

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ ТВЕРСКОЙ ОБЛАСТИ
МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ
ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«ЦЕНТР ОБРАЗОВАНИЯ»
СТРУКТУРНОЕ ПОДРАЗДЕЛЕНИЕ
ЦЕНТР ЦИФРОВОГО ОБРАЗОВАНИЯ ДЕТЕЙ «ИТ-КУБ»**

РАССМОТРЕНО
на заседании
педагогического совета
протокол №5 от 20.05.2026

УТВЕРЖДЕНО
Директор МБОУ «Центр
образования» (ЦЦОД «ИТ-КУБ»)
О.С. Савинцева
Приказ №30 от 20.05.2026



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
««Программирование на языке PYTHON»»**

Направленность: техническая
Общий объем программы: 144 часа
Возраст обучающихся: 12-17 лет
Срок реализации программы: 2 год
Уровень: базовый;

Авторы: педагоги дополнительного образования О.В. Терентьева

Торжок 2026 г

Информационная карта программы

Название	ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА ««Программирование на языке PYTHON»»
Направленность	техническая
Общий объем программы в часах	144 часа
Форма реализации	Очная
Целевая категория обучающихся	Обучающиеся в возрасте 12-17 лет
Аннотация программы	Предлагаемая программа нацелена на создание условий для изучения методов программирования на языке Python; рассмотрение различных парадигм программирования, предлагаемых этим языком (процедурная, функциональная, объектно-ориентированная); подготовка к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях.
Планируемые результаты реализации программы	умение определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей; развитие логических способностей и алгоритмического мышления, умения составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами – линейной, условной и циклической;

1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа **«Программирование на языке Python»** составлена в соответствии с требованиями:

- Федеральный закон от 29.12.2012 № 273 – ФЗ «Об образовании в Российской Федерации»;

- Приказ Минпросвещения России от 27.07.2022 N 629 "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам";

- письма Минобрнауки РФ от 11.12.2006 г. № 06-1844 «О примерных требованиях к программам дополнительного образования детей»;

- письма Минобрнауки РФ от 18.11.2015 г. № 09-3242 «О направлении информации» вместе с методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы);

- приказа Министерства образования Тверской области от 23.09.2022 г. №939/ПК «Об утверждении Регламента проведения независимой оценки качества дополнительных образовательных программ Тверской области»

Направленность программы

Техническая. Программа предназначена для использования в системе дополнительного образования детей.

Новизна данной образовательной программы

Данная программа охватывает алгоритмическое направление, а также вопросы практического использования полученных знаний при решении задач. Предоставляется возможность командной разработки, создания коллективных проектов. Учащиеся смогут увидеть результаты своего труда в сети Интернет.

Актуальность программы

Программа ориентирована на изучение языка программирования Python. Это современный язык программирования, основными достоинствами

которого являются: кроссплатформенность, бесплатность, простой и понятный синтаксис, высокая читаемость кода программы, богатство возможностей. Он активно применяется в самых разных областях от разработки веб-приложений до машинного обучения.

Цель программы - создание условий для изучения методов программирования на языке Python; рассмотрение различных парадигм программирования, предлагаемых этим языком (процедурная, функциональная, объектно-ориентированная); подготовка к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях.

Для успешной реализации поставленной цели необходимо решить ряд образовательных, развивающих и воспитательных **задач**:

Образовательные:

1. Формирование и развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
2. Знакомство с принципами и методами функционального программирования;
3. Знакомство с принципами и методами объектно-ориентированного программирования;
4. Приобретение навыков работы в интегрированной среде разработки на языке Python;
5. Изучение конструкций языка программирования Python;
6. Знакомство с основными структурами данных и типовыми методами обработки этих структур;
7. Приобретение навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;

Развивающие:

1. Развивать образное мышление;
2. Приобретение навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
3. Развитие у обучающихся интереса к программированию;
4. Формирование самостоятельности и творческого подхода к решению задач с использованием средств вычислительной техники;

Воспитательные:

1. Воспитать умение работать в коллективе с учетом личностных качеств учащихся, психологических и возрастных особенностей.
2. Воспитать трудолюбие и уважительное отношения к интеллектуальному труду.
3. Воспитание упорства в достижении результата;
4. Расширение кругозора обучающихся в области программирования.

Отличительная особенность.

