b?
   - A) a = b
   - B) a == b
   - C) a === b
   - D) a != b
6. Какой цикл используется для повторения блока кода фиксированное количество раз?
   - A) if
   - B) while
   - C) for
7. Напиши программу:

Необходимо ввести с клавиатуры 2 целых числа. Найти среднее арифметическое этих чисел и вывести на экран.

_____
_____
_____
_____
_____
_____
_____

Приложение 3

Промежуточная аттестация (базовый уровень)

(максимум-50 баллов)

Дата_____

ФИО _____

Группа_____

Часть 1: Выбор ответов (20 баллов)

Выберите правильный ответ. Каждый вопрос стоит 4 балла.

1. Какой оператор используется для ввода данных от пользователя в Python?
   - a) print()
   - b) input()
   - c) read()
2. Какой из следующих операторов является логическим?
   - a) +
   - b) and
   - c) *
3. Какой метод строки используется для преобразования всех символов в строке в верхний регистр?
   - a) upper()
   - b) capitalize()
   - c) title()
4. Какой из элементов ниже относится к библиотеке Turtle?
   - a) random()
   - b) forward()
   - c) print()
5. Что делает функция random.randint(a, b)?
   - a) Возвращает случайное число с плавающей запятой
   - b) Возвращает случайное целое число в диапазоне от a до b
   - c) Возвращает случайный элемент из списка

Часть 2: Практическое задание (30 баллов)

Напишите код на Python, который выполняет следующие задачи:

1. Запросите у пользователя два целых числа.
2. Напишите код, который вычисляет их сумму, разность, произведение и частное.
3. Выведите результаты на экран с использованием форматированных строк (f-strings).
4. Используйте библиотеку Random для генерации случайного числа от 1 до 10 и сравните его с первым введенным числом, и выведите большее из этих числел.

Приложение 4

Промежуточная аттестация (2 год обучения)

(максимум-50 баллов)

Дата_____

ФИО _____

Группа_____

Часть 1: Выбор ответов (20 баллов)

Выберите правильный ответ. Каждый вопрос стоит 4 балла.

1. Какова основная цель изучения программирования на Python?
   - a) Создание веб-сайтов
   - b) Автоматизация задач
   - c) Разработка игр
   - d) Все вышеперечисленное

2. Какой метод используется для создания основного окна в Tkinter?
   - a) Tk()
   - b) Frame()
   - c) Canvas()
   - d) Window()

3. Какой метод используется для начала рисования с помощью библиотеки Turtle?
   - a) start()
   - b) begin()
   - c) forward()
   - d) penup()

4. Какой метод используется для добавления элемента в конец списка?
   - a) append()
   - b) add()
   - c) insert()
   - d) extend()

5. Какой метод инициализирует все модули Pygame?
   - a) pygame.start()
   - b) pygame.init()
   - c) pygame.setup()
   - d) pygame.begin()

Часть 2: Практическое задание (30 баллов)

Используя Черепашечью графику нарисовать звездное небо.

Приложение 5

Промежуточная аттестация (3 год обучения)

(максимум-50 баллов)

Дата_____

ФИО _____

Группа_____

Часть 1: Выбор ответов. Выберите правильный ответ.

1. Что такое генератор списков в Python?
A) Способ создания списков с помощью циклов
B) Способ создания списков с помощью функции map
C) Способ создания списков с помощью выражений внутри квадратных скобок
D) Способ создания списков с помощью встроенных функций

2. Какое из следующих утверждений о множестве в Python является верным?
A) Множества могут содержать дубликаты
B) Множества упорядочены
C) Множества поддерживают операции объединения и пересечения

3. Как объявить функцию в Python?
A) function my_function():
B) def my_function():
C) func my_function():
D) define my_function():

4. Какой метод используется для рисования линии в библиотеке Turtle?
A) turtle.draw_line()
B) turtle.line()
C) turtle.forward()
D) turtle.move()

5. Как создать класс в Python?
A) class MyClass()
B) create MyClass:
C) class MyClass:
D) MyClass = class()

Часть 2: Практическое задание

Задание: "Управление библиотекой"

Описание: Напишите консольную программу на Python, которая будет управлять небольшой библиотекой. Программа должна позволять пользователю выполнять следующие операции:

1. Добавить книгу в библиотеку.
2. Удалить книгу из библиотеки.
3. Просмотреть список всех книг.
4. Искать книгу по названию или автору.

Требования:

- Каждая книга должна иметь следующие атрибуты: название, автор, год издания.
- Данные о книгах можно хранить в виде списка словарей.
- Программа должна быть интерактивной и предоставлять меню для выбора операции.
- Обработайте возможные ошибки, например, попытку удалить несуществующую книгу или искать книгу, которой нет в библиотеке.

Приложение 6

Итоговая аттестация (базовый уровень)

(максимум-20 баллов)

Дата_____

ФИО _____

Группа_____

Выбор ответов (20 баллов)

Выберите правильный ответ. Каждый вопрос стоит 2 баллов.

1. Что такое логический тип данных в Python?
    a) Тип, который может принимать только одно значение
    b) Тип, который может принимать два значения: True или False
    c) Тип, который используется для работы с числами
    d) Тип, который используется для работы со строками

2. Какой из следующих операторов используется для выхода из цикла?
    a) continue
    b) break
    c) pass
    d) return

3. Что такое вложенные циклы?
    a) Циклы, которые повторяются несколько раз
    b) Циклы внутри других циклов
    c) Циклы, которые никогда не заканчиваются
    d) Циклы, которые выполняются одновременно

4. Какой из перечисленных операторов является расширенным оператором присваивания?
    a) =
    b) +=
    c) ==
    d) !=

5. Как создается список в Python?
    a) []
    b) {}
    c) ()
    d) <>

6. Какой метод используется для добавления элемента в конец списка?

a) append()
b) add()
c) insert()
d) push()

7. Какой метод используется для рисования простых фигур в Pygame?
   a) draw()
   b) paint()
   c) create()
   d) render()

8. Как обрабатываются события нажатия кнопок клавиатуры в Pygame?
   a) Через функцию input()
   b) Через обработчик событий
   c) Через метод draw()
   d) Через метод update()

9. Как загружается изображение в Pygame?
   a) load_image()
   b) import_image()
   c) pygame.image.load()
   d) pygame.load_image()

10. Какой инструмент использовали для создания дизайн макета?
   a) Photoshop
   b) Figma
   c) Paint
   d) Sketch

Часть 2: Практическое задание (30 баллов)

Задание: «Черепашка рисует фигуры»

Описание:

Напишите программу на Python с использованием библиотеки turtle, которая позволяет пользователю рисовать разные геометрические фигуры. Программа должна позволять пользователю выполнять следующие операции: квадрат, круг, треугольник, очистить экран.

Требования:

Для каждой фигуры нужно написать отдельную функцию.

Продвинутый уровень: Добавить меню выбора фигур.

Приложение 7

Итоговая аттестация (2 год обучения)

(максимум-20 баллов)

Дата_____

ФИО _____

Группа_____

Часть 1: Выбор ответов (20 баллов)

Выберите правильный ответ. Каждый вопрос стоит 4 балла.

1. Что такое генератор списков в Python?
   - A) Способ создания списков с помощью циклов
   - B) Способ создания списков с помощью функции map
   - C) Способ создания списков с помощью выражений внутри квадратных скобок
   - D) Способ создания списков с помощью встроенных функций

2. Какое из следующих утверждений о множестве в Python является верным?
   - A) Множества могут содержать дубликаты
   - B) Множества упорядочены
   - C) Множества поддерживают операции объединения и пересечения

3. Как объявить функцию в Python?
   - A) function my_function():
   - B) def my_function():
   - C) func my_function():
   - D) define my_function():

4. Какой метод используется для рисования линии в библиотеке Turtle?
   - A) turtle.draw_line()
   - B) turtle.line()
   - C) turtle.forward()
   - D) turtle.move()

5. Как создать класс в Python?
   - A) class MyClass()
   - B) create MyClass:
   - C) class MyClass:
   - D) MyClass = class()

Часть 2: Практическое задание (30 баллов)

Задание: "Управление библиотекой"

Описание:

Напишите консольную программу на Python, которая будет управлять небольшой библиотекой. Программа должна позволять пользователю выполнять следующие операции:

1. Добавить книгу в библиотеку.
2. Удалить книгу из библиотеки.
3. Просмотреть список всех книг.
4. Искать книгу по названию или автору.

Требования:

- Каждая книга должна иметь следующие атрибуты: название, автор, год издания.
- Данные о книгах можно хранить в виде списка словарей.
- Программа должна быть интерактивной и предоставлять меню для выбора операции.
- Обработайте возможные ошибки, например, попытку удалить несуществующую книгу или искать книгу, которой нет в библиотеке.

Пример интерфейса:

Добро пожаловать в библиотеку!

1. Добавить книгу
2. Удалить книгу
3. Просмотреть список книг
4. Искать книгу
5. Выход

Выберите операцию (1-5):

Приложение 8

Итоговая аттестация (2 год обучения)

(максимум-20 баллов)

Дата_____

ФИО _____

Группа_____

Часть 1: Выбор ответов (20 баллов)

Выберите правильный ответ. Каждый вопрос стоит 4 балла.

1. Как называется принцип объектно-ориентированного программирования, который позволяет скрывать внутренние данные объекта и предоставлять доступ к ним только через специальные методы?

A) Наследование B) Полиморфизм C) Инкапсуляция

2. Какой из перечисленных инструментов является фреймворком для создания веб-приложений на Python?

A) BeautifulSoup B) Pandas

C) Flask D) Matplotlib

3. Какой формат данных чаще всего используется для обмена информацией между веб-сервисами (API) благодаря своей лёгкости и читаемости?

A) CSV B) JSON

C) SQL D) DOCX

4. Какая библиотека Python традиционно используется для создания графиков, диаграмм и визуализации данных?

A) Requests B) SQLAlchemy

C) Matplotlib D) Django

5. Какая технология позволяет работать с базой данных SQL, используя Python-объекты вместо написания чистых SQL-запросов (пример из вашей программы — SQLAlchemy)?

A) Инкапсуляция B) ORM (Object-Relational Mapping)

C) HTML-разметка D) HTTP-запрос

Часть 2: Практическое задание

Задание: «Управление сотрудниками компании»

Описание: Напишите программу на Python, которая работает с JSON-файлом, содержащим данные о сотрудниках компании. Программа должна читать данные из файла, обрабатывать их и выводить в виде форматированной таблицы. Пользователь может выполнять различные операции с данными.

