

Муниципальное бюджетное образовательное учреждение
дополнительного образования
«Центр информационных технологий»
МО Тосненский район Ленинградской области

Принята на заседании
Педагогического совета МБОУ
Протокол № 3
от « 30 » марта 2026 г.

УТВЕРЖДЕНА
Приказ № 10/1 о-д
от «30» марта 2026 г.



Директор МБОУ ДО ЦИТ
Л. И. Левчикова

Дополнительная общеразвивающая образовательная программа
«Программирование на Python»

Срок освоения: 36 дней
Возраст обучающихся: 13-16 лет

Разработчик программы: Сынкова Кристина Юрьевна,
педагог дополнительного образования

г. Тосно
2026

СОДЕРЖАНИЕ

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	3
2. УЧЕБНЫЙ ПЛАН	7
3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК	8
4. РАБОЧАЯ ПРОГРАММА	8
4.1. Содержание программы	9
4.2. Календарно-тематический план	10
5. МЕТОДИЧЕСКИЕ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ.....	13
5.1. Методическое и материально-техническое обеспечение	13
5.2. Информационные источники	15
5.3. Оценочные материалы	15

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеразвивающая программа «Программирование на Python» (далее – Программа) имеет техническую направленность, что предполагает ориентацию занятий по программе на развитие интереса детей к инженерно-техническим и информационным технологиям, технологической грамотности и современных компетенций обучающихся в области технических и естественных наук, инженерных профессий; формированию предпрофессиональных навыков в сфере инженерии и технического творчества.

Адресат: Программа адресована детям в возрасте 11-16 лет. В группе могут быть и мальчики, и девочки. Возможно формирование групп разновозрастных детей.

Наличие базовых знаний по Программе и специальных способностей в данной предметной области не требуется.

Разноуровневость программы предоставляет всем детям возможность занятий независимо от способностей и уровня общего развития. Программа предполагает параллельные процессы освоения содержания программы на его разных уровнях углубленности, доступности и степени сложности исходя из диагностики и стартовых возможностей каждого из участников рассматриваемой программы. Содержание и материал программы организован по принципу дифференциации в соответствии со следующими уровнями сложности:

1. «Стартовый уровень». Предполагает использование и реализацию общедоступных и универсальных форм организации материала, минимальную сложность предлагаемого для освоения содержания программы.

2. «Базовый уровень». Предполагает использование и реализацию таких форм организации материала, которые допускают освоение специализированных знаний и языка, гарантированно обеспечивают трансляцию общей и целостной картины в рамках содержательно-тематического направления программы.

3. «Продвинутый уровень». Предполагает использование форм организации материала, обеспечивающих доступ к сложным (возможно узкоспециализированным) и нетривиальным разделам в рамках содержательно-тематического направления программы. Также предполагает углубленное изучение содержания программы и доступ к профессиональным знаниям в рамках программы. Каждый участник программы имеет право на стартовый доступ к любому из представленных уровней.

Актуальность продиктована необходимостью создать условия для вовлечения детей в приобретение умений в области программирования, технических наук, технологической грамотности и инженерного мышления, а также потребностью детей и родителей в получении необходимых теоретических знаний и навыков в сфере программирования для личностного развития и успешной адаптации к требованиям жизни в современном информационном обществе. Актуальность Программы соотносится с тенденциями развития дополнительного образования и согласно Концепции развития дополнительного образования способствует формированию и развитию имеющихся навыков операционного и логического стиля мышления, исследовательских и творческих качеств.

Нормативно-правовое обеспечение программы

1. Федеральный закон от 29.12.2012 N 273-ФЗ (ред. от 30.12.2021) "Об образовании в Российской Федерации" (с изм. и доп., вступ. в силу с 01.03.2022).

2. Постановление Правительства РФ от 26.12.2017 N 1642 (ред. от 27.02.2023) "Об утверждении государственной программы Российской Федерации "Развитие образования".

3. Приказ Министерства просвещения Российской Федерации от 27.07.2022 г. № 629

«Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

4. Постановление Главного государственного санитарного врача Российской Федерации от 28 сентября 2020 г. № 28 «Об утверждении СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организации воспитания и обучения, отдыха и оздоровления детей и молодежи».