Отличительной особенностью данной программы являются компетенции, которые приобретает ученик по окончании курса:

- знание основ современных языков программирования;
- умение объяснять и использовать на практике как простые, так и сложные структуры данных и конструкции для работы с ними;
- умение искать и обрабатывать ошибки в коде;
- умение разбивать решение задачи на подзадачи;
- способность писать грамотный, красивый код;
- способность анализировать как свой, так и чужой код;
- способность работать с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения профессиональных задач (в том числе на основе системного подхода);

- способность грамотно строить коммуникацию, исходя из целей и ситуации.

Адресат программы: Возраст детей, участвующих в реализации данной программы: от 12 до 17 лет.

Количество обучающихся в группе: Наполняемость в группах - до 12 человек.

Возможные формы проведения занятий:

- на этапе изучения нового материала – лекция, объяснение, рассказ, демонстрация, игра;
- на этапе практической деятельности – беседа, дискуссия, практическая работа;
- на этапе освоения навыков – творческое задание;
- на этапе проверки полученных знаний – публичное выступление с демонстрацией результатов работы, дискуссия, рефлексия.

Рекомендуемые методы проведения занятий:

- метод проблемного обучения;
- метод дизайн-мышления;
- метод проектной деятельности.

Режим занятий: Занятия проводятся 1 раз в неделю по 2 академических часа. Продолжительность 1 академического часа - 45 минут.

Особенности организации образовательного процесса.

Образовательный процесс осуществляется в группах с детьми разного возраста. Состав группы постоянный; количество учащихся 12 человек.

Ожидаемые результаты:

Личностные:

- формирование ответственного отношения к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам;

- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию;
- развитие опыта участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- формирование ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Метапредметные:

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая;
- умение критически оценивать правильность решения учебно-исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;

- владение основами самоконтроля, способность к принятию решений;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий (ИКТ-компетенция);
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Предметные:

- умение определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей, создавать на их основе несложные программы анализа данных, читать и понимать несложные программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня;
- формирование представлений об основных предметных понятиях («информация», «алгоритм», «исполнитель», «модель») и их свойствах;
- развитие логических способностей и алгоритмического мышления, умения составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами – линейной, условной и циклической;
- умение выполнять пошагово (с использованием компьютера или вручную) несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;
- навыки и опыт разработки программ в выбранной среде программирования, включая тестирование и отладку программ; умение использовать основные управляющие конструкции объектно-

ориентированного программирования и библиотеки прикладных программ, выполнять созданные программы;

- умение разрабатывать и использовать компьютерно-математические модели, оценивать числовые параметры моделируемых объектов и процессов, интерпретировать результаты, получаемые в ходе моделирования реальных процессов, анализировать готовые модели на предмет соответствия реальному объекту или процессу;
- формирование умения соблюдать нормы информационной этики и права.

Компетентностный подход реализации программы позволяет осуществить формирование у обучающегося как личностных, так и профессионально-ориентированных компетенций через используемые формы и методы обучения, нацеленность на практические результаты.

В процессе обучения по программе у обучающегося формируются:

универсальные компетенции (SoftSkills):

- умение работать в команде: работа в общем ритме, эффективное распределение задач и др.;
- наличие высокого познавательного интереса;
- умение ориентироваться в информационном пространстве, продуктивно использовать техническую литературу для поиска сложных решений;
- умение ставить вопросы, связанные с темой проекта, осуществлять выбор наиболее эффективных решений задач в зависимости от конкретных условий;
- наличие критического мышления;
- проявление технического мышления, познавательной деятельности, творческой инициативы, самостоятельности;
- способность творчески решать технические задачи;

- готовность и способность к применению теоретических знаний по математике и информатике для решения задач в реальном мире;
- способность правильно организовывать рабочее место и время для достижения поставленных целей;

предметные компетенции (HardSkills):

обучающиеся должны *знать*:

- правила безопасного пользования оборудованием;
- назначение и функции используемых информационных технологий;
- правила создания и представления мультимедийной презентации;
- принципы обработки звуковой и видеоинформации;
- основные свойства алгоритма, типы алгоритмических конструкций: следование, ветвление, цикл; понятие вспомогательного алгоритма;
- способы планирования деятельности, разделение задач на подзадачи, распределения ролей в рабочей группе;
- базовые языки программирования;
- язык программирования Scratch;

обучающиеся должны *уметь*:

- соблюдать технику безопасности;
- составить план проекта, включая: выбор темы; анализ предметной области; разделение задачи на подзадачи;
- использовать основные алгоритмические конструкции для решения задач;
- оперировать информационными объектами, используя графический интерфейс: открывать, именовать, сохранять объекты, архивировать и разархивировать информацию;
- искать информацию с применением правил поиска в компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным темам;

- эффективно использовать интегрированную среду разработки;
- разрабатывать программные проекты на основе использования разных технологий программирования;
- подготовить отчет о проделанной работе; публично выступить с докладом
- составлять алгоритмы управления исполнителями и записывать их на языке программирования Scratch;
- тестировать, отлаживать и выполнять программу по шагам;
- рационально выбирать инструменты для решения поставленной задачи;

обучающиеся должны *владеть*:

- базовыми языками программирования для решения поставленной задачи.

