

Министерство образования и науки Тамбовской области

Тамбовское областное государственное бюджетное
образовательное учреждение дополнительного образования
«Центр развития творчества детей и юношества»

Центр цифрового образования детей «IT-Куб»

Рассмотрена и рекомендована
к утверждению Экспертно-методическим
советом ТОГБОУ ДО «Центр развития
творчества детей и юношества»
протокол от 28.08.2023 № 4

«Утверждаю»
директор ТОГБОУ ДО
«Центр развития творчества
детей и юношества»
И.А. Долгий
приказ от 28.08.2023 № 499

**Дополнительная общеобразовательная
общеразвивающая программа
технической направленности**

«Программирование в Minecraft на языке Python»

(уровень освоения: базовый)

Возраст обучающихся: 10 – 12 лет

Срок реализации: 1 год

Автор-составитель:
Воронов-Олемской Георгий Витальевич,
педагог дополнительного образования

г. Тамбов, 2023 год

Информационная карта программы

1. Учреждение	Тамбовское областное государственное бюджетное образовательное учреждение дополнительного образования «Центр развития творчества детей и юношества»
2. Полное название программы	Дополнительная общеобразовательная общеразвивающая программа технической направленности «Программирование в Minecraft на языке Python»
3. Сведения об авторах	
3.1. Ф.И.О., должность автора	Воронов-Олемской Георгий Витальевич, педагог дополнительного образования
4. Сведения о программе:	
4.1. Нормативная база:	<p>Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями на 06.02.2023г);</p> <p>Распоряжение Правительства Российской Федерации от 29 мая 2015 года № 996-р «Стратегия развития воспитания в Российской Федерации на период до 2025 года»;</p> <p>Национальный проект «Образование» (утвержден Президиумом Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам (протокол от 24.12.2018 г. № 16);</p> <p>Приказ Министерства просвещения Российской Федерации от 3 сентября 2019 г. № 467 «Об утверждении Целевой модели развития региональной системы дополнительного образования детей»</p> <p>Приказ Министерства Просвещения Российской Федерации от 27 сентября 2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;</p> <p>Распоряжение Правительства Российской Федерации от 31 марта 2022 года № 678-р «Концепция развития дополнительного образования детей до 2030 года»;</p> <p>Письмо Министерства образования и науки Российской Федерации от 18 ноября 2015г. № 09-3242 «О направлении информации» (методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы);</p> <p>Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (разработанные Минобрнауки России совместно с ГАОУ ВО «Московский государственный педагогический университет», ФГАУ «Федеральный институт развития образования», АНО ДПО «Открытое образование», 2015г.);</p> <p>Постановление Главного государственного санитарного врача Российской Федерации от 28 сентября 2020 г. № 28 «Об утверждении санитарных правил 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;</p> <p>Устав ТОГБОУ ДО «Центр развития творчества детей и юношества»;</p> <p>Локальные акты, регламентирующие образовательную деятельность ТОГБОУ ДО «Центр развития творчества детей и юношества»</p>
4.3. Область применения	дополнительное образование
4.4. Направленность	техническая

4.6. Вид программы	дополнительная	общеобразовательная	общеразвивающая
	программа		
4.7. Образовательная область	познавательное развитие		
4.8. Уровень освоения	базовый		
4.9. Возраст учащихся	10-12 лет		
4.10. Продолжительность обучения	1 год		

Блок № 1. «Комплекс основных характеристик дополнительной общеобразовательной общеразвивающей программы»

1.1 Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Программирование в Minecraft на языке Python» (далее – Программа) имеет техническую направленность, уровень освоения программы – базовый. Реализация программы позволит обучающимся получить необходимый объем знаний о возможностях использования Python и создать положительную мотивацию, способствующую развитию личностных, метапредметных и предметных умений, а также позволит им раскрыть свой творческий потенциал в сфере современных it-технологий.

Актуальность и практическая значимость программы

При этом Python является современным, востребованным и перспективным языком разработки, позволяющим не только заложить фундамент знаний будущих программистов и инженеров, но получить востребованные и актуальные навыки, необходимые в учебной, проектной и повседневной деятельности.

Язык программирования Python принято считать одним из самых простых в освоении. Обучаться языку возможно имея лишь базовые навыки использования компьютера, установка необходимого ПО не сложнее установки любой другой программы, а синтаксис языка не перегружен и интуитивно понятен. Данные факторы раскрывают Python максимально эффективным инструментом при обучении детей программированию.

Создание компьютерных игр и обучающих программ способствует развитию логики и креативного мышления у детей. Ещё одной значимой стороной обучения программированию является спрос на рынке труда на специалистов данного направления. По итогам освоения программы обучающиеся получают возможность для социального, культурного и профессионального самоопределения, творческой самореализации личности по средствам мероприятий технической направленности, а также овладеют soft и hard компетенциями.

Отличительная особенность программы заключается в возможности получения обучающимися универсальных компетенций, необходимых при дальнейшем изучении не только информационных технологий, но и предметов гуманитарного и естественно-научного цикла. Также в результате изучения парадигмы объектно-ориентированного подхода к программированию происходит формирование базовых знаний и умений для работы с большинством популярных языков и необходимых при освоении других IT-направлений.

Педагогическая целесообразность программы заключается в возможности сформировать у обучающихся представление о базовых понятиях структурного программирования (данных, переменных, ветвлениях, циклах и функциях). В то же время Python является востребованным языком, он отлично подходит для знакомства с концепцией объектно-

ориентированного программирования и активно применяется в различных областях от разработки веб-приложений до машинного обучения. Научившись программировать на языке Python, обучающиеся получают мощный и удобный в использовании инструмент для решения учебных задач и для создания собственных проектов. Вместе с тем чистота и ясность его конструкций позволит в дальнейшем с лёгкостью выучить любой другой язык программирования.