Программа должна позволять пользователю выполнять следующие операции:

- Вывести всех сотрудников в виде таблицы.
- Найти сотрудников с зарплатой выше указанной суммы.
- Найти сотрудников в указанном отделе.
- Добавить нового сотрудника в файл.
- Удалить сотрудника по ID.
- Выход из программы.

Требования:

Данные хранятся в файле employees.json в следующем формате:

```
  {
      "id": 1,
      "имя": "Иван Петров",
      "отдел": "IT",
      "должность": "Разработчик",
      "зарплата": 120000
    },
```

Если файла нет, программа должна создать его с пустым списком.

Вывод данных должен быть в виде красивой таблицы с границами (использовать форматирование строк или библиотеку tabulate).

Все изменения (добавление, удаление) должны сохраняться в JSON-файл.

Приложение 9

Диагностическая карта

Группа №_____

№	Фамилия Имя	Входная диагностика	Промежуточная диагностика	Итоговая диагностика

Приложение 10

Календарный учебный график

(для 1-ой группы по программе «Программирование на Python», Базовый уровень, 1 год обучения)

Дата	Время проведения занятия	Тема занятия	Содержание	Количество часов		
				Всего	Теория	Практика
09.09.2026	15:50 - 17:30	Вводный урок. Инструктаж по технике безопасности в компьютерном классе. Знакомство с Python. IDLE.	Ознакомление с языком Python, его основными возможностями и особенностями. Установка и настройка среды IDLE.	2	1	1
12.09.2026	14:00-15:40	Команды input() и print().	Изучение функций ввода и вывода данных. Примеры использования команд input() для получения данных от пользователя и print() для вывода информации.	2	1	1
16.09.2026	15:50 - 17:30	Параметры sep, end. Переменные. Комментарии. PEP 8	Обзор параметров функций print() для настройки вывода. Понимание переменных, их назначения и комментариев в коде. Ознакомление с PEP 8 — стандартами стиля кода.	2	1	1
19.09.2026	14:00-15:40	Работа с целыми числами.	Изучение работы с целыми числами, их арифметическими операциями и особенностями. Примеры задач на использование целых чисел.	2	1	1
23.09.2026	15:50 - 17:30	Условный оператор. Логические операции and, or, not	Ознакомление с условными операторами и логическими операциями. Примеры применения логических операций в условиях.	2	1	1
26.09.2026	14:00-15:40	Условный оператор. Логические операции and, or, not	Ознакомление с условными операторами и логическими операциями. Примеры применения логических операций в условиях.	2	1	1
30.09.2026	15:50 - 17:30	Вложенный и каскадный условный оператор	Изучение вложенных и каскадных условий. Как строить сложные логические конструкции для принятия решений в коде.	2	1	1
03.10.2026	14:00-15:40	Повторение пройденных тем. Решение задач.	Консолидация знаний путем решения практических задач, связанных с ранее изученными темами.	2	0	2
07.10.2026	15:50 - 17:30	Повторение пройденных тем. Решение задач.	Консолидация знаний путем решения практических задач, связанных с ранее изученными темами.	2	0	2

10.10.2026	14:00-15:40	Проект Калькулятор	Создание простого калькулятора с использованием изученных функций и операторов. Практика программирования.	2	0	2
14.10.2026	15:50 - 17:30	Типы данных int, float, str. Встроенные функции min(), max(), abs(). Оператор in.	Обзор основных типов данных в Python. Изучение встроенных функций для работы с числами и строками. Применение оператора in для проверки содержимого.	2	1	1
17.10.2026	14:00-15:40	Строковый тип данных: индексация и срезы	Строковый тип данных: индексация и срезы | Изучение работы со строками, индексация символов и создание срезов для извлечения подстрок.	2	1	1
21.10.2026	15:50 - 17:30	Методы строк	Обзор методов для работы со строками, таких как .upper(), .lower(), .find() и других. Примеры их использования.	2	1	1
24.10.2026	14:00-15:40	Повторение пройденных тем. Решение задач.	Практика и повторение изученных тем с помощью решения задач по методам работы со строками.	2	0	2
28.10.2026	15:50 - 17:30	Введение в библиотеки. Библиотека Random. Math.	Ознакомление с концепцией библиотек в Python. Изучение библиотек Random и Math, их функций и применения.	2	1	1
31.10.2026	14:00-15:40	Введение в библиотеки. Библиотека Random. Math.	Ознакомление с концепцией библиотек в Python. Изучение библиотек Random и Math, их функций и применения.	2	1	1
07.11.2026	15:50 - 17:30	Мини проект игра с использованием библиотеки Random	Создание простой игры с использованием библиотеки Random для генерации случайных чисел или событий.	2	0	2
11.11.2026	14:00-15:40	Библиотека Turtle. Черепашечья графика.	Ознакомление с библиотекой Turtle для создания графики. Примеры рисования фигур и работы с графическими примитивами.	2	1	1
14.11.2026	15:50 - 17:30	Библиотека Turtle. Черепашечья графика.	Ознакомление с библиотекой Turtle для создания графики. Примеры рисования фигур и работы с графическими примитивами.	2	1	1
18.11.2026	14:00-15:40	Цикл for. Функция range().	Изучение цикла for и функции range() для итерации по последовательностям. Примеры их применения.	2	1	1
21.11.2026	15:50 - 17:30	Цикл for. Функция range().	Изучение цикла for и функции range() для итерации по последовательностям. Примеры их применения.	2	1	1

25.11.2026	14:00-15:40	Использование циклов в черепашечьей графике	Применение циклов для создания анимации и динамических фигур в библиотеке Turtle.	2	1	1
28.11.2026	15:50 - 17:30	Использование циклов в черепашечьей графике	Применение циклов для создания анимации и динамических фигур в библиотеке Turtle.	2	1	1
02.12.2026	14:00-15:40	Повторение пройденных тем. Решение задач.	Консолидация знаний через практические задачи, связанные с циклами и графикой Turtle.	2	0	2
05.12.2026	15:50 - 17:30	Вывод текста в Turtle, работа с окном ввода.	Изучение вывода текста на экран с помощью Turtle. Работа с окнами ввода для взаимодействия с пользователем.	2	1	1
09.12.2026	14:00-15:40	Анимация в Turtle	Создание анимации с использованием библиотеки Turtle. Примеры движущихся объектов и эффектов.	2	1	1
12.12.2026	15:50 - 17:30	Обработка событий в Turtle	Изучение обработки событий, таких как нажатия клавиш и клики мыши, для управления графическими объектами.	2	1	1
16.12.2026	14:00-15:40	Промежуточная аттестация	Проведение промежуточной аттестации.	2	0	2
19.12.2026	15:50 - 17:30	Задачи с использованием библиотека Random в черепашечьей графике.	Решение задач, комбинирующих использование библиотеки Random и графики Turtle для создания интересных проектов.	2	0	2
23.12.2026	14:00-15:40	Мини проект игра с использованием библиотеки Turtle	Создание простой игры с использованием возможностей библиотеки Turtle для графики и анимации.	2	1	1
26.12.2026	15:50 - 17:30	Мини проект игра с использованием библиотеки Turtle	Создание простой игры с использованием возможностей библиотеки Turtle для графики и анимации.	2	1	1
30.12.2026	14:00-15:40	Итоговый урок по Turtle, Random. Подведение итогов.	Обзор и повторение всех тем, связанных с библиотеками Turtle и Random. Подведение итогов курса.	2	1	1
09.01.2027	14:00-15:40	Урок повторение. Логический тип данных.	Повторение логического типа данных, его особенностей и применения в Python.	2	1	1
13.01.2027	15:50 - 17:30	Цикл с предусловием while	Изучение цикла while, его структуры и применения в программировании. Примеры использования.	2	1	1
16.01.2027	14:00-15:40	Операторы break, continue, else.	Ознакомление с операторами break и continue для управления циклами. Оператор else в контексте циклов.	2	1	1

20.01.2027	15:50 - 17:30	Вложенные циклы	Изучение вложенных циклов, их применения и примеры решения задач с использованием вложенных структур.	2	1	1
23.01.2027	14:00-15:40	Частые сценарии при написании циклов. Расширенные операторы присваивания.	Обзор распространенных сценариев использования циклов. Изучение расширенных операторов присваивания.	2	1	1
27.01.2027	15:50 - 17:30	Частые сценарии при написании циклов. Расширенные операторы присваивания.	Обзор распространенных сценариев использования циклов. Изучение расширенных операторов присваивания.	2	1	1
30.01.2027	14:00-15:40	Повторение пройденных тем. Решение задач.	Решение задач для закрепления знаний, полученных в ходе изучения циклов и операторов.	2	0	2
03.02.2027	15:50 - 17:30	Введение в списки.	Ознакомление с типом данных «список», его структурой и основными операциями.	2	1	1
06.02.2027	14:00-15:40	Основы работы со списками. Методы списков	Изучение методов списков, таких как append(), remove(), sort() и их применение.	2	1	1
10.02.2027	15:50 - 17:30	Методы списков. Списочные выражения	Продолжение изучения методов списков. Введение в списочные выражения и их использование для создания новых списков.	2	1	1
13.02.2027	14:00-15:40	Методы списков. Списочные выражения	Продолжение изучения методов списков. Введение в списочные выражения и их использование для создания новых списков.	2	1	1
17.02.2027	15:50 - 17:30	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	1	1
20.02.2027	14:00-15:40	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	1	1
24.02.2027	15:50 - 17:30	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	0	2
27.02.2027	14:00-15:40	Повторение пройденных тем. Решение задач.	Консолидация знаний через практические задачи, связанные со списками и массивами.	2	0	2
03.03.2027	15:50 - 17:30	Мини проект на массивы и списки	Создание проекта, в котором используются списки и массивы для решения определенной задачи.	2	0	2