Способы определения результативности реализации программы и формы подведения итогов реализации программы.

В процессе обучения проводятся разные виды контроля результативности усвоения программного материала.

Текущий контроль проводится на занятиях в виде наблюдения за успехами каждого обучающегося, процессом формирования компетенций. Текущий контроль успеваемости носит безотметочный характер и служит для определения педагогических приемов и методов для индивидуального подхода к каждому обучающемуся, корректировки плана работы с группой.

Периодический контроль проводится по окончании изучения каждой темы в виде представления практических результатов выполнения заданий. Конкретные проверочные задания промежуточной аттестации разрабатывает педагог с учетом возможности проведения промежуточного анализа процесса формирования компетенций. Периодический контроль проводится в виде педагогического анализа результатов анкетирования, тестирования, зачётов, опросов, выполнения учащимися диагностических заданий, участия

обучающихся в мероприятиях (викторинах, соревнованиях), активности обучающихся на занятиях и т.п.

Итоговый контроль проводится в виде педагогического анализа результатов выполнения учащимися диагностических заданий, участия обучающихся в мероприятиях (викторинах, соревнованиях), защиты проектов, решения задач поискового характера. Итоги реализации программы могут подводиться в виде итоговой аттестации следующих форм: защита индивидуального или группового проекта в виде публичного выступления с демонстрацией проектной работы; выставка; соревнование; взаимооценка обучающимися работ друг у друга. В процессе проведения итоговой аттестации оценивается результативность освоения программы.

Критерии оценивания приведены в таблицах 1,2,3.

Таблица 1

Критерии оценивания сформированности компетенций
SoftSkills и HardSkills

Уровень	Описание поведенческих проявлений
1 уровень - недостаточный	Обучающийся не владеет навыком, не понимает его важности, не пытается его применять и развивать.
2 уровень – развивающийся	Обучающийся находится в процессе освоения данного навыка. Обучающийся понимает важность освоения навыков, однако не всегда эффективно применяет его в практике.
3 уровень – опытный пользователь	Обучающийся полностью освоил данный навык. Обучающийся эффективно применяет навык во всех стандартных, типовых ситуациях.
4 уровень – продвинутый пользователь	Особо высокая степень развития навыка. Обучающийся способен применять навык в нестандартных ситуациях или ситуациях повышенной сложности.
5 уровень – мастерство	Уровень развития навыка, при котором обучающийся становится экспертом в среде сверстников. Обучающийся способен передавать остальным необходимые знания и навыки для освоения и развития данного навыка.

Таблица 2

Критерии оценивания проекта

	Критерий	Баллы (от 0 до 3)
Оценка представленной работы: (тема)		
1.	Обоснование выбора темы. Соответствие содержания сформулированной теме, поставленным целям и задачам.	1 – не было обоснования темы, цель сформулирована нечетко, тема раскрыта не полностью 2 – был обоснован выбор темы, цель сформулирована нечетко, тема раскрыта не полностью 3 – было обоснование выбора темы, цель сформулирована в соответствии с темой, тема раскрыта полностью
2.	Рефлексия Владение рефлексией; социальное и прикладное значение полученных результатов (для чего? чему научились?), выводы	0 – нет выводов 1 – выводы по работе представлены неполно 2 – выводы полностью соответствуют теме и цели работы
Оценка выступления участников:		
3.	Качество публичного выступления, владение материалом	1 – участник читает текст 2 – участник допускает речевые и грамматические ошибки 3 – речь участника грамотная и безошибочная, хорошо владеет материалом
4.	Качество представления продукта проекта.	1 – участники представляют продукт 2 – оригинальность представления продукта 3 – оригинальность представления и качество выполнения продукта
5.	Умение вести дискуссию, корректно защищать свои идеи, эрудиция докладчика	1 – не умеет вести дискуссию, слабо владеет материалом 2 – участник испытывает затруднения в умении отвечать на вопросы комиссии и слушателей 3 – участник умеет вести дискуссию. Доказательно и корректно защищает свои идеи
6.	Дополнительные баллы (креативность - новые оригинальные идеи и	0-3