Адресат программы: программа предназначена для детей в возрасте от 10 до 12 лет.

Возрастные особенности обучающихся

Подростковый возраст – остро протекающий переход от детства к взрослости. С одной стороны, для этого сложного периода показательные негативные проявления, дисгармоничность в строении личности, вызывающий характер его поведения по отношению к взрослым. С другой стороны, подростковый возраст отличается и множеством положительных факторов: возрастает самостоятельность ребенка, более разнообразными и содержательными становятся отношения с другими детьми и взрослыми, значительно расширяется сфера его деятельности. Главное, данный период отличается выходом ребенка на качественно новую социальную позицию, в которой формируется его сознательное отношение к себе как члену общества.

Условия набора обучающихся: для обучения в объединении принимаются все желающие, независимо от уровня первоначальных знаний. Данная программа является базой для перехода на более сложные программы обучения.

Состав группы: постоянный. Нормы наполнения групп – 10 – 12 человек.

Объем и срок освоения программы: программа реализуется в течение 1 учебного года (144 академических часа).

Формы обучения: программа предполагает использование очной и дистанционной формы обучения.

Режим занятий, периодичность и продолжительность занятий: в программе предусматривается выделение фиксированного количества часов на освоение учащимися определенного раздела, при этом обучающиеся могут самостоятельно распределить учебную нагрузку, предоставив выполненное задание к установленной контрольной дате.

1.2 Цель и задачи программы

Цель программы: формирование у обучающихся базовых навыков прикладной разработки на языке программирования Python для решения практических задач и разработки продуктов.

Образовательные задачи:

сформировать навыки алгоритмического и логического мышления, грамотной разработки программ;

познакомить с принципами и методами объектно-ориентированного программирования;

обучить навыкам работы в системе программирования Python;

изучить конструкцию языка программирования Python;

обучить навыкам разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;

Развивающие задачи:

развить у обучающихся интереса к программированию;

развить логическое мышление, творческий и познавательный потенциал обучающегося, его коммуникативных способностей;

развить навыки инженерного мышления, умения работать как по предложенным инструкциям, так и находить свои собственные пути решения поставленных задач;

развить навыки эффективной деятельности в проекте;

развить стрессоустойчивость;

развить способности к самоанализу, самопознанию.

Воспитательные задачи:

воспитать мотивацию учащихся к изобретательству, созданию собственных инженерных и программных реализаций;

привить стремление к получению качественного законченного результата;

привить информационную культуру: ответственное отношение к информации;

сформировать потребность в самостоятельном приобретении и применении знаний, потребность к постоянному саморазвитию;

воспитать социально-значимые качества личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность.

1.3 Содержание программы Учебный план

№ п/п	Название раздела, темы	Количество часов			Форма аттестации/контроля
		всего	теория	практика	
	Вводное занятие	2	1	1	Стартовая диагностика
1	Раздел «История создания языка Python и игры Minecraft»	9	4	8	
1.1.	История создания языка программирования Python	2	1	0	Беседа, практическое задание
1.2.	Установка и настройка необходимых компонентов для программирования в Minecraft с помощью языка Python.	2	1	0	Тест-опрос, практическое задание
1.3.	Запуск spigot и создания профиля игры, создания нового мира в режиме креатив. Тестирования и запуск.	3	1	2	Беседа, практическое задание
1.4	Знакомство с IDLE средой разработки python и установка PyCharm. Запуск редактора кода PyCharm и создание первого проекта. Загрузка модулей mcpi и Minecraft с помощью Pycharm. Проверка работы программы python в Minecraft.	2	1	1	Опрос, практическое занятие
2.	Раздел «Телепортация с помощью переменных»	10	6	10	
2.1.	Что такое переменные и как они устроены?	4	2	2	Опрос, Практическое задание
2.2.	Какие типы данных бывают?	4	2	2	Опрос, практическое задание
2.3.	Проект: «Телепортация игрока с помощью переменных»	2	1	1	Контрольное занятие
3.	Раздел «Математика, моментальное строительство и суперпрыжки»	18	9	9	

3.1.	Выражения и команды, операции	2	1	1	Беседа, практическое задание
3.2.	Сложение	2	1	1	Опрос, творческая работа
3.3.	Проект «Башенка из блоков»	2	1	1	Беседа, самостоятельная работа
3.4.	Проект «Суперпрыжок»	2	1	1	Взаимооценки обучающимися работ друг друга
3.5.	Вычитание	2	1	1	Опрос, практическое задание
3.6.	Проект «Изменение блока под ногами игрока», математические операции и аргументы	2	1	1	Опрос, практическое задание
3.7	Умножение и деление	3	1	2	Контрольное занятие
3.8	Возведение в степень, скобки и порядок выполнения действий, а также работа со случайными числами	3	2	1	Опрос. Контрольное занятие
4	Раздел «Общение в чате при помощи строк»	12	6	18	
4.1.	Понятие типа string, что такое строки?	2	1	1	Беседа, творческая работа
4.2.	Функции ввода и вывода: print() и input()	2	1	1	Опрос, самостоятельная работа
4.3.	Проект «Отправление сообщение в чат»	2	1	1	Тест-опрос, практическое задание
4.4.	Склейка строк и преобразования числа в строку	2	1	1	Опрос, практическое задание
4.5	Проект «Позвольте пользователю выбрать тип блока»	2	1	1	Беседа, педагогическое наблюдение
4.6	Проект «Отчет о перемещениях»	2	1	1	Опрос, контрольное занятие
5.	Раздел «Истина и ложь при помощи булевых значений»	20	9	11	
5.1.	Булевы значения: основы	3	1	2	Тест-опрос, творческая работа
5.2.	Склейка строки и булевых значения. Проект:	3	1	2	Опрос, практическое задание