5. Конвенция о правах ребенка (принята резолюцией 44/25 Генеральной Ассамблеи от 20 ноября 1989 г.).

6. Концепция развития дополнительного образования детей до 2030 года (распоряжение Правительства Российской Федерации от 31.03.2022 № 678-р).

7. Стратегия развития воспитания в Российской Федерации до 2025 года (распоряжение Правительства РФ от 29.05.2015 N 996-р).

8. Концепция духовно-нравственного развития и воспитания личности гражданина России.

9. Постановление Главного государственного санитарного врача Российской Федерации от 28 января 2021 г. № 2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» // Статья VI. Гигиенические нормативы по устройству, содержанию и режиму работы организаций воспитания и обучения, отдыха и оздоровления детей и молодежи (Требования к организации образовательного процесса, таблица 6.6)

10. Письмо Минпросвещения России от 31.01.2022 № ДГ-245/06 «О направлении методических рекомендаций по реализации дополнительных общеобразовательных программ с применением электронного обучения и дистанционных образовательных технологий».

11. Профессиональный стандарт «Педагог дополнительного образования детей и взрослых» (Приказ Минтруда и соц. защиты РФ от 22 сентября 2021 года N 652н).

12. Письмо Министерства образования и науки Российской Федерации от 18.11.2015 09-3242 «О направлении методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)».

13. Закон Ленинградской области "Об образовании в Ленинградской области" от 16.04.2024 N 46-оз с 1 сентября 2024 года.

14. Устав и локальный акт МАОУ ДО «ЦИТ».

Уровень освоения – общекультурный.

Объем Программы – 72 часа, **срок освоения** – 36 дней.

Срок реализации программы – 1 год; периодичность занятий – 1 раз в неделю; длительность одного занятия — 2 академических часа. Учебный час составляет 45 мин. Между двумя учебными часами перерыв для отдыха – 5 мин. Максимальная наполняемость групп - 12 человек. Во время осенних и весенних каникул в общеобразовательных организациях в соответствии с п.11 ч.1. ст.34 ФЗ «Об образовании в РФ» № 273-ФЗ допускается свободное посещение учащимися занятий. В это время допускается временное изменение расписания, места и формы проведения занятий (выходы в социокультурные учреждения).

Цель Программы – познакомить обучающихся с основами программирования на языке Python, обучить алгоритмическому подходу, расширить навыки решения задач с использованием компьютерных технологий и подготовить к дальнейшему изучению программирования и информационных технологий.

Задачи Программы

Образовательные:

- познакомить с основными элементами программирования;
- научить работать с различными типами данных на языке Python;
- дать представление о написании программ на языке Python для решения задач разной сложности;
- познакомить с базовыми понятиями теории алгоритмов.

Развивающие:

- совершенствовать умение действовать по алгоритму;
- сформировать умение писать чистый, понятный и документированный код;
- развивать умение планировать свои действия с учётом фактора времени.

Воспитательные:

- создать условия для развития взаимопомощи и сотрудничества путём совместного выполнения задач;
- содействовать развитию самостоятельности в достижении желаемого результата программы.

Планируемые результаты

Предметные:

- будет знать основные элементы программирования;
- будет владеть основами работы с различными типами данных на языке Python;
- будет иметь представление о том, как писать программы на языке Python для решения задач разной сложности;

Метапредметные:

- будет усовершенствовано умение действовать по алгоритму;
- обучающийся сможет писать чистый, понятный и документированный код;
- обучающийся сможет планировать свои действия с учётом фактора времени.

Личностные:

- будут развиты умения взаимопомощи и сотрудничества путём совместного выполнения задач;
- обучающийся сможет самостоятельно достигать желаемого результата программы.

Организационно-педагогические условия реализации

Язык программы: русский.

Форма обучения: очная.

Условия формирования группы: списочный состав групп формируется в соответствии с технологическим регламентом и с учетом особенностей реализации программы.

Условия набора: набор детей осуществляется на свободной основе, принимаются все желающие.

Количество обучающихся: не менее 10 человек.

Особенности реализации: программа предусматривает только аудиторную работу с обучающимися с учетом возрастных и индивидуальных особенностей обучающихся.