06.03.2027	14:00-15:40	Мини проект на массивы и списки	Создание проекта, в котором используются списки и массивы для решения определенной задачи.	2	0	2
10.03.2027	15:50 - 17:30	Введение в библиотеку Pygame. Простые фигуры.	Ознакомление с библиотекой Pygame, ее возможностями. Рисование простых фигур на экране.	2	1	1
13.03.2027	14:00-15:40	Введение в библиотеку Pygame. Простые фигуры.	Ознакомление с библиотекой Pygame, ее возможностями. Рисование простых фигур на экране.	2	1	1
17.03.2027	15:50 - 17:30	Pygame. Обзор методов.	Изучение основных методов библиотеки Pygame, их предназначение и примеры использования.	2	1	1
20.03.2027	14:00-15:40	Pygame. Обзор методов.	Изучение основных методов библиотеки Pygame, их предназначение и примеры использования.	2	1	1
24.03.2027	15:50 - 17:30	Pygame. Обработчик событий на клик кнопок клавиатуры. Движение фигур.	Реализация обработки событий клавиатуры и перемещение фигур по экрану	2	1	1
27.03.2027	14:00-15:40	Pygame. Обработчик событий на клик кнопок клавиатуры. Движение фигур.	Реализация обработки событий клавиатуры и перемещение фигур по экрану	2	1	1
31.03.2027	15:50 - 17:30	Картинки в pygame. Загрузка картинок.	Изучение загрузки изображений в Pygame и их отображение на экране.	2	1	1
03.04.2027	14:00-15:40	Музыка и звуки в pygame	Ознакомление с добавлением звуковых эффектов и музыки в игры Pygame.	2	1	1
07.04.2027	15:50 - 17:30	Pygame. Обработчик событий на клик мыши. Движение фигур.	Реализация обработки событий мыши и движение фигур в зависимости от действий пользователя.	2	1	1
10.04.2027	14:00-15:40	Pygame. Обработчик событий на клик мыши. Движение фигур.	Реализация обработки событий мыши и движение фигур в зависимости от действий пользователя.	2	1	1
14.04.2027	15:50 - 17:30	Создание игрового цикла	Изучение структуры игрового цикла, его элементов и принципов работы.	2	1	1
17.04.2027	14:00-15:40	Начало работы с проектом. Выбор, разбор, анализ проекта	Выбор и анализ проекта, который будет реализован с использованием Pygame.	2	1	1
21.04.2027	15:50 - 17:30	Теория цветов. Цветовая схема.	Изучение теории цветов, цветовых моделей и создание цветовых схем для визуального оформления проекта.	2	1	1
24.04.2027	14:00-15:40	Знакомство с Figma. Создание дизайн макета будущего мини приложения.	Ознакомление с инструментом Figma для создания макетов пользовательских интерфейсов	2	1	1

28.04.2027	15:50 - 17:30	Знакомство с Figma. Создание дизайн макета будущего мини приложения.	Ознакомление с инструментом Figma для создания макетов пользовательских интерфейсов	2	1	1
05.05.2027	14:00-15:40	Реализация дизайна проекта Pygame на основе дизайн макета Figma.	Применение созданного макета в Pygame, реализация визуального оформления игры.	2	1	1
08.05.2027	15:50 - 17:30	Реализация дизайна проекта Pygame на основе дизайн макета Figma.	Применение созданного макета в Pygame, реализация визуального оформления игры.	2	1	1
12.05.2027	15:50 - 17:30	Итоговая самостоятельная работа	Проведение итогового аттестационного теста.	2	0	2
15.05.2027	14:00-15:40	Реализация функционала проекта. Обработка коллизий.	Имплементация логики игры, включая обработку столкновений между объектами.	2	1	1
19.05.2027	15:50 - 17:30	Реализация функционала проекта. Создание уровней и управление состояниями	Создание уровней в игре, управление состояниями (меню, игра, пауза) и переходы между ними.	2	1	1
22.05.2027	14:00-15:40	Реализация функционала проекта. Создание уровней и управление состояниями	Создание уровней в игре, управление состояниями (меню, игра, пауза) и переходы между ними.	2	0	2
26.05.2027	15:50 - 17:30	Сдача проекта. Создание исполняемого файла exe.	Завершение реализации всех функций проекта, проверка работоспособности и отладка. Подготовка проекта к сдаче, создание исполняемого файла (.exe) для удобства запуска игры.	2	1	1
29.05.2027	14:00-15:40	Заключительный урок. Подведение итогов . Ознакомление с планом на следующий год.	Обсуждение итогов обучения, анализ достигнутых результатов и план на будущее обучение.	2	1	1
		Итого		**144**	**57**	**87**

Календарный учебный график

(для 2-ой группы по программе «Программирование на Python», базовый уровень, 1 год обучения)

Дата	**Время проведения занятия**	**Тема занятия**	**Содержание**	**Количество часов**		
				Всего	**Теория**	**Практика**
11.09.2026	15:50 - 17:30	Вводный урок. Инструктаж по технике безопасности в компьютерном классе. Знакомство с Python. IDLE.	Ознакомление с языком Python, его основными возможностями и особенностями. Установка и настройка среды IDLE.	2	1	1

15.09.2026	15:50 - 17:30	Команды input() и print().	Изучение функций ввода и вывода данных. Примеры использования команд input() для получения данных от пользователя и print() для вывода информации.	2	1	1
18.09.2026	15:50 - 17:30	Параметры sep, end. Переменные. Комментарии. PEP 8	Обзор параметров функций print() для настройки вывода. Понимание переменных, их назначения и комментариев в коде. Ознакомление с PEP 8 — стандартами стиля кода.	2	1	1
22.09.2026	15:50 - 17:30	Работа с целыми числами.	Изучение работы с целыми числами, их арифметическими операциями и особенностями. Примеры задач на использование целых чисел.	2	1	1
25.09.2026	15:50 - 17:30	Условный оператор. Логические операции and, or, not	Ознакомление с условными операторами и логическими операциями. Примеры применения логических операций в условиях.	2	1	1
29.09.2026	15:50 - 17:30	Условный оператор. Логические операции and, or, not	Ознакомление с условными операторами и логическими операциями. Примеры применения логических операций в условиях.	2	1	1
02.10.2026	15:50 - 17:30	Вложенный и каскадный условный оператор	Изучение вложенных и каскадных условий. Как строить сложные логические конструкции для принятия решений в коде.	2	1	1
06.10.2026	15:50 - 17:30	Повторение пройденных тем. Решение задач.	Консолидация знаний путем решения практических задач, связанных с ранее изученными темами.	2	0	2
09.10.2026	15:50 - 17:30	Повторение пройденных тем. Решение задач.	Консолидация знаний путем решения практических задач, связанных с ранее изученными темами.	2	0	2
13.10.2026	15:50 - 17:30	Проект Калькулятор	Создание простого калькулятора с использованием изученных функций и операторов. Практика программирования.	2	0	2
16.10.2026	15:50 - 17:30	Типы данных int, float, str. Встроенные функции min(), max(), abs(). Оператор in.	Обзор основных типов данных в Python. Изучение встроенных функций для работы с числами и строками. Применение оператора in для проверки содержимого.	2	1	1
20.10.2026	15:50 - 17:30	Строковый тип данных: индексация и срезы	Строковый тип данных: индексация и срезы | Изучение работы со строками, индексация символов и создание срезов для извлечения подстрок.	2	1	1

23.10.2026	15:50 - 17:30	Методы строк	Обзор методов для работы со строками, таких как .upper(), .lower(), .find() и других. Примеры их использования.	2	1	1
27.10.2026	15:50 - 17:30	Повторение пройденных тем. Решение задач.	Практика и повторение изученных тем с помощью решения задач по методам работы со строками.	2	0	2
30.10.2026	15:50 - 17:30	Введение в библиотеки. Библиотека Random. Math.	Ознакомление с концепцией библиотек в Python. Изучение библиотек Random и Math, их функций и применения.	2	1	1
03.11.2026	15:50 - 17:30	Введение в библиотеки. Библиотека Random. Math.	Ознакомление с концепцией библиотек в Python. Изучение библиотек Random и Math, их функций и применения.	2	1	1
06.11.2026	15:50 - 17:30	Мини проект игра с использованием библиотеки Random	Создание простой игры с использованием библиотеки Random для генерации случайных чисел или событий.	2	0	2
10.11.2026	15:50 - 17:30	Библиотека Turtle. Черепашечья графика.	Ознакомление с библиотекой Turtle для создания графики. Примеры рисования фигур и работы с графическими примитивами.	2	1	1
13.11.2026	15:50 - 17:30	Библиотека Turtle. Черепашечья графика.	Ознакомление с библиотекой Turtle для создания графики. Примеры рисования фигур и работы с графическими примитивами.	2	1	1
17.11.2026	15:50 - 17:30	Цикл for. Функция range().	Изучение цикла for и функции range() для итерации по последовательностям. Примеры их применения.	2	1	1
20.11.2026	15:50 - 17:30	Цикл for. Функция range().	Изучение цикла for и функции range() для итерации по последовательностям. Примеры их применения.	2	1	1
24.11.2026	15:50 - 17:30	Использование циклов в черепашечьей графике	Применение циклов для создания анимации и динамических фигур в библиотеке Turtle.	2	1	1
27.11.2026	15:50 - 17:30	Использование циклов в черепашечьей графике	Применение циклов для создания анимации и динамических фигур в библиотеке Turtle.	2	1	1
01.12.2026	15:50 - 17:30	Повторение пройденных тем. Решение задач.	Консолидация знаний через практические задачи, связанные с циклами и графикой Turtle.	2	0	2
04.12.2026	15:50 - 17:30	Вывод текста в Turtle, работа с окном ввода.	Изучение вывода текста на экран с помощью Turtle. Работа с окнами ввода для взаимодействия с пользователем.	2	1	1

08.12.2026	15:50 - 17:30	Анимация в Turtle	Создание анимации с использованием библиотеки Turtle. Примеры движущихся объектов и эффектов.	2	1	1
11.12.2026	15:50 - 17:30	Промежуточная аттестация	Проведение промежуточной аттестации.	2	0	2
15.12.2026	15:50 - 17:30	Обработка событий в Turtle	Изучение обработки событий, таких как нажатия клавиш и клики мыши, для управления графическими объектами.	2	1	1
18.12.2026	15:50 - 17:30	Задачи с использованием библиотека Random в черепашечьей графике.	Решение задач, комбинирующих использование библиотеки Random и графики Turtle для создания интересных проектов.	2	0	2
22.12.2026	15:50 - 17:30	Мини проект игра с использованием библиотеки Turtle	Создание простой игры с использованием возможностей библиотеки Turtle для графики и анимации.	2	1	1
25.12.2026	15:50 - 17:30	Мини проект игра с использованием библиотеки Turtle	Создание простой игры с использованием возможностей библиотеки Turtle для графики и анимации.	2	1	1
29.12.2026	15:50 - 17:30	Итоговый урок по Turtle, Random. Подведение итогов.	Обзор и повторение всех тем, связанных с библиотеками Turtle и Random. Подведение итогов курса.	2	1	1
12.01.2027	15:50 - 17:30	Урок повторение. Логический тип данных.	Повторение логического типа данных, его особенностей и применения в Python.	2	1	1
15.01.2027	15:50 - 17:30	Цикл с предусловием while	Изучение цикла while, его структуры и применения в программировании. Примеры использования.	2	1	1
19.01.2027	15:50 - 17:30	Операторы break, continue, else.	Ознакомление с операторами break и continue для управления циклами. Оператор else в контексте циклов.	2	1	1
22.01.2027	15:50 - 17:30	Вложенные циклы	Изучение вложенных циклов, их применения и примеры решения задач с использованием вложенных структур.	2	1	1
26.01.2027	15:50 - 17:30	Частые сценарии при написании циклов. Расширенные операторы присваивания.	Обзор распространенных сценариев использования циклов. Изучение расширенных операторов присваивания.	2	1	1
29.01.2027	15:50 - 17:30	Частые сценарии при написании циклов. Расширенные операторы присваивания.	Обзор распространенных сценариев использования циклов. Изучение расширенных операторов присваивания.	2	1	1