	«Отставить разрушение блоков»				
5.3.	Операции сравнения: равно. Проект «Игрок в воде»	2	1	1	Беседа. Самостоятельная работа.
5.4.	Операции сравнения: неравно. Проект «Игрок в воздухе»	2	1	1	Опрос. Творческая работа.
5.5	Операции сравнения: больше и меньше, больше или равно, меньше или равно. Проекты: «Игрок над землёй», «Далеко ли игрок от дома»	2	1	1	Опрос. Практическое задание.
5.6	Логические операции: логическое и. Проект: «Игрок под водой»	2	1	1	Педагогическое наблюдение. Практическое задание.
5.7	Логические операции: логическое или. Проект: «Игрок на дереве»	2	1	1	Педагогическое наблюдение, практическое задание
5.8	Логические операции: логическое не. Проект: «Это не тот блок»	2	1	1	Практическое задание
5.9	Порядок выполнения логических операций	2	1	1	Опрос. Контрольное занятие
6.	Раздел «Конструкция if»	14	5	9	
6.1.	Конструкции if и else. Проект: «Как сделать кратер. Можно ли предотвратить разрушения»	3	1	1	Решение задач повышенной сложности.
6.2.	Конструкция elif. Цепочки конструкции elif.	3	1	1	Решение задач повышенной сложности.
6.3	Вложенные конструкции if. Проект: «Потайная дверь»	3	1	2	Решение задач повышенной сложности.
6.4	Проверка диапазона значений с помощью if. Проект: «Ограничьте область телепортации»	3	1	2	Подготовка творческих проектов.
6.5	Логические операции и конструкции if.	2	1	1	Решение задач повышенной сложности. Подготовка творческих проектов.
7.	Раздел «Цикл while»	18	8	10	

7.1	Проект: «Телепортация в случайные места с помощью цикла»	2	1	1	Опрос, практическое задание
7.2	Управление циклами с помощью переменной count	2	1	1	Опрос, практическое задание
7.3	Бесконечный цикл. Проект: «Водяное проклятие»	3	1	2	Самостоятельная работа
7.4	Логические операции и цикл while	3	1	2	Опрос, практическое задание
7.5	Вложенные конструкции if и цикл while	2	1	1	Опрос, практическое задание
7.6	Проверка диапазона значений в условии while	2	1	1	Опрос, практическое задание
7.7	Выход из цикла while с помощью break. Проект: «Постоянный чат на основе цикла while»	2	1	1	Подготовка творческих проектов. Зачет.
7.8	Конструкция while-else. Проект: «Горячо или холодно»	2	1	1	Решение задач повышенной сложности. Подготовка творческих проектов.
8.	Раздел «Функции как источник больших возможностей»	16	6	10	
8.1	Создание и вызов собственных функций.	3	1	2	Решение задач повышенной сложности.
8.2	Функции принимают аргументы, рефакторинг кода. Проект: «Посадите лес, да здравствует рефакторинг»	3	1	2	Решение задач повышенной сложности.
8.3	Комментирование с помощью строк документации, переносы строк в списке аргументов. Проект: «Напоминалка типов блоков»	3	1	2	Решение задач повышенной сложности.
8.4	If и while внутри функции. Проект: «Цвет шерсти»	3	1	2	Решение задач повышенной сложности.
8.5	Понятие глобальные и локальные переменные	4	2	2	Контрольное занятие
9.	Раздел «Списка, словаря и удары по блокам»	12	4	8	
9.1	Работа со списками. Доступ к элементам списка, изменение элементов списка.	3	1	2	Опрос, практическое задание

9.2	Работа со строками как со списками. Основные методы списков. Проект: «Столбик-секундамер»	3	1	2	Опрос, практическое задание
9.3	Кортежи. Методы кортежей. Проект: «Случайный блок»	3	1	2	Опрос, практическое задание
9.4	Словари, создание словаря, методы словарей. Проект: «Удары по блокам и таблицы с результатами»	3	1	2	Подготовка творческих проектов.
10.	Раздел «Цикл for»	11	4	7	
10.1	Простой цикл for. Функция range()	3	1	2	Педагогическое наблюдение, практическое задание
10.2	Эксперименты с функции range(). Проект: «Постройка колонн и пирамид»	2	1	1	Педагогическое наблюдение, практическое задание
10.3	Выход из цикла с помощью break. Проект: «Алмазоискатель»	3	1	2	Решение задач повышенной сложности. Подготовка творческих проектов.
10.4	Вложенные циклы for и многомерные списки. Проект: «Создание пиксель-арта при помощи блоков»	3	1	2	Решение задач повышенной сложности. Подготовка творческих проектов.
	Итоговое занятие	2	0	2	Презентация и защита творческих проектов
	ИТОГО:	144	40	104	

Содержание учебного плана

Вводное занятие.

Теория. Введение в программирование. Техника безопасности.

Практика. Стартовая диагностика. Знакомство с программным окружением.

Раздел 1. «История создания языка Python и игры Minecraft»

Тема 1.1. История создания языка программирования Python

Теория. История развития языков программирования. Проблемы, с которыми столкнулись разработчики перед созданием Python. Основные принципы и цели создания языка Python.

Практика. Знакомство с синтаксисом языка Python. Основные конструкции и структуры языка. Обзор основных библиотек и фреймворков, доступных в Python. Примеры простых программ на языке Python.

Тема 1.2. Установка и настройка компонентов для программирования в Minecraft с помощью Python.

Теория. Введение в программирование в Minecraft. Обзор необходимых компонентов для работы с Minecraft и языком Python.