Основной принцип построения занятия использование различных форм образовательной деятельности на основе этапов, которые могут объединяться в одном занятии:

1. Организационный этап (создание эмоционального настроения в группе, упражнения и игры с целью привлечения внимания детей);
2. Мотивационный этап (сообщение темы занятия, пояснение тематических понятий, выяснение исходного уровня знаний детей по данной теме);
3. Практический этап (проектирование и разработка программ, отработка полученных навыков на практике);
4. Рефлексивный этап (презентация работ друг другу или игра в созданные программы, обобщение полученных знаний, подведение итогов занятия).

Формы проведения занятий

Программа предусматривает только аудиторные занятия.

Занятие строится в игровой форме лекций и практических занятий.

Формы организации занятий:

Учебные занятия проводятся в группах, сочетая следующие формы:

- фронтальная: участвует весь состав объединения – беседа, объяснение, показ, игровая ситуация;
- групповая: работа в малых группах (парах) при выполнении определенных заданий педагога или групповых проектов;
- индивидуальная: при необходимости педагог оказывает индивидуальную помощь ребенку при выполнении заданий.

Приёмы и методы организации учебно- воспитательного процесса

Методы, в основе которых лежит способ организации занятия:

1. Словесный (устное изложение, беседа).
2. Наглядный (показ иллюстраций, наблюдение, показ педагогом, работа по образцу).
3. Практический (практическая работа).

Методы, в основе которых лежит уровень деятельности детей:

1. Объяснительно-иллюстративный – дети воспринимают и усваивают готовую информацию.
2. Репродуктивный – учащиеся воспроизводят полученные знания и освоенные способы деятельности.
3. Частично-поисковый – участие детей в поиске, решение поставленной задачи совместно с педагогом.
4. Исследовательский – самостоятельная творческая работа учащихся.

Методы, в основе которых лежит форма организации деятельности учащихся занятия:

1. Фронтальный – одновременная работа со всеми учащимися.
2. Групповой – организация работы по малым группам (от 2 до 7 человек).
3. Парный – организация работы по парам.
4. Индивидуальный – индивидуальное выполнение заданий, решение проблем.

Приёмы: игры, упражнения, решение проблемных ситуаций, диалог, устное изложение, беседа, наблюдение, работа по образцу, тренинг, практические работы и др.

Материально-техническое оснащение

Для успешной реализации Программы необходимы:

- классный кабинет, оборудованный компьютерами в соответствии с санитарными нормами: столами и стульями для педагога и обучающихся, классной доской, шкафами для хранения учебной литературы и наглядных пособий;
- компьютер для учителя, мультимедийный проектор и/или экспозиционный экран (интерактивная доска);
- программное обеспечение для занятий: среда программирования PyCharm, интернет-браузер, графические редакторы Paint Tool SAI и Paint.

2. УЧЕБНЫЙ ПЛАН

№	Раздел, тема занятия	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1	Вводное занятие. Техника безопасности при работе за компьютером.	2	1	1	В процессе занятия
2	Синтаксис языка Python.	6	3	3	В процессе занятия
3	Типы данных в языке Python.	4	2	2	В процессе занятия
4	Составление алгоритма программы.	6	3	3	В процессе занятия
5	Условные и логические операторы	6	3	3	В процессе занятия
6	Циклы.	8	4	4	В процессе занятия
7	Введение в объектно-ориентированное программирование.	10	3	4	В процессе занятия
8	Разработка игр	12	4	8	В процессе занятия
9	Работа в командных проектах и решение групповых задач.	6		6	В процессе занятия
10	Разработка индивидуальных проектов на свободную тему.	10		10	В процессе занятия, защита проектов
11	Контрольное занятие.	2		2	Итоговый контроль

Итого	72	22	50	
--------------	----	----	----	--

3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Этапы образовательного процесса	1 год обучения
Дата начала занятий	1 сентября
Дата окончания занятий	31 мая
Количество учебных недель	36
Количество учебных дней	36
Количество учебных часов	72
Режим занятий	Продолжительность работы - 130 минут (2 академических часа по 45 минут)

4. РАБОЧАЯ ПРОГРАММА

Особенность обучения

Программа «Программирование на Python» построена по принципу, который предполагает чередование различных форм представления материала для знакомства.