02.02.2027	15:50 - 17:30	Повторение пройденных тем. Решение задач.	Решение задач для закрепления знаний, полученных в ходе изучения циклов и операторов.	2	0	2
05.02.2027	15:50 - 17:30	Введение в списки.	Ознакомление с типом данных «список», его структурой и основными операциями.	2	1	1
09.02.2027	15:50 - 17:30	Основы работы со списками. Методы списков	Изучение методов списков, таких как append(), remove(), sort() и их применение.	2	1	1
12.02.2027	15:50 - 17:30	Методы списков. Списочные выражения	Продолжение изучения методов списков. Введение в списочные выражения и их использование для создания новых списков.	2	1	1
16.02.2027	15:50 - 17:30	Методы списков. Списочные выражения	Продолжение изучения методов списков. Введение в списочные выражения и их использование для создания новых списков.	2	1	1
19.02.2027	15:50 - 17:30	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	1	1
26.02.2027	15:50 - 17:30	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	1	1
02.03.2027	15:50 - 17:30	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	0	2
05.03.2027	15:50 - 17:30	Повторение пройденных тем. Решение задач.	Консолидация знаний через практические задачи, связанные со списками и массивами.	2	0	2
09.03.2027	15:50 - 17:30	Мини проект на массивы и списки	Создание проекта, в котором используются списки и массивы для решения определенной задачи.	2	0	2
12.03.2027	15:50 - 17:30	Мини проект на массивы и списки	Создание проекта, в котором используются списки и массивы для решения определенной задачи.	2	0	2
15.03.2027	17:40 - 19:20	Введение в библиотеку Pygame. Простые фигуры.	Ознакомление с библиотекой Pygame, ее возможностями. Рисование простых фигур на экране.	2	1	1
16.03.2027	15:50 - 17:30	Введение в библиотеку Pygame. Простые фигуры.	Ознакомление с библиотекой Pygame, ее возможностями. Рисование простых фигур на экране.	2	1	1
19.03.2027	15:50 - 17:30	Pygame. Обзор методов.	Изучение основных методов библиотеки Pygame, их предназначение и примеры использования.	2	1	1

23.03.2027	15:50 - 17:30	Pygame. Обзор методов.	Изучение основных методов библиотеки Pygame, их предназначение и примеры использования.	2	1	1
26.03.2027	15:50 - 17:30	Pygame. Обработчик событий на клик кнопок клавиатуры. Движение фигур.	Реализация обработки событий клавиатуры и перемещение фигур по экрану	2	1	1
30.03.2027	15:50 - 17:30	Pygame. Обработчик событий на клик кнопок клавиатуры. Движение фигур.	Реализация обработки событий клавиатуры и перемещение фигур по экрану	2	1	1
02.04.2027	15:50 - 17:30	Картинки в pygame. Загрузка картинок.	Изучение загрузки изображений в Pygame и их отображение на экране.	2	1	1
06.04.2027	15:50 - 17:30	Музыка и звуки в pygame	Ознакомление с добавлением звуковых эффектов и музыки в игры Pygame.	2	1	1
09.04.2027	15:50 - 17:30	Pygame. Обработчик событий на клик мыши. Движение фигур.	Реализация обработки событий мыши и движение фигур в зависимости от действий пользователя.	2	1	1
13.04.2027	15:50 - 17:30	Pygame. Обработчик событий на клик мыши. Движение фигур.	Реализация обработки событий мыши и движение фигур в зависимости от действий пользователя.	2	1	1
16.04.2027	15:50 - 17:30	Создание игрового цикла	Изучение структуры игрового цикла, его элементов и принципов работы.	2	1	1
20.04.2027	15:50 - 17:30	Начало работы с проектом. Выбор, разбор, анализ проекта	Выбор и анализ проекта, который будет реализован с использованием Pygame.	2	1	1
23.04.2027	15:50 - 17:30	Теория цветов. Цветовая схема.	Изучение теории цветов, цветовых моделей и создание цветовых схем для визуального оформления проекта.	2	1	1
30.04.2027	15:50 - 17:30	Знакомство с Figma. Создание дизайн макета будущего мини приложения.	Ознакомление с инструментом Figma для создания макетов пользовательских интерфейсов	2	1	1
04.05.2027	15:50 - 17:30	Знакомство с Figma. Создание дизайн макета будущего мини приложения.	Ознакомление с инструментом Figma для создания макетов пользовательских интерфейсов	2	1	1
07.05.2027	15:50 - 17:30	Реализация дизайна проекта Pygame на основе дизайн макета Figma.	Применение созданного макета в Pygame, реализация визуального оформления игры.	2	1	1
11.05.2027	15:50 - 17:30	Реализация дизайна проекта Pygame на основе дизайн макета Figma.	Применение созданного макета в Pygame, реализация визуального оформления игры.	2	1	1
14.05.2027	15:50 - 17:30	Итоговая самостоятельная работа	Проведение итогового аттестационного теста.	2	0	2
18.05.2027	15:50 - 17:30	Реализация функционала проекта. Обработка коллизий.	Имплементация логики игры, включая обработку столкновений между объектами.	2	1	1

21.05.2027	15:50 - 17:30	Реализация функционала проекта. Создание уровней и управление состояниями	Создание уровней в игре, управление состояниями (меню, игра, пауза) и переходы между ними.	2	1	1
24.05.2027	17:40 - 19:20	Реализация функционала проекта. Создание уровней и управление состояниями	Создание уровней в игре, управление состояниями (меню, игра, пауза) и переходы между ними.	2	0	2
25.05.2027	15:50 - 17:30	Сдача проекта. Создание исполняемого файла exe.	Завершение реализации всех функций проекта, проверка работоспособности и отладка. Подготовка проекта к сдаче, создание исполняемого файла (.exe) для удобства запуска игры.	2	1	1
28.05.2027	15:50 - 17:30	Заключительный урок. Подведение итогов . Ознакомление с планом на следующий год.	Обсуждение итогов обучения, анализ достигнутых результатов и план на будущее обучение.	2	1	1
		Итого		**144**	**57**	**87**

Календарный учебный график

(для 3-ой группы по программе «Программирование на Python», базовый уровень, 1 год обучения)

Дата	**Время проведения занятия**	**Тема занятия**	**Содержание**	**Количество часов**		
				Всего	**Теория**	**Практика**
11.09.2026	17:40 - 19:20	Вводный урок. Инструктаж по технике безопасности в компьютерном классе. Знакомство с Python. IDLE.	Ознакомление с языком Python, его основными возможностями и особенностями. Установка и настройка среды IDLE.	2	1	1
15.09.2026	17:40 - 19:20	Команды input() и print().	Изучение функций ввода и вывода данных. Примеры использования команд input() для получения данных от пользователя и print() для вывода информации.	2	1	1
18.09.2026	17:40 - 19:20	Параметры sep, end. Переменные. Комментарии. PEP 8	Обзор параметров функций print() для настройки вывода. Понимание переменных, их назначения и комментариев в коде. Ознакомление с PEP 8 — стандартами стиля кода.	2	1	1
22.09.2026	17:40 - 19:20	Работа с целыми числами.	Изучение работы с целыми числами, их арифметическими операциями и особенностями. Примеры задач на использование целых чисел.	2	1	1
25.09.2026	17:40 - 19:20	Условный оператор. Логические операции and, or, not	Ознакомление с условными операторами и логическими операциями. Примеры применения логических операций в условиях.	2	1	1

29.09.2026	17:40 - 19:20	Условный оператор. Логические операции and, or, not	Ознакомление с условными операторами и логическими операциями. Примеры применения логических операций в условиях.	2	1	1
02.10.2026	17:40 - 19:20	Вложенный и каскадный условный оператор	Изучение вложенных и каскадных условий. Как строить сложные логические конструкции для принятия решений в коде.	2	1	1
06.10.2026	17:40 - 19:20	Повторение пройденных тем. Решение задач.	Консолидация знаний путем решения практических задач, связанных с ранее изученными темами.	2	0	2
09.10.2026	17:40 - 19:20	Повторение пройденных тем. Решение задач.	Консолидация знаний путем решения практических задач, связанных с ранее изученными темами.	2	0	2
13.10.2026	17:40 - 19:20	Проект Калькулятор	Создание простого калькулятора с использованием изученных функций и операторов. Практика программирования.	2	0	2
16.10.2026	17:40 - 19:20	Типы данных int, float, str. Встроенные функции min(), max(), abs(). Оператор in.	Обзор основных типов данных в Python. Изучение встроенных функций для работы с числами и строками. Применение оператора in для проверки содержимого.	2	1	1
20.10.2026	17:40 - 19:20	Строковый тип данных: индексация и срезы	Строковый тип данных: индексация и срезы Изучение работы со строками, индексация символов и создание срезов для извлечения подстрок.	2	1	1
23.10.2026	17:40 - 19:20	Методы строк	Обзор методов для работы со строками, таких как .upper(), .lower(), .find() и других. Примеры их использования.	2	1	1
27.10.2026	17:40 - 19:20	Повторение пройденных тем. Решение задач.	Практика и повторение изученных тем с помощью решения задач по методам работы со строками.	2	0	2
30.10.2026	17:40 - 19:20	Введение в библиотеки. Библиотека Random. Math.	Ознакомление с концепцией библиотек в Python. Изучение библиотек Random и Math, их функций и применения.	2	1	1
03.11.2026	17:40 - 19:20	Введение в библиотеки. Библиотека Random. Math.	Ознакомление с концепцией библиотек в Python. Изучение библиотек Random и Math, их функций и применения.	2	1	1
06.11.2026	17:40 - 19:20	Мини проект игра с использованием библиотеки Random	Создание простой игры с использованием библиотеки Random для генерации случайных чисел или событий.	2	0	2
10.11.2026	17:40 - 19:20	Библиотека Turtle. Черепашечья графика.	Ознакомление с библиотекой Turtle для создания графики. Примеры рисования фигур и работы с графическими примитивами.	2	1	1