Практика. Установка и настройка Minecraft и Python. Проверка работоспособности инструментов. Создание первой программы для управления персонажем в Minecraft с использованием Python.

Тема 1.3. Запуск spigot и создания профиля игры, создания нового мира в режиме креатив. Тестирования и запуск.

Теория. История развития игры Minecraft. Основные концепции и механики игры. Режимы игры: выживание и креативный режим. Создание профиля игры и настройка параметров.

Практика. Запуск сервера Spigot для игры Minecraft. Создание нового игрового мира в режиме креатив. Тестирование и запуск игры.

Тема 1.4 Знакомство с IDLE средой разработки python и установка PyCharm. Запуск редактора кода PyCharm и создание первого проекта.

Загрузка модулей mcpi и Minecraft с помощью Pycharm. Проверка работы программы python в Minecraft.

Знакомство с IDLE средой разработки Python и установка PyCharm.

Теория. Ознакомление с IDLE - средой разработки Python. Установка и настройка PyCharm - редактора кода для Python.

Практика. Запуск редактора кода PyCharm и создание первого проекта.

PyCharm. Проверка работы программы Python в Minecraft.

Теория. Описание модулей mcpi и Minecraft для работы с игрой Minecraft через Python. Их установка и подключение в проекте PyCharm.

Практика. Загрузка модулей mcpi и Minecraft с помощью PyCharm. Написание программы на языке Python для работы с игрой Minecraft. Проверка работы программы в Minecraft.

Раздел 2 «Телепортация с помощью переменных»

Тема 2.1. Что такое переменные и как они устроены

Теория. Определение переменных и их роль в программировании. Принцип работы переменных в языке Python. Особенности и правила именования переменных в Python.

Практика. Создание и использование переменных в Python. Примеры работы с различными типами данных в переменных. Применение переменных в простых программах на языке Python.

Тема 2.2. Какие типы данных бывают

Теория. Знакомство с такими типами данных как: `int`, `float`, `str`, `bool`

Практика. Применение на практике при решении математических задач

Тема 2.3. Проект «Телепортация игрока с помощью переменных»

Теория. Применение системы координат при перемещении позиции игрока.

Практика. Написание кода для изменения локации игрока на основе изменения значения переменных.

Раздел 3 «Математика, моментальное строительство и суперпрыжки»

Тема 3.1. Выражения и команды, операции

Теория. Понятие математических выражений и для чего они применяются на практике, а также какие команды и операции имеют.

Практика. Примеры решения задач с команд и операций путем записывания их в математическом выражении.

Тема 3.2. Сложение

Теория. Определение операции сложения. Основные свойства сложения, включая коммутативность и ассоциативность. Понятие нейтрального элемента сложения. Расширенные операции сложения, такие как сложение десятичных дробей и комплексных чисел.

Практика. Примеры простых сложений чисел. Использование операции сложения в решении задач. Работа с различными типами данных, включая целые числа, десятичные дроби и строки. Применение операции сложения в языке Python.

Темы 3.3. Проект «Башенка из блоков»

Теория. Изучение команд для построения блоков и получения их координат. Быстрое построение конструкций с помощью начальной и конечной координаты. Изменение блока или блоков и их атрибутов с помощью команд модуля `mcpi`.

Практика. Построение башни с помощью команд `setblock`, `setblocks`

Тема 3.4. Проект «Суперпрыжок»

Теория. Получения текущих координат игрока с помощью команды `getTilesPos()`, а также изменение координаты `x` для «прыжка» с помощью команды `getTilesPos()`

Практика. Изменение координат `x`, `y`, `z` для «Суперпрыжка»

Тема 3.5. Вычитание

Теория. История развития операций вычитания. Проблемы, с которыми столкнулись математики перед созданием операции вычитания. Основные принципы и цели создания операции вычитания.

Практика. Знакомство с правилами вычитания. Основные методы и подходы операции вычитания. Примеры простых вычислений с использованием операции вычитания.

Тема 3.6 Проект «Изменение блока под ногами игрока», математические операции и аргументы

Теория. Создание простого мода для игры Minecraft. Изменение текстуры блока под ногами игрока. Использование математических операций для расчета новых параметров мода. Работа с аргументами и настройками мода.

Практика. Примеры кода для изменения блока под ногами игрока в игре Minecraft с использованием языка Python

Тема 3.7. Умножение и деление

Теория. Операции умножения и деления в языке Python. Их синтаксис и приоритет выполнения. Примеры использования операций умножения и деления в программировании.

Практика. Написание программ на языке Python, в которых используются операции умножения и деления. Решение простых математических задач, включающих указанные операции. Обработка возможных ошибок и проверка входных данных.

Тема 3.8. Возведение в степень, скобки и порядок выполнения действий, а также работа со случайными числами

Теория. Возведение в степень, скобки и порядок выполнения действий, а также работа со случайными числами.

Практика. Пример решения задач при помощи данных операторов

Раздел 4 «Общение в чате при помощи строк»

Тема 4.1. Понятие типа string, что такое строки?

Теория. Определение типа string и его роль в языке программирования. Особенности работы с строками. Примеры операций и методов, доступных для работы со строками.

Практика. Знакомство с базовыми операциями над строками в Python. Создание и изменение строк. Конкатенация и форматирование строк. Примеры использования методов для работы со строками.

Тема 4.2. Функции ввода и вывода print() и input().