Задачи

Образовательные:

- познакомить с основными элементами программирования;
- научить работать с различными типами данных на языке Python;
- дать представление о написании программ на языке Python для решения задач разной сложности;
- познакомить с базовыми понятиями теории алгоритмов.

Развивающие:

- совершенствовать умение действовать по алгоритму;
- сформировать умение писать чистый, понятный и документированный код;
- развивать умение планировать свои действия с учётом фактора времени.

Воспитательные:

- создать условия для развития взаимопомощи и сотрудничества путём совместного выполнения задач;
- содействовать развитию самостоятельности в достижении желаемого результата программы.

4.1. Содержание программы

1. Вводное занятие. Техника безопасности при работе за компьютером

Теория: Правила техники безопасности и поведения в аудитории. Язык программирования Python. Знакомство со средой программирования PyCharm (1 час).

Практика: игровая ситуация «Давайте познакомимся» (1 час).

2. Синтаксис языка Python.

Теория: Правила написания кода программы и его документирование. Переменные. Подключение библиотек. Математические операторы (3 часа).

Практика: выполнение задания «Математические вычисления» (3 часа).

3. Типы данных в языке Python

Теория: Типы переменных. Использование разных типов для решения задач (2 часа).

Практика: выполнение заданий «Определение правильного типа данных» и «Преобразование типов данных» (2 часа).

4. Составление алгоритма программы

Теория: Определение входных и выходных данных. Выделение шагов. Понятие блок-схем. Виды блоков. Построение блок-схем (3 часа).

Практика: создание алгоритмов для небольших программ, игра «Продолжи алгоритм» (3 часа).

5. Условные и логические операторы

Теория: Понятие условных операторов. Инструкция if – elif – else. Логические операторы and, or, not. Проверка истинности условий (3 часа).

Практика: выполнение заданий «Условия» и «Определение истинности» (3 часа).

6. Циклы

Теория: Понятие циклов в программировании. Циклы while и for. Конструкция Switch-Case. Операторы break и continue (4 часа).

Практика: выполнение заданий «Калькулятор», «Генератор случайных чисел» и «Поиск минимального и максимального элемента» (4 часа).

7. Введение в объектно-ориентированное программирование

Теория: Функции. Классы и объекты. Организация кода с использованием модулей (3 часа).

Практика: создание игры «Крестики-нолики» и приложения «Управление задачами» (4 часа).

8. Введение в объектно-ориентированное программирование

Теория: Этапы разработки игры. Подготовка материалов. Библиотека pygame и её особенности. Создание визуальной среды. Создание анимаций с помощью функций. (4 часа).

Практика: разработка игры-платформера (8 часов).

9. Работа в командных проектах и решение групповых задач

Практика: выполнение заданий по ролям, создание программы в паре (6 часов).

10. Разработка индивидуальных проектов на свободную тему

Практика: разработка своего проекта на свободную тему. Выбор темы и планирование, подготовка материалов, программирование и тестирование (10 часов).

11. Контрольное занятие

Практика: викторина «Элементы программирования», представление и защита своих проектов (2 часа).

Планируемые результаты

Предметные:

- будет знать основные элементы программирования;
- будет владеть основами работы с различными типами данных на языке Python;
- будет иметь представление о том, как писать программы на языке Python для решения задач разной сложности;

Метапредметные:

- будет усовершенствовано умение действовать по алгоритму;
- обучающийся сможет писать чистый, понятный и документированный код;
- обучающийся сможет планировать свои действия с учётом фактора времени.

Личностные:

- будут развиты умения взаимопомощи и сотрудничества путём совместного выполнения задач;
- обучающийся сможет самостоятельно достигать желаемого результата программы.

4.2. Календарно-тематический план

№ п/п	Месяц	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля/аттестации
1	Сентябрь	Фронтальная, групповая	2	Вводное занятие. Техника безопасности при работе за компьютером	СОШ №4	Наблюдение, опрос, игра – текущий контроль
2		Фронтальная, групповая, индивидуальная	2	Синтаксис на языке Python. Правила написания кода программы и его документирование.	СОШ №4	Наблюдение, обсуждение – текущий контроль
3		Фронтальная, групповая, индивидуальная	2	Синтаксис на языке Python. Переменные и их создание.	СОШ №4	Наблюдение, обсуждение – текущий контроль
4		Фронтальная, групповая, индивидуальная	2	Синтаксис на языке Python. Подключение библиотек.	СОШ №4	Наблюдение, обсуждение – текущий контроль