	пути решения, особое мнение эксперта)	
--	---------------------------------------	--

Таблица 3

Критерии оценивания уровня освоения программы

Уровни освоения программы	Результат
Высокий уровень освоения программы	Обучающиеся демонстрируют высокую заинтересованность в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом тестировании показывают отличное знание теоретического материала, практическое применение знаний воплощается в качественный продукт
Средний уровень освоения программы	Обучающиеся демонстрируют достаточную заинтересованность в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом тестировании показывают хорошее знание теоретического материала, практическое применение знаний воплощается в продукт, требующий незначительной доработки
Низкий уровень освоения программы	Обучающиеся демонстрируют низкий уровень заинтересованности в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом тестировании показывают недостаточное знание теоретического материала, практическая работа не соответствует требованиям

2. Содержание программы

2.1 УЧЕБНЫЙ ПЛАН

дополнительной общеобразовательной общеразвивающей программы
«Программирование на языке Python».

№	Наименование тем	Количество часов		
		всего	теорет.	практ.
1.	Числа, строки, переменная, простые функции	2	1	1
2.	Логический тип, сравнения, если иначе	2	1	1
3.	Преобразование типов данных, практика	2	1	1
4.	Цикл while	2	1	1
5.	Ошибки и отладчик	2	1	1
6.	Цикл for	4	2	2
7.	True, False, break, continue	2	1	1
8.	Вложенные циклы	2	1	1
9.	Множества	2	1	1
10.	Строки, индексация	2	1	1
11.	Строки, срезы	2	1	1
12.	Списки	2	1	1
13.	Кортежи и преобразование коллекций	2	1	1
14.	split и join, списочные выражения	4	2	2
15.	Методы списков и строк	4	2	2
16.	Вложенные списки	4	2	2
17.	Словари	2	1	1
18.	Функции	2	1	1
19.	Возвращение значений из функции	2	1	1
20.	Области видимости	2	1	1
21.	Передача параметров	2	1	1
22.	Переменное число аргументов	2	1	1
23.	Функции как объект и лямбдафункции	2	1	1
24.	Потоковый ввод	4	2	2
25.	Встроенные модули	4	2	2
26.	Работа с графическими файлами	4	2	2
27.	Морфология	2	1	1
28.	Работа с форматами office	4	2	2
29.	Введение в ООП	4	2	2
30.	Полиморфизм	4	2	2
31.	Магические методы	4	2	2
32.	Наследование	4	2	2
33.	Шахматы	10	5	5

34.	Проектная деятельность	12	4	8
35.	Защита итогового проекта	2	1	1
36.	Решение задач	10	4	6
37.	«Развитие критического мышления»	18	9	9
ИТОГО:		144	72	72

2.2. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН
дополнительной общеобразовательной общеразвивающей программы
«Программирование на языке Python».

№	Наименование тем	Количество часов			Форма аттестации / контроля
		всего	теорет.	практ.	
1.	Числа, строки, переменная, простые функции	2	1	1	Тестирование по пройденному материалу
2.	Логический тип, сравнения, если иначе	2	1	1	Тестирование по пройденному материалу
3.	Преобразование типов данных, практика	2	1	1	Тестирование по пройденному материалу
4.	Цикл while	2	1	1	Тестирование по пройденному материалу
5.	Ошибки и отладчик	2	1	1	Тестирование по пройденному материалу
6.	Цикл for	4	2	2	Тестирование по пройденному материалу
7.	True, False, break, continue	2	1	1	Тестирование по пройденному материалу
8.	Вложенные циклы	2	1	1	Тестирование по пройденному материалу
9.	Множества	2	1	1	Тестирование по пройденному материалу
10.	Строки, индексация	2	1	1	Тестирование по пройденному материалу
11.	Строки, срезы	2	1	1	Тестирование по пройденному материалу
12.	Списки	2	1	1	Тестирование по пройденному материалу
13.	Кортежи и преобразование коллекций	2	1	1	Тестирование по пройденному материалу
14.	split и join, списочные выражения	4	2	2	Тестирование по пройденному материалу