Теория. Основы ввода и вывода данных:

- Операции ввода и вывода данных
- Работа с функциями input() и print()
- Использование различных типов данных при вводе и выводе

Практика. Ввод и вывод данных:

- Запрос ввода данных у пользователя с помощью функции input()
- Вывод данных на экран с помощью функции print()
- Использование различных типов данных при вводе и выводе

Тема 4.3. Проект «Отправление сообщения в чат»

Теория. Знакомство с функцией модуля Minecraft PostToChat() для отправки в чат Minecraft сообщения написанное на языке python

Практика. Написание сообщений в чат при помощи языка python и функции PostToChat()

Тема 4.4. Склеивка строк и преобразование числа в строку

Теория. Операции со строками в языке Python. Методы для склейки строк. Преобразование числа в строку с использованием функции str().

Практика. Примеры склейки строк с помощью оператора "+". Примеры использования методов для склейки строк. Примеры преобразования числа в строку с использованием функции str().

Тема 4.5. Проект «Позвольте пользователю выбрать тип блока»

Теория. Применение изученных функций input(), print(), setblock(), setblock(), для построения блока и изменения его атрибутов.

Практика. Изменения блока при помощи функций выше

Тема 4.6. Проект «Отчет о перемещениях»

Теория. Знакомство с модулем time() и его функцией sleep

Практика. Написание программы, которая будет совершать отчет о перемещениях игрока в чате на основе изменения координат

Раздел 5 «Истина и ложь при помощи булевых значений»

Тема 5.1. Булевы значения: основы

Теория. Понятие булевых значений. Логические операции и операторы сравнения в языке Python. Применение булевых значений в условных выражениях и циклах.

Практика. Создание простых программ с использованием булевых значений. Примеры использования логических операций и операторов сравнения для принятия решений в программе.

Тема 5.2. Склейка строки и булевых значения. Проект: «Отставить разрушение блоков»

Теория. Преобразование булевых типов в строки и наоборот. Применение функции setting("world_immutable", "True").

Практика. Применение изученного материала на практике

Тема 5.3. Операции сравнения: равно. Проект «Игрок в воде»

Теория. Теория операции "=", а также примеры использования операции в коде.

Практика. Примеры решения задач и проверка игрока на нахождении в "воде" на основе получения сведения координат и применения оператора "="

Тема 5.4. Операции сравнения: не равно. Проект «Игрок в воздухе»

Теория. Теория операции "!=", а также примеры использования операции в коде.

Практика. Примеры решения задач и проверка игрока на нахождении в "воздухе" на основе получения сведения координат и применения оператора "!="

Тема 5.5. Операции сравнения: больше и меньше, больше или равно, меньше или равно. Проекты: «Игрок над землей», «Далеко ли игрок от дома»

Теория. Теория операций “<”, “>”, “<=”, “>=”, а также примеры использования операции в коде. Какие способы существуют для расчета длины.

Практика. Примеры решения задач и проверка игрока на нахождении в “воздухе” на основе получения сведения координат и применения операторов: “<”, “>”, “<=”, “>=”. Написание программы для расчета текущих координат от местонахождения дома при помощи теоремы Пифагора.

Тема 5.6. Логические операции: логическое и. Проект: «Игрок под водой»

Теория. Теория операции, ключевое слово “and”. Как узнать на основе координат, что игрок находится под водой.

Практика. Создания проекта и примеры его решения

Тема 5.7. Логические операции: логическое или. Проект: «Игрок на дереве»

Теория. Теория операции, ключевое слово “or”. Как узнать на основе координат, что игрок находится на дереве.

Практика. Создание проекта и примеры его решения.

Тема 5.8. Логические операции: логическое не. Проект: «Это не тот блок»

Теория. Теория операции, ключевое слово “not”. Как узнать на основе координат, что блок не принадлежит к искомому.

Практика. Создание проекта и примеры его решения.

Тема 5.9. Порядок выполнения логических операций

Теория. Какой общепринятые порядок выполнения логических операций существует.

Практика. Применения порядка выполнения логических операций на практике.

Раздел 6 «Конструкция if»

Тема 6.1. Конструкции if и else. Проект: «Как сделать кратер. Можно ли предотвратить разрушения»

Теория. Конструкция if и else является основным средством ветвления в языке программирования Python. С её помощью можно задавать условия выполнения определенных блоков кода в зависимости от истинности или ложности некоторого выражения.

Практика. Применение изученного материала на практике при решении задач

Тема 6.2. Конструкция elif. Цепочки конструкции elif.

Теория. Теория применения конструкции elif и для чего она может потребоваться

Практика. Пример решения задач

Тема 6.3. Вложенные конструкции if. Проект: «Потайная дверь»

Теория. Определение вложенных конструкций и пример выполнения кода.

Практика. Использование вложенных конструкций при решении задач, а также для построения потайной двери и прочих сооружений

Тема 6.4. Проверка диапазона значений с помощью if. Проект: «Ограничьте область телепортации»

Теория. Как можно использовать оператор ветвления if и переменные для проверки диапазонов значения.

Практика. Написание программы с помощью, которой игрок может ограничить область телепортации

Тема 6.5. Логические операции и конструкция if

Теория. Объединение логических операторов и конструкции if для решения более сложных задач

Практика. Примеры решения задач

Раздел 7 «Цикл while»

Тема 7.1. Проект «Телепортация в случайные места с помощью цикла while.

Теория. Введение в цикл while

Практика. Пример программы, которая может телепортировать игрока в случайные координаты

Тема 7.2. Управление циклами с помощью переменной count.

Теория. Введение в понятия «счётчик» в цикле while

Практика. Решения задач при помощи цикла while и переменной count

Тема 7.3. Бесконечный цикл. Проект «Водяное проклятье»

Теория. В каких сферах применения бесконечный цикл может приводить к ошибкам выполнения программы, а в других случаях он необходим для грамотного решения задачи

Практик. Пример выполнения программы, в которой будет появляться вода, куда бы не пошел игрок

Тема 7.4. Логические операции и цикл while

Теория. Как логические операции могут применяться в циклах while.