5	Октябрь	Фронтальная, групповая, индивидуальная	2	Типы данных в языке Python. Кортежи и словари	СОШ №4	Наблюдение, обсуждение – текущий контроль
6		Фронтальная, групповая, индивидуальная	2	Типы данных в языке Python. Множества	СОШ №4	Наблюдение, обсуждение – текущий контроль
7		Фронтальная, групповая, индивидуальная	2	Составление алгоритма программы. Выделение шагов.	СОШ №4	Наблюдение, обсуждение – текущий контроль
8		Фронтальная, групповая, индивидуальная	2	Составление алгоритма программы. Разветвления. Основные виды блоков.	СОШ №4	Наблюдение, обсуждение – текущий контроль
9	Ноябрь	Фронтальная, групповая, индивидуальная	2	Составление алгоритма программы. Построение блок-схем	СОШ №4	Наблюдение, обсуждение – текущий контроль
10		Фронтальная, групповая, индивидуальная	2	Условные и логические операторы. Инструкция if – elif – else	СОШ №4	Наблюдение, обсуждение – текущий контроль
11		Фронтальная, групповая, индивидуальная	2	Условные и логические операторы. Логические операторы and, or, not	СОШ №4	Наблюдение, обсуждение – текущий контроль
12		Фронтальная, групповая, индивидуальная	2	Условные и логические операторы. Проверка истинности условий	СОШ №4	Наблюдение, обсуждение – текущий контроль
13	Декабрь	Фронтальная, групповая, индивидуальная	2	Циклы. Операторы while и for	СОШ №4	Наблюдение, обсуждение – текущий контроль
14		Фронтальная, групповая, индивидуальная	2	Циклы. Операторы while и for	СОШ №4	Наблюдение, обсуждение – текущий контроль
15		Фронтальная, групповая, индивидуальная	2	Циклы. Конструкция Switch-Case.	СОШ №4	Наблюдение, обсуждение – текущий контроль
16		Фронтальная, групповая, индивидуальная	2	Циклы. Операторы break и continue.	СОШ №4	Наблюдение, обсуждение – текущий контроль
17	Январь	Фронтальная, групповая, индивидуальная	2	Введение в объектно-ориентированное	СОШ №4	Наблюдение, обсуждение –

				программирование. Классы и объекты		текущий контроль
18		Фронтальная, групповая, индивидуальная	2	Введение в объектно-ориентированное программирование. Классы и объекты	СОШ №4	Наблюдение, обсуждение – текущий контроль
19		Фронтальная, групповая, индивидуальная	2	Введение в объектно-ориентированное программирование. Классы и объекты	СОШ №4	Наблюдение, обсуждение – текущий контроль
20		Фронтальная, групповая, индивидуальная	2	Введение в объектно-ориентированное программирование. Функции	СОШ №4	Наблюдение, обсуждение – текущий контроль
21	Февраль	Фронтальная, групповая, индивидуальная	2	Введение в объектно-ориентированное программирование. Модули	СОШ №4	Наблюдение, обсуждение – текущий контроль
22		Фронтальная, групповая, индивидуальная	2	Разработка игр. Этапы создания игр.	СОШ №4	Наблюдение, обсуждение – текущий контроль
23		Фронтальная, групповая, индивидуальная	2	Разработка игр. Подготовка материалов.	СОШ №4	Наблюдение, обсуждение – текущий контроль
24		Фронтальная, групповая, индивидуальная	2	Разработка игр. Подготовка материалов.	СОШ №4	Наблюдение, обсуждение – текущий контроль
25	Март	Фронтальная, групповая, индивидуальная	2	Разработка игр. Библиотека pygame и её особенности	СОШ №4	Наблюдение, обсуждение – текущий контроль
26		Фронтальная, групповая, индивидуальная	2	Разработка игр. Создание визуальной среды	СОШ №4	Наблюдение, обсуждение – текущий контроль
27		Фронтальная, групповая	2	Разработка игр. Создание анимаций	СОШ №4	Наблюдение, обсуждение – текущий контроль
28		Фронтальная, групповая	2	Работа в командных проектах и решение групповых задач	СОШ №4	Наблюдение, обсуждение – текущий контроль
29	Апрель	Фронтальная, групповая	2	Работа в командных проектах и решение групповых задач	СОШ №4	Наблюдение, обсуждение – текущий контроль
30		Индивидуальная	2	Работа в командных проектах и решение групповых задач	СОШ №4	Наблюдение, обсуждение – текущий контроль