15.	Методы списков и строк	4	2	2	Тестирование по пройденному материалу
16.	Вложенные списки	4	2	2	Тестирование по пройденному материалу
17.	Словари	2	1	1	Тестирование по пройденному материалу
18.	Функции	2	1	1	Демонстрация проектов
19.	Возвращение значений из функции	2	1	1	Тестирование по пройденному материалу
20.	Области видимости	2	1	1	Тестирование по пройденному материалу
21.	Передача параметров	2	1	1	Тестирование по пройденному материалу
22.	Переменное число аргументов	2	1	1	Тестирование по пройденному материалу
23.	Функции как объект и лямбдафункции	2	1	1	Тестирование по пройденному материалу
24.	Потоковый ввод	4	2	2	Тестирование по пройденному материалу
25.	Встроенные модули	4	2	2	Тестирование по пройденному материалу
26.	Работа с графическими файлами	4	2	2	Тестирование по пройденному материалу
27.	Морфология	2	1	1	Тестирование по пройденному материалу
28.	Работа с форматами office	4	2	2	Тестирование по пройденному материалу
29.	Введение в ООП	4	2	2	Тестирование по пройденному материалу
30.	Полиморфизм	4	2	2	Тестирование по пройденному материалу
31.	Магические методы	4	2	2	Тестирование по пройденному материалу
32.	Наследование	4	2	2	Тестирование по пройденному материалу
33.	Шахматы	10	5	5	Тестирование по пройденному материалу
34.	Проектная деятельность	12	4	8	Тестирование по пройденному материалу
35.	Защита итогового проекта	2	1	1	Демонстрация проектов

36.	Решение задач	10	4	6	
37	«Развитие критического мышления»	18	9	9	Вводное тестирование. Итоговое тестирование
ИТОГО:		144	72	72	

2.3 СОДЕРЖАНИЕ ЗАНЯТИЙ
по дополнительной общеобразовательной общеразвивающей
программе «Программирование на языке Python».

№	Наименование тем	Количество часов			Содержание занятий
		всего	теорет.	практ.	
1.	Числа, строки, переменная, простые функции	2	1	1	Введение
2.	Логический тип, сравнения, если иначе	2	1	1	Изучение логического типа данных (bool), операций сравнения, условного оператора if-else, построение ветвлений в программах.
3.	Преобразование типов данных, практика	2	1	1	Освоение явного и неявного преобразования типов (int, float, str), решение задач на приведение типов в реальных сценариях.
4.	Цикл while	2	1	1	Изучение цикла с предусловием while, написание программ с повторяющимися действиями, управление условиями выхода из цикла.
5.	Ошибки и отладчик	2	1	1	Знакомство с типами ошибок (синтаксические, логические, исключения), освоение приемов отладки кода и использования отладчика.

6.	Цикл for	4	2	2	Изучение цикла с параметром for, итерация по диапазонам и коллекциям, особенности работы с range.
7.	True, False, break, continue	2	1	1	Углубление в логические значения, управление циклами с помощью break и continue, оптимизация выполнения циклов.
8.	Вложенные циклы	2	1	1	Создание конструкций с циклами внутри циклов, решение задач на обработку табличных данных и комбинаторику.
9.	Множества	2	1	1	Изучение структуры данных set, операций над множествами (объединение, пересечение, разность), применение для уникальных элементов.
10.	Строки, индексация	2	1	1	Знакомство со строками как последовательностями символов, доступ к элементам по индексу, основные свойства неизменяемости.
11.	Строки, срезы	2	1	1	Освоение механизма извлечения подстрок с помощью срезов, шаги и отрицательные индексы в срезах.

12.	Списки	2	1	1	Изучение списков как изменяемых коллекций, методы добавления, удаления и изменения элементов.
13.	Кортежи и преобразование коллекций	2	1	1	Знакомство с неизменяемыми кортежами, преобразование между различными типами коллекций (list, tuple, set).
14.	split и join, списочные выражения	4	2	2	Работа с методами split и join для обработки строк, освоение генераторов списков для лаконичного создания коллекций.
15.	Методы списков и строк	4	2	2	Обзор и практика применения основных методов списков (append, remove, sort) и строк (upper, replace, find).
16.	Вложенные списки	4	2	2	Создание и обработка многомерных списков, работа с матрицами и табличными данными.
17.	Словари	2	1	1	Изучение структуры dict, работа с парами ключ-значение, методы словарей и их применение.
18.	Функции	2	1	1	Введение в создание функций, параметры, вызов, повторное

					использование кода, инкапсуляция логики.
19.	Возвращение значений из функции	2	1	1	Использование return для передачи результатов работы функции, возврат нескольких значений.
20.	Области видимости	2	1	1	Понятие локальных и глобальных переменных, правила доступа к переменным в разных блоках кода.
21.	Передача параметров	2	1	1	Способы передачи аргументов в функции (позиционные, именованные), изменяемые и неизменяемые типы.
22.	Переменное число аргументов	2	1	1	Использование *args и **kwargs для создания гибких функций, принимающих произвольное количество параметров.
23.	Функции как объект и лямбдафункции	2	1	1	Рассмотрение функций как объектов первого класса, создание анонимных функций с lambda, передача функций в качестве аргументов.
24.	Потоковый ввод	4	2	2	Организация ввода данных из файлов и стандартного потока ввода, обработка больших объемов информации.