Практика. Решение задач, где используется заданное количество выполнения одного цикла

Тема 7.5. Вложенные конструкции if в цикле while

Теория. Где и как может применяться конструкция if в цикле while.

Практика. Пример решения задач

Тема 7.6. Проверка диапазона значений в условии while

Теория. Использование логических операторов or, and, not в цикле while

Практика. Решение задач, которые имеют в себе многократное повторение

Тема 7.7. Выход из цикла while с помощью break. Проект: «Постоянный чат на основе цикла while»

Теория. В Python, для выхода из цикла while можно использовать оператор break. Оператор break позволяет немедленно прервать

выполнение цикла и перейти к следующей инструкции после цикла.

Практика. Написание постоянного чата в Minecraft, с использованием ключевого слова break для аварийного завершения выполнения программы

Тема 7.8. Конструкция while-else. Проект «Горячо-или-холодно»

Теория. Применение ключевого слова “else” в цикле while

Практика. Пример решения задач

Раздел 8 «Функции как источник больших возможностей»

Тема 8.1. Создание и вызов собственных функций

Теория. Введение в создание собственных функций с помощью ключевого слова def. Синтаксические правила при написании функций

Практика. Примеры решения задач при помощи функций

Тема 8.2. Функции принимают аргументы, рефакторинг кода.

Проект: «Посадите лес, да здравствует рефакторинг»

Теория. Понятие аргументов в функциях, введение в функциональное программирование

Практика. Пример написания программы, которая может посадить деревья на заданной территории, при помощи созданных функций

Тема 8.3. Комментирование с помощью строк документации, переносы строк в списке аргументов. Проект: «Напоминалка типов блоков»

Теория. Пример написания комментария к программе, какие виды комментариев бывают в языке python.

Практика. Написание программы, которая будет определять типы блоков

Тема 8.4. If и while внутри функции. Проект: «Цвет шерсти»

Теория. Как можно применять цикл while и конструкцию if в функции.

Практика. Написание кода, который будет создавать разноцветные куски шерсти с заданными паттернами.

Тема 8.5. Понятие глобальные и локальные переменные

Теория. Введение в понятие глобальные и локальные переменные, в чем заключается их отличие.

Практика. Решение задач на глобальные переменные

Раздел 9 «Списки, словари и удары по блокам»

Тема 9.1. Работа со списками. Доступ к элементам списка, изменение элементов списка.

Теория. Введение в коллекции данных, ключевые понятия в списках. Методы и основные функции списков

Практика. Решение задач, где применяются списки

Тема 9.2. Работа со строками как со списками. Основные методы списков. Проект: «Столбик-секундамер»

Теория. Использование метода `join()` для объединения элементов списка в одну строку. Метод `split()` позволяет разделить строку на

список элементов, используя заданный разделитель.

Практика. Решение задач на использования методов split и join

Тема 9.3. Кортежи. Методы кортежей. Проект: «Случайный блок»

Теория. Понятие кортежа. В чем главное отличие кортежа от списка. Основные методы

Практика. Использование кортежа для получения из него заданных параметров для получения случайного блока.

Тема 9.4. Словари, создание словаря, методы словарей. Проект: «Удары по блокам и таблицы с результатами»

Теория. Словари, создание словаря, методы словарей.

Практика. Пример решения проекта «Удары по блокам и таблицы с результатами»

Раздел 10 «Цикл for»

Тема 10.1. Простой цикл for. Функция range()

Теория. Введение в цикл for, в чем заключается его отличие от цикла for. Функция range(), в каких случаях применяется.

Практика. Решение задач с использованием функции range()

Тема 10.2. Эксперименты с функции range(). Проект: «Постройка колонн и пирамид»

Теория. Использование цикла for вместо while для оптимизации кода.

Практика. Быстрая постройка конструкции при помощи цикла for

Тема 10.3. Выход из цикла с помощью break. Проект: «Алмазоискатель»

Теория. Применение ключевого слова break для выхода из циклов.

Практика. Написание кода с помощью циклов и ключевого слова break для поиска алмазов в Minecraft

Тема 10.4. Вложенные циклы for и многомерные списки. Проект: «Создание пиксель-арта при помощи блоков»

Теория. Использование вложенных циклов и многомерных (массивов) списков для оптимизации программы.

Практика. Создание пиксельной картины при помощи вложенных циклов и многомерных (массивов) списков.

Итоговое занятие

Подведение итогов и защита творческих проектов.

1.4. Планируемые результаты

Программа направлена на достижение учащимися следующих **личностных** результатов:

формирование ответственного отношения к учению, готовности и способности учащегося к саморазвитию и самообразованию на основе мотивации к обучению и познанию;

формирование целостного мировоззрения, соответствующего современному уровню общества;

развитие осознанного и ответственного отношения к собственным поступкам в сфере использования информации;

формирование коммуникативной компетентности в различных сферах деятельности.

Метапредметные результаты:

умение самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности;

владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;

умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы;

умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;

смысловое чтение;

владение устной и письменной речью;

умение осознанно использовать речевые средства в соответствии с задачей коммуникации;

формирование и развитие компетентности в области использования информационно-коммуникационных технологий (далее ИКТ-компетенции).

Предметные результаты:

в результате реализации программы обучающиеся будут

знать:

место языка Python среди языков программирования высокого уровня;

что такое операция, операнд и их характеристики;

принципиальные отличия величин, структурированных и неструктурированных;

основные операторы языка Python, их синтаксис;

правила описания функций в Python и построение вызова;
принципиальные отличия между формальными, локальными
и глобальными переменными;

область действия описаний в функциях;

свойства данных типа «массив», «матрица»;

уметь:

записывать примеры арифметических и логических выражений всех атрибутов, которые могут в них входить;

разрабатывать программы обработки числовой и символьной информации;

разрабатывать программы (линейные, разветвляющиеся и с циклами);

воспроизводить алгоритмы сортировки массивов и двумерных

массивов, поиска в упорядоченном массиве, распространять эти алгоритмы на сортировку и поиск в нечисловых массивах;

читать и записывать текстовые файлы в заданном формате.