31		Индивидуальная	2	Разработка индивидуальных проектов на свободную тему. Выбор темы и планирование	СОШ №4	Наблюдение, обсуждение – текущий контроль
32		Индивидуальная	2	Разработка индивидуальных проектов на свободную тему. Создание алгоритма	СОШ №4	Наблюдение, обсуждение – текущий контроль
33	Май	Индивидуальная	2	Разработка индивидуальных проектов на свободную тему. Подготовка материалов	СОШ №4	Наблюдение, обсуждение – текущий контроль
34		Индивидуальная	2	Разработка индивидуальных проектов на свободную тему. Программирование и оформление	СОШ №4	Наблюдение, обсуждение – текущий контроль
35		Групповая	2	Разработка индивидуальных проектов на свободную тему. Отладка и тестирование	СОШ №4	Наблюдение, обсуждение – текущий контроль
36		Групповая	2	Контрольное занятие. Защита проектов.	СОШ №4	Защита проектов, викторина – итоговый контроль

5. МЕТОДИЧЕСКИЕ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Методическое и материально-техническое обеспечение

№	Блок программы из УТП	Материально-техническое оснащение, дидактико-методический материал	Форма занятия	Формы, методы, приёмы подачи материала	Формы аттестации/контроля

1	Вводное занятие. Техника безопасности при работе за компьютером.	Мультимедийная презентация по теме, инструкция по технике безопасности	Фронтальная	Словесный	Наблюдение, игра, опрос – текущий контроль
2	Синтаксис языка Python.	Мультимедийная презентация по теме, примеры программ	Фронтальная, групповая, индивидуальная	Словесно-наглядный	Наблюдение, обсуждение – текущий контроль
3	Типы данных в языке Python.	Мультимедийная презентация по теме, примеры программ с использованием разных типов данных.	Фронтальная, групповая, индивидуальная	Словесно-наглядный	Наблюдение, обсуждение – текущий контроль
4	Составление алгоритма программы.	Мультимедийная презентация по теме, задания, карточки для игры.	Фронтальная, индивидуальная	Наглядно-практический	Наблюдение, обсуждение – текущий контроль
5	Условные и логические операторы.	Мультимедийная презентация по теме, задания, карточки для игры.	Фронтальная, групповая, индивидуальная	Наглядно-практический	Наблюдение, обсуждение – текущий контроль
6	Циклы.	Задания, инструкции, блок-схемы.	Групповая, индивидуальная	Наглядно-практический	Наблюдение, обсуждение – текущий контроль
7	Введение в объектно-ориентированное программирование	Мультимедийная презентация по теме, задания, блок-схемы.	Фронтальная, групповая, индивидуальная	Наглядно-практический	Наблюдение, обсуждение – текущий контроль
8	Разработка игр	Мультимедийная презентация по теме.	Фронтальная, групповая, индивидуальная	Наглядно-практический	Наблюдение, обсуждение – текущий контроль
9	Работа в командных проектах и решение групповых задач.	Темы проектов, задания, блок-схемы алгоритмов.	Групповая	Практический	Наблюдение, обсуждение – текущий контроль
10	Разработка индивидуальных проектов на свободную тему.	Темы проектов, блок-схемы	Индивидуальная	Практический	Наблюдение, обсуждение – текущий контроль
11	Контрольное занятие.	Мультимедийная презентация с викториной,	Фронтальная, индивидуальная	Практический	Защита проектов, викторина –

		карточки заданиями	с			ИТОГОВЫЙ контроль
--	--	-----------------------	---	--	--	----------------------