25.	Встроенные модули	4	2	2	Знакомство со стандартной библиотекой Python, импорт и использование встроенных модулей (math, random, datetime).
26.	Работа с графическими файлами	4	2	2	Основы обработки изображений, чтение и сохранение графических файлов, простые манипуляции с пикселями.
27.	Морфология	2	1	1	Введение в лингвистический анализ текста с помощью библиотек (py morphology2 и др.), нормализация слов.
28.	Работа с форматами office	4	2	2	Чтение и запись файлов Excel и Word с использованием библиотек (openpyxl, python-docx), автоматизация обработки документов.
29.	Введение в ООП	4	2	2	Основные понятия объектно-ориентированного программирования: классы, объекты, атрибуты, методы.
30.	Полиморфизм	4	2	2	Реализация полиморфизма в Python, переопределение методов,

					единый интерфейс для разных типов объектов.
31.	Магические методы	4	2	2	Знакомство со специальными методами класса (init , str , add и др.), настройка поведения объектов.
32.	Наследование	4	2	2	Построение иерархий классов, наследование атрибутов и методов, расширение функциональности родительских классов.
33.	Шахматы	10	5	5	Применение полученных знаний для создания логики шахматной игры, моделирование фигур и правил ходов.
34.	Проектная деятельность	12	4	8	Выбор темы индивидуального или группового проекта, планирование этапов работы, распределение задач.
35.	Защита итогового проекта	2	1	1	Презентация разработанного проекта, демонстрация работоспособности кода, ответы на вопросы и самоанализ.
36.	Решение задач	10	4	6	Комплексное повторение пройденного материала через решение разнообразных задач

					различного уровня сложности.
37	«Развитие критического мышления»	18	9	9	Специальное занятие с метапредметным подходом: анализ проблемных ситуаций, поиск нестандартных решений, рефлексия собственного мышления.
ИТОГО:		144	72	72	

2.4. Календарный учебный график реализации программы

Год обучения	Название программы	Количество часов			Количество учебных		Даты начала и окончания	Продолжительность каникул
		все го	теория	практика	недель	дней		
1	Программирование на языке Python	72	22	50	36	36	01.09.26 31.05.27	10 дней, январь
2	Программирование на языке Python	72	22	50	36	36	01.09.27 31.05.28	9 дней, январь
	Итого	144	44	100	72	72		19

3. Организационно-педагогические условия реализации программы

«Программирование на языке Python»

3.1 Материально-техническое обеспечение

Программа реализуется на базе Муниципального бюджетного общеобразовательного учреждения «Центр образования». Помещение – учебный кабинет, оформленный в соответствии с профилем проводимых занятий и оборудованный в соответствии с санитарными нормами.

№ п/п	Наименование	Количество, шт.
1.	Презентационное оборудование	
1.1	Моноблочное интерактивное устройство TeachTouch 65"	1
2.	Компьютерное оборудование	
2.1	Ноутбук HP 15-bc419ur (4GS86EA)	13
2.2	Мышь A4Tech N-708X-1 Grey USB	13
3.	Программное обеспечение	
3.1	Офисное ПО Office Standart 2019	13
3.2	Scratch 3.0	13

3.2 Информационное обеспечение

Список рекомендуемой литературы

Для педагога

1. К. Ю. Поляков, Е. А. Еремин. Информатика. Углублённый уровень. Учебник для 10 класса в 2 частях. М.: БИНОМ. Лаборатория знаний, 2014.
2. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011.
3. Задачи по программированию. Под ред. С.М.Окулова, М.: БИНОМ. Лаборатория знаний, 2006.
4. С. М. Окулов. Основы программирования. М.: Бином. Лаборатория знаний, 2012.

Для обучающегося

1. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011.
2. Информатика и ИКТ. Задачник-практикум в 2 частях. Под ред. И.Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.

Для родителей

1. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011.
2. Информатика и ИКТ. Задачник-практикум в 2 частях. Под ред. И.Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.