13.11.2026	17:40 - 19:20	Библиотека Turtle. Черепашечья графика.	Ознакомление с библиотекой Turtle для создания графики. Примеры рисования фигур и работы с графическими примитивами.	2	1	1
17.11.2026	17:40 - 19:20	Цикл for. Функция range().	Изучение цикла for и функции range() для итерации по последовательностям. Примеры их применения.	2	1	1
20.11.2026	17:40 - 19:20	Цикл for. Функция range().	Изучение цикла for и функции range() для итерации по последовательностям. Примеры их применения.	2	1	1
24.11.2026	17:40 - 19:20	Использование циклов в черепашечьей графике	Применение циклов для создания анимации и динамических фигур в библиотеке Turtle.	2	1	1
27.11.2026	17:40 - 19:20	Использование циклов в черепашечьей графике	Применение циклов для создания анимации и динамических фигур в библиотеке Turtle.	2	1	1
01.12.2026	17:40 - 19:20	Повторение пройденных тем. Решение задач.	Консолидация знаний через практические задачи, связанные с циклами и графикой Turtle.	2	0	2
04.12.2026	17:40 - 19:20	Вывод текста в Turtle, работа с окном ввода.	Изучение вывода текста на экран с помощью Turtle. Работа с окнами ввода для взаимодействия с пользователем.	2	1	1
08.12.2026	17:40 - 19:20	Анимация в Turtle	Создание анимации с использованием библиотеки Turtle. Примеры движущихся объектов и эффектов.	2	1	1
11.12.2026	17:40 - 19:20	Промежуточная аттестация	Проведение промежуточной аттестации.	2	0	2
15.12.2026	17:40 - 19:20	Обработка событий в Turtle	Изучение обработки событий, таких как нажатия клавиш и клики мыши, для управления графическими объектами.	2	1	1
18.12.2026	17:40 - 19:20	Задачи с использованием библиотека Random в черепашечьей графике.	Решение задач, комбинирующих использование библиотеки Random и графики Turtle для создания интересных проектов.	2	0	2
22.12.2026	17:40 - 19:20	Мини проект игра с использованием библиотеки Turtle	Создание простой игры с использованием возможностей библиотеки Turtle для графики и анимации.	2	1	1
25.12.2026	17:40 - 19:20	Мини проект игра с использованием библиотеки Turtle	Создание простой игры с использованием возможностей библиотеки Turtle для графики и анимации.	2	1	1
29.12.2026	17:40 - 19:20	Итоговый урок по Turtle, Random. Подведение итогов.	Обзор и повторение всех тем, связанных с библиотеками Turtle и Random. Подведение итогов курса.	2	1	1
12.01.2027	17:40 - 19:20	Урок повторение. Логический тип данных.	Повторение логического типа данных, его особенностей и применения в Python.	2	1	1

15.01.2027	17:40 - 19:20	Цикл с предусловием while	Изучение цикла while, его структуры и применения в программировании. Примеры использования.	2	1	1
19.01.2027	17:40 - 19:20	Операторы break, continue, else.	Ознакомление с операторами break и continue для управления циклами. Оператор else в контексте циклов.	2	1	1
22.01.2027	17:40 - 19:20	Вложенные циклы	Изучение вложенных циклов, их применения и примеры решения задач с использованием вложенных структур.	2	1	1
26.01.2027	17:40 - 19:20	Частые сценарии при написании циклов. Расширенные операторы присваивания.	Обзор распространенных сценариев использования циклов. Изучение расширенных операторов присваивания.	2	1	1
29.01.2027	17:40 - 19:20	Частые сценарии при написании циклов. Расширенные операторы присваивания.	Обзор распространенных сценариев использования циклов. Изучение расширенных операторов присваивания.	2	1	1
02.02.2027	17:40 - 19:20	Повторение пройденных тем. Решение задач.	Решение задач для закрепления знаний, полученных в ходе изучения циклов и операторов.	2	0	2
05.02.2027	17:40 - 19:20	Введение в списки.	Ознакомление с типом данных «список», его структурой и основными операциями.	2	1	1
09.02.2027	17:40 - 19:20	Основы работы со списками. Методы списков	Изучение методов списков, таких как append(), remove(), sort() и их применение.	2	1	1
12.02.2027	17:40 - 19:20	Методы списков. Списочные выражения	Продолжение изучения методов списков. Введение в списочные выражения и их использование для создания новых списков.	2	1	1
16.02.2027	17:40 - 19:20	Методы списков. Списочные выражения	Продолжение изучения методов списков. Введение в списочные выражения и их использование для создания новых списков.	2	1	1
19.02.2027	17:40 - 19:20	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	1	1
26.02.2027	17:40 - 19:20	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	1	1
02.03.2027	17:40 - 19:20	Введение в Массивы. Решение задач с массивами.	Ознакомление с массивами, их структурой и основными операциями. Решение практических задач	2	0	2
05.03.2027	17:40 - 19:20	Повторение пройденных тем. Решение задач.	Консолидация знаний через практические задачи, связанные со списками и массивами.	2	0	2
09.03.2027	17:40 - 19:20	Мини проект на массивы и списки	Создание проекта, в котором используются списки и массивы для решения определенной задачи.	2	0	2

12.03.2027	17:40 - 19:20	Мини проект на массивы и списки	Создание проекта, в котором используются списки и массивы для решения определенной задачи.	2	0	2
16.03.2027	17:40 - 19:20	Введение в библиотеку Pygame. Простые фигуры.	Ознакомление с библиотекой Pygame, ее возможностями. Рисование простых фигур на экране.	2	1	1
18.03.2027	17:40 - 19:20	Введение в библиотеку Pygame. Простые фигуры.	Ознакомление с библиотекой Pygame, ее возможностями. Рисование простых фигур на экране.	2	1	1
19.03.2027	17:40 - 19:20	Pygame. Обзор методов.	Изучение основных методов библиотеки Pygame, их предназначение и примеры использования.	2	1	1
23.03.2027	17:40 - 19:20	Pygame. Обзор методов.	Изучение основных методов библиотеки Pygame, их предназначение и примеры использования.	2	1	1
26.03.2027	17:40 - 19:20	Pygame. Обработчик событий на клик кнопок клавиатуры. Движение фигур.	Реализация обработки событий клавиатуры и перемещение фигур по экрану	2	1	1
30.03.2027	17:40 - 19:20	Pygame. Обработчик событий на клик кнопок клавиатуры. Движение фигур.	Реализация обработки событий клавиатуры и перемещение фигур по экрану	2	1	1
02.04.2027	17:40 - 19:20	Картинки в pygame. Загрузка картинок.	Изучение загрузки изображений в Pygame и их отображение на экране.	2	1	1
06.04.2027	17:40 - 19:20	Музыка и звуки в pygame	Ознакомление с добавлением звуковых эффектов и музыки в игры Pygame.	2	1	1
09.04.2027	17:40 - 19:20	Pygame. Обработчик событий на клик мыши. Движение фигур.	Реализация обработки событий мыши и движение фигур в зависимости от действий пользователя.	2	1	1
13.04.2027	17:40 - 19:20	Pygame. Обработчик событий на клик мыши. Движение фигур.	Реализация обработки событий мыши и движение фигур в зависимости от действий пользователя.	2	1	1
16.04.2027	17:40 - 19:20	Создание игрового цикла	Изучение структуры игрового цикла, его элементов и принципов работы.	2	1	1
20.04.2027	17:40 - 19:20	Начало работы с проектом. Выбор, разбор, анализ проекта	Выбор и анализ проекта, который будет реализован с использованием Pygame.	2	1	1
23.04.2027	17:40 - 19:20	Теория цветов. Цветовая схема.	Изучение теории цветов, цветовых моделей и создание цветовых схем для визуального оформления проекта.	2	1	1
30.04.2027	17:40 - 19:20	Знакомство с Figma. Создание дизайн макета будущего мини приложения.	Ознакомление с инструментом Figma для создания макетов пользовательских интерфейсов	2	1	1
04.05.2027	17:40 - 19:20	Знакомство с Figma. Создание дизайн макета будущего мини приложения.	Ознакомление с инструментом Figma для создания макетов пользовательских интерфейсов	2	1	1

07.05.2027	17:40 - 19:20	Реализация дизайна проекта Pygame на основе дизайн макета Figma.	Применение созданного макета в Pygame, реализация визуального оформления игры.	2	1	1
11.05.2027	17:40 - 19:20	Реализация дизайна проекта Pygame на основе дизайн макета Figma.	Применение созданного макета в Pygame, реализация визуального оформления игры.	2	1	1
14.05.2027	17:40 - 19:20	Итоговая самостоятельная работа работа	Проведение итогового аттестационного теста.	2	0	2
17.05.2027	17:40 - 19:20	Реализация функционала проекта. Обработка коллизий.	Имплементация логики игры, включая обработку столкновений между объектами.	2	1	1
18.05.2027	17:40 - 19:20	Реализация функционала проекта. Создание уровней и управление состояниями	Создание уровней в игре, управление состояниями (меню, игра, пауза) и переходы между ними.	2	1	1
21.05.2027	17:40 - 19:20	Реализация функционала проекта. Создание уровней и управление состояниями	Создание уровней в игре, управление состояниями (меню, игра, пауза) и переходы между ними.	2	0	2
25.05.2027	17:40 - 19:20	Сдача проекта. Создание исполняемого файла exe.	Завершение реализации всех функций проекта, проверка работоспособности и отладка. Подготовка проекта к сдаче, создание исполняемого файла (.exe) для удобства запуска игры.	2	1	1
28.05.2027	17:40 - 19:20	Заключительный урок. Подведение итогов . Ознакомление с планом на следующий год.	Обсуждение итогов обучения, анализ достигнутых результатов и план на будущее обучение.	2	1	1
		Итого:		**144**	**57**	**87**

Календарный учебный график

(для 4-ой группы по программе «Программирование на Python», средний уровень, 2 год обучения)

Дата	**Время проведения занятия**	**Тема занятия**	**Содержание**	**Количество часов**		
План				**Всего**	**Теория**	**Практика**
10.09.2026	09:00 - 10:40	Вводный урок	Инструктаж по технике безопасности в компьютерном классе. Основные понятия и терминология. Инструменты и среды разработки.	2	2	0
14.09.2026	09:00 - 10:40	Повторение основных типов, циклов, условий	Синтаксис и структура кода. Переменные и типы данных. Условные операторы и циклы. Решение задач на логику	2	1	1
17.09.2026	09:00 - 10:40	Повторение основных типов, циклов, условий	Понятие алгоритмов и блок-схем. Решение задач популярных алгоритмов	2	1	1
21.09.2026	09:00 - 10:40	Библиотека Tkinter	Основы работы в tkinter.	2	1	1
24.09.2026	09:00 - 10:40	Библиотека Tkinter	Реализация калькулятора на Tkinter	2	1	1