БЛОК №2. «КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ РЕАЛИЗАЦИИ ДОПОЛНИТЕЛЬНОЙ ОБЩЕОБРАЗОВАТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ»

2.1. Календарный учебный график

Учебный год по дополнительной общеобразовательной общеразвивающей программе «Программирование на языке Python» начинается 11 сентября и заканчивается 31 мая, число учебных недель по программе – 36, число учебных дней – 72, количество учебных часов – 144 (Приложение).

Продолжительность каникул с 1 июня по 31 августа 2024 года.

Этапы образовательного процесса	Сроки проведения
Промежуточная аттестация	Декабрь 2023 Май 2024
Итоговая аттестация	Май 2024

2.2. Условия реализации программы

Материально-техническое обеспечение

Для успешной реализации содержания программы необходимо следующее программное и техническое обеспечение:

ноутбук LENOVO IdeaPad S340-15API, 15.6", IPS, AMD Ryzen 5 3500U 2.1ГГц, 12Гб, 512Гб SSD, AMD RadeonVega 8, Windows 10, 81NC009JRU – 12 штук;

интерактивная панель [LMP6501ELRU] Lumien 65" 3840 x 2160 @ 60 Hz, инфракрасный тачскрин 20 касаний, яркость 450cd/m², контрастность 1200:1, матовое покрытие, память 3GB DDR4 + 32GB, Android 8.0, колонки 2x15 Вт, пульт ДУ, 2 стилуса – 1 штука;

комплект программного обеспечения "Мой Офис ОБРАЗОВАНИЕ JetBrainsPyCharm (Python)" – 1 штука;

роутер ASUS RT-AC66U rev B1 // роутер 802.11b/g/n/ac, до 450 + 1300Мбит/с, 2,4 + 5 гГц, 3 антенны, USB, GBT LAN ; 90IG0300-BM3100 – 1 штука;

стол – 12 штук;

стул – 12 штук.

Методическое обеспечение

Методы и формы обучения по программе определяются требованиями федерального государственного образовательного стандарта нового поколения, учетом возрастных и индивидуальных способностей учащихся, дистанционным характером обучения.

Основные приоритеты методики преподавания по данной программе:

междисциплинарная интеграция, содействующая становлению целостного мировоззрения;

интерактивность;

личностно-деятельностный подход в обучении;

вариативное образование, предполагающее построение индивидуальных траекторий обучения и вариативное изменение образовательных моделей, что делает образовательный процесс более гибким и способным удовлетворять разнообразные образовательные потребности личности;

субъект-субъектное педагогическое взаимодействие учащихся и педагогов по достижению совместных целей.

Основные технологии, формы и методы обучения

Образовательный процесс строится по двум основным видам деятельности:

обучение детей теоретическим знаниям (вербальная информация, излагаемая педагогом на основе современных педагогических технологий);

самостоятельная и практическая работа учащихся (изучение основ программирования, выполнение практических заданий, создание собственных проектов и т.д.).

В программе реализуются теоретические и практические блоки, что позволяет наиболее полно охватить и реализовать потребности учащихся, сформировать практические навыки в области программирования. В ходе выполнения самостоятельных работ, учащиеся приобретают навыки работы с различными ресурсами, используемыми для создания собственных проектов, на основе чего происходит выбор оптимальных средств для представления информации в сети Интернет. Таким образом, данная программа позволяет развить у учащихся творческий склад мышления, способности к самостоятельному поиску, решению поставленных проблем, и создать условия для творческого самовыражения личности, что в полной мере соответствует тем требованиям, которые обозначены во ФГОС нового поколения.

Программа имеет разноуровневый характер и рассчитана на учащихся с разным уровнем подготовки. Учебный материал распределен по принципу последовательного расширения и углубления теоретических знаний, приобретения практических умений и навыков.

Кадровое обеспечение программы

Педагоги, организующие образовательный процесс по данной программе должны иметь высшее техническое образование или пройти подготовку на курсах повышения квалификации по применению информационно-коммуникационных технологий. Важным условием, необходимым для реализации программы, является умение педагога осуществлять личностно-деятельностный подход к организации обучения, проектировать индивидуальную образовательную траекторию учащегося, разрабатывать и эффективно применять инновационные образовательные технологии.

2.3. Форма аттестации

Система отслеживания, контроля и оценки результатов процесса обучения по данной программе имеет три основных элемента:

Стартовая диагностика. При приеме детей в объединение педагог проводит тестирование уровня развития мотивации ребенка к обучению для дальнейшего определения образовательного маршрута. Результаты тестирования фиксируются в сводных таблицах.

Текущий контроль предусматривает: тестирование, опросы, соревнования, педагогическое наблюдение, взаимооценка обучающимися работ друг друга, практическое задание, творческая работа, контрольное занятие, зачет, олимпиада, презентация творческих работ.

Уровень освоения программы отслеживается также с помощью выполнения заданий по разработке различных элементов. Задания подбираются в соответствии с пройденным материалом.

Промежуточный контроль, итоговая диагностика. Основной формой подведения итогов является подготовка и защита творческих проектов.