5.2. Информационные источники

Список источников, используемый педагогом

1. Бондаренко А.М. Проектная деятельность – запуск механизма развития личности ребёнка // Эксперимент и инновации в школе, 2011. – №3.
2. Лутц М. Изучаем Python, пер. с англ. — СПб.: Символ Плюс, 2009. — 848 с.
3. Абрамов С.А, Гнездилова Г.Г., Капустина Е.Н., М.И. Селюн. Задачи по программированию. – М.: Наука, 1998. — 168 с.
4. Любанович Б. Простой Python. Современный стиль программирования / Б. Любанович. - СПб.: Питер, 2016. — 204 с.
5. Саммерфильд М. Python на практике / М. Саммерфильд, пер. А.А. Слинкин – М.: ДМК-Пресс, 2014. — 148 с.
6. Фёдоров Д. Ю. Программирование на языке высокого уровня Python: учебное пособие для прикладного бакалавриата. — М. : Издательство Юрайт, 2019. — 161 с.

Список источников, рекомендованный обучающимся:

1. Бэрри П. Изучаем программирование на Python. — М., 2017. — 624 с.
2. Буйначев С. К. Основы программирования на языке Python: учебное пособие. — Екатеринбург: Изд-во Урал. ун-та, 2014. — 91 с.
3. Бхаргава А. Грожаем алгоритмы: иллюстрированное пособие для программистов и любопытствующих. — СПб.: Питер, 2017. — 288 с.
4. Сэнд У., Сэнд К. «Hello World! Занимательное программирование на языке Python» - М.: – 2016. — 114 с.
5. Майк МакГрат «Программирование на Python для начинающих» Эксмо, 2015. — 204 с.
6. Python/Учебник: учебный ресурс — [https:// ru.wikibooks.org/ wiki/ Python/ %D0%A3%D1%87%D0%B5%D0%B1%D0%](https://ru.wikibooks.org/wiki/Python/%D0%A3%D1%87%D0%B5%D0%B1%D0%)
7. Python 3 для начинающих: учебный ресурс — <https://pythonworld.ru/samouchitel-python>

5.3. Оценочные материалы

В процессе реализации программы используются следующие виды контроля:

1. Текущий контроль – в форме наблюдения за игровыми ситуациями с учётом лексического материала конкретной темы.
2. Итоговый контроль (подведения итогов реализации программы) – в форме практических игровых ситуаций с использованием всего объема изученного материала.

Педагогические категории, в которых предлагается выражать результаты знакомства с элементами программирования это:

1. Знания и умения: способность и готовность к их предъявлению по заданию педагога.
2. Характеристики поведения/действий: активность и интерес к коммуникации, самостоятельность.

Текущий контроль – в форме наблюдения за детьми в процессе выполнения заданий с использованием изучаемых элементов программирования и действия по заданному алгоритму (таблица 1);

Таблица 1

Диагностическая карта текущего контроля
(проводится в конце изучения по каждой теме)

№ п/п	Фамилия, имя	Знания			Умения			Сумма баллов	Уровень освоения
		Техника безопасности	Элементы программирова ния по теме	Синтаксис программы	Написание чистого и понятного кода	Составление алгоритма программы	Самостоятельность в достижении желаемого результата программы		
									Высокий Средний Низкий

Текущая оценка приобретенных знаний и умений в программировании оценивается по критериальной системе в соответствии с таблицей 1.

1. Техника безопасности

Высокий – всегда следует рекомендациям техники безопасности (3 балла)

Средний – соблюдает рекомендации техники безопасности в большинстве случаев, иногда нуждается в напоминаниях правил (2 балла)

Низкий – часто нарушает рекомендации техники безопасности, требует постоянного напоминания правил (1 балл)

2. Элементы программирования по теме

Высокий – знает 80% элементов (3 балла)

Средний – знает 60% элементов (2 балла)

Низкий – знает 40% элементов (1 балл)

3. Синтаксис программы

Высокий – безошибочно использует все ключевые конструкции и операторы, уверенно читает и понимает код (3 балла)

Средний – допускает некоторые ошибки в использовании ключевых конструкций и операторов, читает и понимает простой код, но испытывает затруднения с более сложными структурами (2 балла)

Низкий – часто допускает ошибки при использовании ключевых конструкций и операторов, испытывает серьезные трудности при чтении и понимании кода (1 балл)

4. Написание чистого и понятного кода

Высокий – пишет чистый, понятный код и документирует его (3 балла)

Средний – соблюдает большинство правил написания кода, но не документирует его (2 балла)