28.09.2026	09:00 - 10:40	Библиотека Tkinter	Реализация калькулятора на Tkinter	2	0	2
01.10.2026	09:00 - 10:40	Введение в библиотеку Turtle	Основные методы библиотеки. Рисование простых фигур. Ввод вывод текста.	2	1	1
05.10.2026	09:00 - 10:40	Введение в библиотеку Turtle	Рисование простой картины	2	1	1
08.10.2026	09:00 - 10:40	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
12.10.2026	09:00 - 10:40	Списки	Основы работы со списками. Итерация списков. Индексы и срезы	2	1	1
15.10.2026	09:00 - 10:40	Списки	Методы работы со списками	2	1	1
19.10.2026	09:00 - 10:40	Списки	Вложенные списки. Задачи на списки.	2	0	2
22.10.2026	09:00 - 10:40	Аналог игры "Виселица" на Turtle	Проектирование игры. Например, вместо виселицы звезда или снежинка.	2	1	1
26.10.2026	09:00 - 10:40	Аналог игры "Виселица" на Turtle	Реализация игры	2	1	1
29.10.2026	09:00 - 10:40	Аналог игры "Виселица" на Turtle	Реализация игры	2	0	2
02.11.2026	09:00 - 10:40	Кортежи	Понятие кортежей. Методы кортежей и применение кортежей.	2	1	1
05.11.2026	09:00 - 10:40	Кортежи	Примеры задач на кортежи	2	1	1
09.11.2026	09:00 - 10:40	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
12.11.2026	09:00 - 10:40	Словари	Основы словарей, методы.	2	1	1
16.11.2026	09:00 - 10:40	Словари	Примеры задач на словари	2	1	1
19.11.2026	09:00 - 10:40	Повторение пройденного материала	Интеграция тем. Повторение основных моментов. Тестирование знаний по пройденным темам.	2	1	1
23.11.2026	09:00 - 10:40	Понятие JSON	Структура JSON. Работа с JSON в Python (модуль `json`). Примеры использования JSON для передачи данных	2	1	1
26.11.2026	09:00 - 10:40	Pygame: Продвинутые возможности	Работа с изображениями и звуками.	2	1	1
30.11.2026	09:00 - 10:40	Pygame: Продвинутые возможности	Спрайты, анимация	2	0	2
03.12.2026	09:00 - 10:40	Pygame: Продвинутые возможности	Спрайты, анимация	2	1	1
07.12.2026	09:00 - 10:40	Разработка игры "Змейка"	Проектирование игры.	2	1	1
10.12.2026	09:00 - 10:40	Промежуточная аттестация	Проведение промежуточной аттестационной работы	2	0	2
14.12.2026	09:00 - 10:40	Разработка игры "Змейка"	Реализация игрового процесса.	2	1	1
17.12.2026	09:00 - 10:40	Разработка игры "Змейка"	Реализация игрового процесса.	2	0	2

21.12.2026	09:00 - 10:40	Разработка игры "Змейка"	Добавление уровней и сложности.	2	0	2
24.12.2026	09:00 - 10:40	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
28.12.2026	09:00 - 10:40	Повторение пройденного материала	Интеграция тем. Повторение основных моментов. Решение задач.	2	1	1
11.01.2027	09:00 - 10:40	Генератор списков, словарей	Определение генераторов. Применение. Преимущества использования генераторов. Практические примеры.	2	1	1
14.01.2027	09:00 - 10:40	Множества	Теория множест в математике и в пайтон.	2	1	1
18.01.2027	09:00 - 10:40	Множества	Методы множеств. Задачи на множества	2	1	1
21.01.2027	09:00 - 10:40	Функции	Определение и вызов функций. Возврат значений	2	1	1
25.01.2027	09:00 - 10:40	Функции	Аргументы и параметры.	2	1	1
28.01.2027	09:00 - 10:40	Функции	Решение задач с использованием функций	2	0	2
01.02.2027	09:00 - 10:40	Лямбда функции	Определение и использование.	2	1	1
04.02.2027	09:00 - 10:40	Лямбда функции	Примеры применения лямбда функций в Python	2	0	2
08.02.2027	09:00 - 10:40	Библиотека Turtle	Установка и настройка. Основные команды Turtle.	2	1	1
11.02.2027	09:00 - 10:40	Библиотека Turtle	Создание графических картинок из простых фигур	2	1	1
15.02.2027	09:00 - 10:40	Библиотека Turtle	Создание сложных рисунков, используя циклы, функции. Выгрузка картинки. Импорт в Фигму.	2	1	1
18.02.2027	09:00 - 10:40	Библиотека Turtle	Создание сложных рисунков, используя циклы, функции. Выгрузка картинки. Импорт в Фигму.	2	1	1
22.02.2027	09:00 - 10:40	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
25.02.2027	09:00 - 10:40	Библиотека Matplotlib	Установка и основы работы. Построение графиков и визуализация данных. Примеры создания различных типов графиков	2	1	1
01.03.2027	09:00 - 10:40	Библиотека Matplotlib	Мини приложение, рисующее график нескольких функций	2	1	1
04.03.2027	09:00 - 10:40	Повторение пройденного материала	Интеграция тем. Повторение основных моментов. Тестирование знаний по пройденным темам.	2	1	1
11.03.2027	09:00 - 10:40	Основы объектно-ориентированного программирования. Создание классов и объектов.	Основы объектно-ориентированного программирования. Создание классов и объектов.	2	1	1
15.03.2027	09:00 - 10:40	Наследование и полиморфизм	Наследование и полиморфизм	2	1	1
18.03.2027	09:00 - 10:40	Наследование и полиморфизм	Наследование и полиморфизм	2	0	2

22.03.2027	09:00 - 10:40	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
25.03.2027	09:00 - 10:40	Разработка телеграмм бота	Основы работы с API Telegram. Использование библиотеки `python-telegram-bot`. Примеры создания простых ботов	2	1	1
29.03.2027	09:00 - 10:40	Разработка телеграмм бота	Разработка собственного бота. Добавление дополнительных функций в телеграмм бот.	2	1	1
01.04.2027	09:00 - 10:40	Разработка телеграмм бота	Расширение функционала бота	2	0	2
05.04.2027	09:00 - 10:40	Разработка телеграмм бота	Использование меню в боте	2	0	2
08.04.2027	09:00 - 10:40	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
12.04.2027	09:00 - 10:40	Компьютерное зрение	Введение в компьютерное зрение. Использование библиотеки OpenCV. Примеры обработки изображений	2	1	1
15.04.2027	09:00 - 10:40	Компьютерное зрение	Обнаружение лиц. Подключение модели. Примеры обработки видео.	2	1	1
19.04.2027	09:00 - 10:40	OpenAI и ChatGPT	Основы работы с API OpenAI. Примеры взаимодействия с ChatGPT. Применение в проектах и разработках	2	1	1
22.04.2027	09:00 - 10:40	OpenAI и ChatGPT	Подключение в телеграм бот	2	1	1
26.04.2027	09:00 - 10:40	Повторение пройденного материала	Интеграция тем. Повторение основных моментов. Решение задач.	2	1	1
29.04.2027	09:00 - 10:40	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
03.05.2027	09:00 - 10:40	Фигма. Плагины.	Теория цвета. Плагины. Наложение эффектов.	2	1	1
06.05.2027	09:00 - 10:40	Фигма. Плагины.	Дизайн игры Крестики-Нолики	2	1	1
10.05.2027	09:00 - 10:40	Разработка игры "Крестики-нолики"	Проектирование игры.	2	1	1
13.05.2027	09:00 - 10:40	Разработка игры "Крестики-нолики"	Реализация логики и интерфейса	2	1	1
17.05.2027	09:00 - 10:40	Итоговая аттестация	Проведение итоговой аттестационной работы	2	0	2
20.05.2027	09:00 - 10:40	Разработка игры "Крестики-нолики"	Возможные улучшения и расширения	2	0	2
24.05.2027	09:00 - 10:40	Разработка игры "Крестики-нолики"	Защита проекта "Крестики-нолики"	2	0	2
27.05.2027	09:00 - 10:40	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
31.05.2027	09:00 - 10:40	Заключительный урок	Обзор пройденного материала Рекомендации по дальнейшему обучению	2	2	0
			Итого	144	59	85

Календарный учебный график

(для 5-ой группы по программе «Программирование на Python», средний уровень, 2 год обучения)

Дата	Время проведения занятия	Тема занятия	Содержание	Количество часов		
План				Всего	Теория	Практика
09.09.2026	17:40 - 19:25	Вводный урок	Инструктаж по технике безопасности в компьютерном классе. Основные понятия и терминология. Инструменты и среды разработки.	2	2	0
12.09.2026	15:50 - 17:30	Повторение основных типов, циклов, условий	Синтаксис и структура кода. Переменные и типы данных. Условные операторы и циклы. Решение задач на логику	2	1	1
16.09.2026	17:40 - 19:25	Повторение основных типов, циклов, условий	Понятие алгоритмов и блок-схем. Решение задач популярных алгоритмов	2	1	1
19.09.2026	15:50 - 17:30	Библиотека Tkinter	Основы работы в tkinter.	2	1	1
23.09.2026	17:40 - 19:25	Библиотека Tkinter	Реализация калькулятора на Tkinter	2	1	1
26.09.2026	15:50 - 17:30	Библиотека Tkinter	Реализация калькулятора на Tkinter	2	0	2
30.09.2026	17:40 - 19:25	Введение в библиотеку Turtle	Основные методы библиотеки. Рисование простых фигур. Ввод вывод текста.	2	1	1
03.10.2026	15:50 - 17:30	Введение в библиотеку Turtle	Рисование простой картины	2	1	1
07.10.2026	17:40 - 19:25	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
10.10.2026	15:50 - 17:30	Списки	Основы работы со списками. Итерация списков. Индексы и срезы	2	1	1
14.10.2026	17:40 - 19:25	Списки	Методы работы со списками	2	1	1
17.10.2026	15:50 - 17:30	Списки	Вложенные списки. Задачи на списки.	2	0	2
21.10.2026	17:40 - 19:25	Аналог игры "Виселица" на Turtle	Проектирование игры. Например, вместо виселицы звезда или снежинка.	2	1	1
24.10.2026	15:50 - 17:30	Аналог игры "Виселица" на Turtle	Реализация игры	2	1	1
28.10.2026	17:40 - 19:25	Аналог игры "Виселица" на Turtle	Реализация игры	2	0	2
31.10.2026	15:50 - 17:30	Кортежи	Понятие кортежей. Методы кортежей и применение кортежей.	2	1	1
07.11.2026	17:40 - 19:25	Кортежи	Примеры задач на кортежи	2	1	1
11.11.2026	15:50 - 17:30	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
14.11.2026	17:40 - 19:25	Словари	Основы словарей, методы.	2	1	1
18.11.2026	15:50 - 17:30	Словари	Примеры задач на словари	2	1	1
21.11.2026	17:40 - 19:25	Повторение пройденного материала	Интеграция тем. Повторение основных моментов. Тестирование знаний по пройденным темам.	2	1	1