Низкий – практически не соблюдает правил написания кода (1 балл)

5. Составление алгоритма программы

Высокий – может составить алгоритм программы и написать по нему программу (3 балла)

Средний – может составить алгоритм программы с помощью педагога (2 балла)

Низкий – затрудняется составить алгоритм программы (1 балл)

6. Самостоятельность в достижении желаемого результата программы

Высокий – самостоятельно пишет программу и достигает желаемого результата (3 балла)

Средний – самостоятельно пишет программу, но достигает желаемого результата с помощью педагога (2 балла)

Низкий – затрудняется написать программу (1 балл)

Итоговая оценка приобретённых умений программирования оценивается по критериальной системе и представлена в таблице 2.

Таблица 2

Итоговая диагностическая карта

№ п/п	Фамилия, имя	Обучение				Развитие		Воспитание		Сумма баллов	Уровень освоения
		Техника безопасности	Основные элементы программирования	Синтаксис программы	Составление алгоритма программы	Написание чистого, понятного и документированного кода	Действие по алгоритму	Взаимопомощь и сотрудничество	Самостоятельность в достижении желаемого результата программы		
											Высокий Средний Низкий

Итоговая оценка приобретенных знаний и умений в программировании оценивается по критериальной системе в соответствии с таблицей 2.

1. Техника безопасности

Высокий – всегда следует рекомендациям техники безопасности (3 балла)

Средний – соблюдает рекомендации техники безопасности в большинстве случаев, иногда нуждается в напоминаниях правил (2 балла)

Низкий – часто нарушает рекомендации техники безопасности, требует постоянного напоминания правил (1 балл)

2. Основные элементы программирования

Высокий – знает 80% элементов (3 балла)

Средний – знает 60% элементов (2 балла)

Низкий – знает 40% элементов (1 балл)

3. Синтаксис программы

Высокий – безошибочно использует все ключевые конструкции и операторы, уверенно читает и понимает код (3 балла)

Средний – допускает некоторые ошибки при использовании ключевых конструкций и операторов, читает и понимает простой код, но испытывает затруднения с более сложными структурами (2 балла)

Низкий – часто допускает ошибки при использовании ключевых конструкций и операторов, испытывает серьезные трудности при чтении и понимании кода (1 балл)

4. Составление алгоритма программы

Высокий – составленный алгоритм эффективен и логичен, учитывает возможные варианты входных данных и предусматривает все сценарии (3 балла)

Средний – составленный алгоритм понятен и логичен, но мог бы быть более структурированным, учитывает основные варианты входных данных, но не все возможные сценарии (2 балла)

Низкий – составленный алгоритм программы запутанный и неясный, не учитывает многие важные варианты входных данных и содержит ошибки в построении. (1 балл)

5. Написание чистого и понятного кода

Высокий – пишет чистый, понятный код и документирует его (3 балла)

Средний – соблюдает большинство правил написания кода, но не документирует его (2 балла)

Низкий – практически не соблюдает правил написания кода (1 балл)

6. Действие по алгоритму

Высокий – точно следует составленному алгоритму при написании программы (3 балла)

Средний – следует составленному алгоритму при написании программы, но иногда допускает ошибки (2 балла)

Низкий – имеет трудности с пониманием и следованием составленному алгоритму при написании программы (1 балл)

7. Взаимопомощь и сотрудничество

Высокий – активно помогает другим учащимся в решении задач и преодолении трудностей, эффективно работает в команде. (3 балла)

Средний – участвует в обмене знаниями с другими учащимися, способен работать в команде, но не всегда проявляет активную инициативу (2 балла)

Низкий – не заинтересован в обмене знаниями или сотрудничестве, проявляет заинтересованность только при помощи педагога (1 балл)

8. Самостоятельность в достижении желаемого результата программы

Высокий – самостоятельно пишет программу и достигает желаемого результата (3 балла)

Средний – самостоятельно пишет программу, но достигает желаемого результата с помощью педагога (2 балла)

Низкий – затрудняется написать программу так, чтобы достичь желаемого результата (1 балл)

Конечные критерии уровня освоения Программы:

24 - 17 баллов – высокий уровень.

16 - 9 баллов – средний уровень.

Менее 9 баллов – низкий уровень.