25.11.2026	15:50 - 17:30	Понятие JSON	Структура JSON. Работа с JSON в Python (модуль `json`). Примеры использования JSON для передачи данных	2	1	1
28.11.2026	17:40 - 19:25	Pygame: Продвинутые возможности	Работа с изображениями и звуками.	2	1	1
02.12.2026	15:50 - 17:30	Pygame: Продвинутые возможности	Спрайты, анимация	2	0	2
05.12.2026	17:40 - 19:25	Pygame: Продвинутые возможности	Спрайты, анимация	2	1	1
09.12.2026	15:50 - 17:30	Разработка игры "Змейка"	Проектирование игры.	2	1	1
12.12.2026	17:40 - 19:25	Промежуточная аттестация	Проведение промежуточной аттестационной работы	2	0	2
16.12.2026	15:50 - 17:30	Разработка игры "Змейка"	Реализация игрового процесса.	2	1	1
19.12.2026	17:40 - 19:25	Разработка игры "Змейка"	Реализация игрового процесса.	2	0	2
23.12.2026	15:50 - 17:30	Разработка игры "Змейка"	Добавление уровней и сложности.	2	0	2
26.12.2026	17:40 - 19:25	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
30.12.2026	15:50 - 17:30	Повторение пройденного материала	Интеграция тем. Повторение основных моментов. Решение задач.	2	1	1
09.01.2027	15:50 - 17:30	Генератор списков, словарей	Определение генераторов. Применение. Преимущества использования генераторов. Практические примеры.	2	1	1
13.01.2027	17:40 - 19:25	Множества	Теория множест в математике и в пайтон.	2	1	1
16.01.2027	15:50 - 17:30	Множества	Методы множеств. Задачи на множества	2	1	1
20.01.2027	17:40 - 19:25	Функции	Определение и вызов функций. Возврат значений	2	1	1
23.01.2027	15:50 - 17:30	Функции	Аргументы и параметры.	2	1	1
27.01.2027	17:40 - 19:25	Функции	Решение задач с использованием функций	2	0	2
30.01.2027	15:50 - 17:30	Лямбда функции	Определение и использование.	2	1	1
03.02.2027	17:40 - 19:25	Лямбда функции	Примеры применения лямбда функций в Python	2	0	2
06.02.2027	15:50 - 17:30	Библиотека Turtle	Установка и настройка. Основные команды Turtle.	2	1	1
10.02.2027	17:40 - 19:25	Библиотека Turtle	Создание графических картинок из простых фигур	2	1	1
13.02.2027	15:50 - 17:30	Библиотека Turtle	Создание сложных рисунков, используя циклы, функции. Выгрузка картинки. Импорт в Фигму.	2	1	1
17.02.2027	17:40 - 19:25	Библиотека Turtle	Создание сложных рисунков, используя циклы, функции. Выгрузка картинки. Импорт в Фигму.	2	1	1

20.02.2027	15:50 - 17:30	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
24.02.2027	17:40 - 19:25	Библиотека Matplotlib	Установка и основы работы. Построение графиков и визуализация данных. Примеры создания различных типов графиков	2	1	1
27.02.2027	15:50 - 17:30	Библиотека Matplotlib	Мини приложение, рисующее график нескольких функций	2	1	1
03.03.2027	17:40 - 19:25	Повторение пройденного материала	Интеграция тем. Повторение основных моментов. Тестирование знаний по пройденным темам.	2	1	1
06.03.2027	15:50 - 17:30	Основы объектно-ориентированного программирования. Создание классов и объектов.	Основы объектно-ориентированного программирования. Создание классов и объектов.	2	1	1
10.03.2027	17:40 - 19:25	Наследование и полиморфизм	Наследование и полиморфизм	2	1	1
13.03.2027	15:50 - 17:30	Наследование и полиморфизм	Наследование и полиморфизм	2	0	2
17.03.2027	17:40 - 19:25	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
20.03.2027	15:50 - 17:30	Разработка телеграмм бота	Основы работы с API Telegram. Использование библиотеки `python-telegram-bot`. Примеры создания простых ботов	2	1	1
24.03.2027	17:40 - 19:25	Разработка телеграмм бота	Разработка собственного бота. Добавление дополнительных функций в телеграмм бот.	2	1	1
27.03.2027	15:50 - 17:30	Разработка телеграмм бота	Расширение функционала бота	2	0	2
31.03.2027	17:40 - 19:25	Разработка телеграмм бота	Использование меню в боте	2	0	2
03.04.2027	15:50 - 17:30	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
07.04.2027	17:40 - 19:25	Компьютерное зрение	Введение в компьютерное зрение. Использование библиотеки OpenCV. Примеры обработки изображений	2	1	1
10.04.2027	15:50 - 17:30	Компьютерное зрение	Обнаружение лиц. Подключение модели. Примеры обработки видео.	2	1	1
14.04.2027	17:40 - 19:25	OpenAI и ChatGPT	Основы работы с API OpenAI. Примеры взаимодействия с ChatGPT. Применение в проектах и разработках	2	1	1
17.04.2027	15:50 - 17:30	OpenAI и ChatGPT	Подключение в телеграм бот	2	1	1
21.04.2027	17:40 - 19:25	Повторение пройденного материала	Интеграция тем. Повторение основных моментов. Решение задач.	2	1	1
24.04.2027	15:50 - 17:30	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1

28.04.2027	17:40 - 19:25	Фигма. Плагины.	Теория цвета. Плагины. Наложение эффектов.	2	1	1
05.05.2027	15:50 - 17:30	Фигма. Плагины.	Дизайн игры Крестики-Нолики	2	1	1
08.05.2027	17:40 - 19:25	Разработка игры "Крестики-нолики"	Проектирование игры.	2	1	1
12.05.2027	17:40 - 19:25	Разработка игры "Крестики-нолики"	Реализация логики и интерфейса	2	1	1
15.05.2027	15:50 - 17:30	Итоговая аттестация	Проведение итоговой аттестационной работы	2	0	2
19.05.2027	17:40 - 19:25	Разработка игры "Крестики-нолики"	Возможные улучшения и расширения	2	0	2
22.05.2027	15:50 - 17:30	Разработка игры "Крестики-нолики"	Защита проекта "Крестики-нолики"	2	0	2
26.05.2027	17:40 - 19:25	Решение задач на логику	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
29.05.2027	15:50 - 17:30	Заключительный урок	Обзор пройденного материала Рекомендации по дальнейшему обучению	2	2	0
			Итого	144	59	85

Календарный учебный график

(для 6-ой группы по программе «Программирование на Python», средний уровень, 3 год обучения)

Дата	**Время проведения занятия**	**Тема занятия**	**Содержание**	**Количество часов**		
План				**Всего**	**Теория**	**Практика**
10.09.2026	15:50 - 17:30	Вводный урок	Инструктаж по технике безопасности в компьютерном классе. Основные понятия и терминология. Инструменты и среды разработки.	2	2	0
14.09.2026	15:50 - 17:30	Типы данных: int, float, str, bool	Типы данных: int, float, str, bool	2	1	1
17.09.2026	15:50 - 17:30	Структуры данных: списки, кортежи	Структуры данных: списки, кортежи	2	1	1
21.09.2026	15:50 - 17:30	Словари и множества	Словари и множества	2	1	1
24.09.2026	15:50 - 17:30	Словари и множества	Словари и множества	2	1	1
28.09.2026	15:50 - 17:30	Словари и множества	Словари и множества	2	0	2
01.10.2026	15:50 - 17:30	Условные конструкции и циклы	Условные конструкции и циклы	2	1	1
05.10.2026	15:50 - 17:30	Условные конструкции и циклы	Условные конструкции и циклы	2	1	1
08.10.2026	15:50 - 17:30	Условные конструкции и циклы	Условные конструкции и циклы	2	0	2
12.10.2026	15:50 - 17:30	Списки	Основы работы со списками. Итерация списков. Индексы и срезы	2	1	1
15.10.2026	15:50 - 17:30	Списки	Методы работы со списками	2	1	1
19.10.2026	15:50 - 17:30	Списки	Вложенные списки. Задачи на списки.	2	0	2
22.10.2026	15:50 - 17:30	Классы, объекты, наследование	Классы, объекты, наследование	2	1	1
26.10.2026	15:50 - 17:30	Классы, объекты, наследование	Классы, объекты, наследование	2	1	1

29.10.2026	15:50 - 17:30	Классы, объекты, наследование	Классы, объекты, наследование	2	0	2
02.11.2026	15:50 - 17:30	Классы, объекты, наследование	Классы, объекты, наследование	2	1	1
05.11.2026	15:50 - 17:30	Классы, объекты, наследование	Классы, объекты, наследование	2	1	1
09.11.2026	15:50 - 17:30	Полиморфизм, инкапсуляция	Полиморфизм, инкапсуляция	2	1	1
12.11.2026	15:50 - 17:30	Полиморфизм, инкапсуляция	Полиморфизм, инкапсуляция	2	1	1
16.11.2026	15:50 - 17:30	Полиморфизм, инкапсуляция	Полиморфизм, инкапсуляция	2	0	2
19.11.2026	15:50 - 17:30	Полиморфизм, инкапсуляция	Полиморфизм, инкапсуляция	2	0	2
23.11.2026	15:50 - 17:30	Практика: создание игры с ООП	Практика: создание игры с ООП	2	1	1
26.11.2026	15:50 - 17:30	Практика: создание игры с ООП	Практика: создание игры с ООП	2	1	1
30.11.2026	15:50 - 17:30	Практика: создание игры с ООП	Практика: создание игры с ООП	2	0	2
03.12.2026	15:50 - 17:30	Практика: создание игры с ООП	Практика: создание игры с ООП	2	0	2
07.12.2026	15:50 - 17:30	Промежуточная аттестация	Промежуточная аттестация	2	0	2
10.12.2026	15:50 - 17:30	Структура HTML, CSS (базово)	Структура HTML, CSS (базово)	2	1	1
14.12.2026	15:50 - 17:30	Адаптивная верстка (основы)	Реализация игрового процесса.	2	1	1
17.12.2026	15:50 - 17:30	Понятие деплоя, тонкости и методы	Реализация игрового процесса.	2	1	1
21.12.2026	15:50 - 17:30	Структура HTML, CSS (базово)	Структура HTML, CSS (базово)	2	1	1
24.12.2026	15:50 - 17:30	Основы веб-приложений	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
28.12.2026	15:50 - 17:30	Flask/Django: простое веб-приложение	Flask/Django: простое веб-приложение	2	1	1
11.01.2027	15:50 - 17:30	Flask/Django: простое веб-приложение	Flask/Django: простое веб-приложение	2	1	1
14.01.2027	15:50 - 17:30	Flask/Django: простое веб-приложение	Flask/Django: простое веб-приложение	2	0	2
18.01.2027	15:50 - 17:30	Flask/Django: простое веб-приложение	Flask/Django: простое веб-приложение	2	0	2
21.01.2027	15:50 - 17:30	Валидация, семантические теги	Валидация, семантические теги	2	1	1
25.01.2027	15:50 - 17:30	Валидация, семантические теги	Валидация, семантические теги	2	1	1
28.01.2027	15:50 - 17:30	Проект на веб-приложение	Проект на веб-приложение	2	1	1
01.02.2027	15:50 - 17:30	Проект на веб-приложение	Проект на веб-приложение	2	1	1
04.02.2027	15:50 - 17:30	Проект на веб-приложение	Проект на веб-приложение	2	0	2
08.02.2027	15:50 - 17:30	Библиотеки Requests, BeautifulSoup	Библиотеки Requests, BeautifulSoup	2	1	1
11.02.2027	15:50 - 17:30	Библиотеки Requests, BeautifulSoup	Библиотеки Requests, BeautifulSoup	2	1	1
15.02.2027	15:50 - 17:30	Библиотеки Requests, BeautifulSoup	Библиотеки Requests, BeautifulSoup	2	1	1

18.02.2027	15:50 - 17:30	Библиотеки Requests, BeautifulSoup	Библиотеки Requests, BeautifulSoup	2	1	1
22.02.2027	15:50 - 17:30	Библиотеки Requests, BeautifulSoup	Библиотеки Requests, BeautifulSoup	2	0	2
25.02.2027	15:50 - 17:30	Работа с API (JSON)	Работа с API (JSON)	2	1	1
01.03.2027	15:50 - 17:30	Работа с API (JSON)	Работа с API (JSON)	2	1	1
04.03.2027	15:50 - 17:30	Работа с API (JSON)	Работа с API (JSON)	2	0	2
11.03.2027	15:50 - 17:30	Pandas: обработка таблиц	Pandas: обработка таблиц	2	1	1
15.03.2027	15:50 - 17:30	Pandas: обработка таблиц	Pandas: обработка таблиц	2	1	1
18.03.2027	15:50 - 17:30	Pandas: обработка таблиц	Pandas: обработка таблиц	2	1	1
22.03.2027	15:50 - 17:30	Pandas: обработка таблиц	Pandas: обработка таблиц	2	1	1
25.03.2027	15:50 - 17:30	Pandas: обработка таблиц	Pandas: обработка таблиц	2	1	1
29.03.2027	15:50 - 17:30	Pandas: обработка таблиц	Pandas: обработка таблиц	2	1	1
01.04.2027	15:50 - 17:30	Pandas: обработка таблиц	Pandas: обработка таблиц	2	0	2
05.04.2027	15:50 - 17:30	Визуализация (Matplotlib/Seaborn)	Визуализация (Matplotlib/Seaborn)	2	1	1
08.04.2027	15:50 - 17:30	Визуализация (Matplotlib/Seaborn)	Визуализация (Matplotlib/Seaborn)	2	1	1
12.04.2027	15:50 - 17:30	Визуализация (Matplotlib/Seaborn)	Визуализация (Matplotlib/Seaborn)	2	0	2
15.04.2027	15:50 - 17:30	Подготовка отчета по данным	Подготовка отчета по данным	2	1	1
19.04.2027	15:50 - 17:30	Подготовка отчета по данным	Подготовка отчета по данным	2	1	1
22.04.2027	15:50 - 17:30	Подготовка отчета по данным	Подготовка отчета по данным	2	0	2
26.04.2027	15:50 - 17:30	CSV, JSON, SQLite	CSV, JSON, SQLite	2	1	1
29.04.2027	15:50 - 17:30	CSV, JSON, SQLite	CSV, JSON, SQLite	2	1	1
03.05.2027	15:50 - 17:30	CSV, JSON, SQLite	CSV, JSON, SQLite	2	0	2
06.05.2027	15:50 - 17:30	ORM (SQLAlchemy)	ORM (SQLAlchemy)	2	1	1
10.05.2027	15:50 - 17:30	ORM (SQLAlchemy)	ORM (SQLAlchemy)	2	1	1
13.05.2027	15:50 - 17:30	ORM (SQLAlchemy)	ORM (SQLAlchemy)	2	0	2
17.05.2027	15:50 - 17:30	Итоговая аттестация	Проведение итоговой аттестационной работы	2	0	2
20.05.2027	15:50 - 17:30	Определение идеи, создание дизайна проекта	Определение идеи, создание дизайна проекта	2	0	2
24.05.2027	15:50 - 17:30	Реализация проекта	Защита проекта "Крестики-нолики"	2	0	2
27.05.2027	15:50 - 17:30	Оформление проекта и защита	Решение задач на развитии логического мышления и навыков решения задач, требующих аналитического подхода	2	1	1
31.05.2027	15:50 - 17:30	Заключительный урок	Обзор пройденного материала	2	2	0
				144	52	92

Приложение 9

Рабочая программа воспитания
и календарный план воспитательной работы студии
«Программирование на Python» на 2026-2027 учебный год

1. Цель, задачи и результат воспитательной работы

Цель воспитания: Основной целью является создание условий для полноценного развития каждого обучающегося, включая его физическое, интеллектуальное, эмоциональное и социальное благополучие.

Задачи воспитания:

- Организовать воспитательную работу с учётом развития самостоятельных возможностей обучающихся на данном этапе.
- Способствовать формированию сплочённого коллектива
- Формировать качества каждого ребёнка в зависимости от личных особенностей, интересов, склонностей, состояния здоровья, возрастных особенностей, характера и психики ребёнка

Результат воспитания:

- Сформированные ценности: осознание и придерживание нравственных, социальных и культурных ценностей.
- Личностное развитие: Ученики демонстрируют развитие самостоятельности, самоуважения и лидерских качеств.
- Физическое благополучие: Ученики поддерживают физическое здоровье и ведут активный образ жизни.
- Успешная учеба: Ученики достигают хороших результатов в учебе и развивают навыки самоорганизации и учебной деятельности.
- Эмоциональное благополучие: Ученики умеют справляться с эмоциональными трудностями и стрессом.
- Самореализация: Ученики находят себя в различных сферах деятельности и проявляют свой потенциал.
- Участие в общественной жизни: Ученики активно участвуют в мероприятиях и проектах школы и сообщества.

2. Планируемые результаты реализации программы воспитания

Работа с коллективом обучающихся

Работа с коллективом обучающихся детского объединения нацелена на:

- развитие практических навыков в организации органов самоуправления, а также изучение основ этики и психологии общения, технологий социального и творческого проектирования;
- обучение организаторским умениям и навыкам самоорганизации, а также формирование чувства ответственности как за собственные действия, так и за действия окружающих;
- стимулирование творческого, культурного и коммуникативного потенциала участников через вовлечение в совместную общественно-полезную деятельность;
- поддержку формирования активной гражданской позиции и сознательного участия в жизни общества;
- воспитание уважительного отношения к труду, окружающей природе и своему городу;
- развитие навыков командной работы и сотрудничества, что способствует улучшению взаимодействия между участниками;
- формирование критического мышления и способности к принятию обоснованных решений в различных ситуациях;
- привлечение внимания к социальным и экологическим вопросам, что способствует

формированию социальной ответственности;
- поощрение инициативности и креативности в реализации проектов и идей, что укрепляет уверенность в своих силах.

3. Работа с родителями

Работа с родителями обучающихся детского объединения нацелена на:
- развитие практических навыков в организации органов самоуправления, а также изучение основ этики и психологии общения и технологий социального и творческого проектирования;
- обучение навыкам организаторской деятельности, самоорганизации и формированию чувства ответственности как за себя, так и за других;
- стимулирование творческого, культурного и коммуникативного потенциала детей через участие в совместной общественно полезной деятельности;
- поддержание формирования активной гражданской позиции у родителей и детей;
- воспитание осознанного отношения к труду, природе и своему городу.

Календарный план воспитательной работы на 2026-2027 уч. год

№	Мероприятие	Задачи	Сроки проведения	Примечание
Модуль «Воспитательная среда»				
СЕНТЯБРЬ				
1	День интернета в России	Формирование навыков поведения в информационном обществе с целью обеспечения информационной безопасности	сентябрь	
ОКТЯБРЬ				
2	Интернет безопасность	Нормы и правила поведения детей в сети Интернет	октябрь	
НОЯБРЬ				
3	День народного единства	Формирование патриотизма, уважения к истории и традициям Родины; формирование чувства гордости и уважения к защитникам государства; развитие мировоззренческих убеждений на основе осмысления исторических событий.	ноябрь	
ДЕКАБРЬ				
4	День конституции	Знакомство с понятием конституции РФ Привить патриотизм к своей стране Знать историю своего государста	Декабрь	
ЯНВАРЬ				
5	День рождения Википедии	Экскурс в Википедию Расширение кругозора детей	январь	
ФЕВРАЛЬ				
8	День компьютерщика (14 февраля)	Урок из истории первых компьютеров, о развитии языков программирования Развитие кругозора	февраль	

МАРТ				
9	Международный женский день	Создать атмосферу уважения и понимания, а также способствовать личностному и нравственному развитию учащихся.	март	
АПРЕЛЬ				
10	День космонавтики	Прививать интерес к изучению космоса и истории космонавтики. Воспитывать чувство патриотизма и гражданственности.	апрель	
МАЙ				
10	День победы	Воспитывать чувство патриотизма и гражданской ответственности. Знание истории.	май	
Модуль «Работа с родителями»				
1.	Организационное родительское собрание	Знакомство родителей с целями и задачами обучения по данной ДОП, особенностями организации учебного процесса, режимом работы и учебным графиком	сентябрь	
2.	Индивидуальные консультации для родителей	Решение вопросов социального и педагогического характера	в течение учебного года	
3.	Открытые занятия для родителей	Знакомство родителей с промежуточными результатами работы объединения	декабрь, апрель	
4.	Итоговое родительское собрание	Подведение итогов работы объединения, знакомство с результатами итоговой аттестации обучающихся	май	