Время проведения	Цель проведения	Формы контроля
<i>Стартовая диагностика</i>		
В начале учебного года	Определение уровня развития детей, их способностей	Тестирование, анкетирование
<i>Текущий контроль</i>		
В течение всего учебного года	<ul style="list-style-type: none"> ➤ Определение степени усвоения обучающимися учебного материала. ➤ Определение готовности детей к восприятию нового материала. ➤ Повышение ответственности и заинтересованности обучающихся в обучении. ➤ Выявление отстающих и опережающих обучение. ➤ Подбор наиболее эффективных методов и средств обучения. 	Тестирование, опрос, соревнование, педагогическое наблюдение, взаимооценки обучающимися работ друг друга, зачет
<i>Промежуточный контроль</i>		
По окончании изучения темы или раздела. В конце месяца, полугодия.	<ul style="list-style-type: none"> ➤ Определение степени усвоения обучающимися учебного материала. ➤ Определение результатов обучения. 	Творческая работа, опрос, контрольное занятие, олимпиада (решение задач повышенной сложности)
<i>Итоговая диагностика</i>		
В конце учебного года или курса обучения	<ul style="list-style-type: none"> ➤ Определение изменения уровня развития детей, их творческих способностей. ➤ Определение результатов обучения. 	Подготовка к защите проектов, защита проектов.

	<ul style="list-style-type: none"> ➤ Ориентирование обучающихся на дальнейшее обучение. ➤ Получение сведений для совершенствования образовательной программы и методов обучения. 	
--	--	--

Формы отслеживания и контроля развивающих и воспитательных результатов:

- оценка устойчивости интереса обучающихся к занятиям с помощью наблюдения педагога и самооценки обучающихся;
- оценка устойчивости интереса обучающихся к участию в мероприятиях, направленных на формирование и развитие общекультурных компетенций с помощью наблюдения педагога и самооценки обучающихся;
- статистический учет сохранности контингента обучающихся;
- сравнительный анализ успешности выполнения заданий обучающимися на начальном и последующих этапах освоения программы;
- анализ творческих и проектных работ обучающихся;
- создание банка индивидуальных достижений обучающихся;
- оценка степени участия и активности обучающегося в командных проектах, соревновательной и конкурсной деятельности;
- оценка динамики показателей развития познавательных способностей обучающихся (внимания, памяти, изобретательности, логического и пространственного мышления и т.д.) с помощью наблюдения педагога и самооценки обучающихся;
- наблюдение и фиксирование изменений в личности и поведении обучающихся с момента поступления в объединение и по мере их участия в деятельности;
- индивидуальные и коллективные беседы с обучающимися.

Критериями выполнения программы служат:

знания, умения и навыки учащихся, позволяющие им комплексно использовать информационные технологии для получения необходимой информации и создания собственных проектов, стабильный интерес к изучению информационно-коммуникационных технологий и их использования в различных сферах деятельности.

2.4. Оценочные материалы

Диагностика развития теоретических знаний и практических навыков осуществляется с помощью диагностических контрольных заданий по следующим критериям:

№ п/п	Оцениваемые параметры	Критерии	Методы диагностики
Теоретическая подготовка учащихся			
1	Теоретические знания по основным разделам учебного плана программы	Соответствие теоретических знаний программным требованиям	Тест-опрос
Практическая работа учащихся			
3	Практические умения и навыки, знания по основным разделам учебного плана программы	Соответствие практических умений и навыков программным требованиям	Опрос, педагогическое наблюдение
4	Владение специальным программным обеспечением	Отсутствие затруднений при работе в среде разработки Лего	Анализ информации
5	Творческие навыки	Сформированный интерес к избранному виду деятельности	Индивидуальный проект

Оценка результатов.

По итогам составляется таблица отслеживания образовательных результатов, в которой обучающиеся по каждой теме выходят на следующие уровни шкалы оценки:

1. Высокий результат – полное освоение содержания;
2. Средний – базовый уровень;
3. Низкий – освоение материала на минимально допустимом уровне (Приложение 2).

2.5 Методическое обеспечение программы

Педагогические технологии

В процессе обучения по программе, используются разнообразные педагогические технологии:

технологии развивающего обучения, направленные на общее целостное развитие личности, на основе активно-деятельного способа обучения, учитывающие закономерности развития и особенности индивидуума;

технологии личностно-ориентированного обучения, направленные на развитие индивидуальных познавательных способностей каждого обучающегося, максимальное выявление, раскрытие и использование его опыта;

технологии дифференцированного обучения, обеспечивающие обучение каждого обучающегося на уровне его возможностей и способностей;

технологии сотрудничества, реализующие демократизм, равенство, партнерство в отношениях педагога и обучающегося, совместно вырабатывают цели, содержание, дают оценки, находясь в состоянии сотрудничества, сотворчества;

проектные технологии – достижение цели через детальную разработку проблемы, которая должна завершиться реальным, осязаемым практическим результатом, оформленным тем или иным образом;

компьютерные технологии, формирующие умение работать с информацией, исследовательские умения, коммуникативные способности.

В практике выступают различные комбинации этих технологий, их элементов.

Формы организации учебных занятий:

- индивидуальное задание;
- комбинированное занятие;
- лекция;
- беседа;
- олимпиада;
- практическое занятие;
- самостоятельная работа;
- мастер-класс;
- итоговое занятие.

Методы образовательной деятельности

В программе кроме традиционных методов используются:

эвристический метод;

исследовательский метод обучения, дающий обучающимся возможность проявить себя, показать свои возможности, добиться определенных результатов;

метод проблемного изложения материала, когда перед обучающимся ставится некая задача, позволяющая решить определенный этап процесса обучения и перейти на новую ступень обучения;

самостоятельная работа; диалог и дискуссия;

приемы дифференцированного обучения, обеспечивающие обучение каждого обучающегося на уровне его возможностей и способностей.

Еще одним основным методом организации учебной деятельности по программе является метод кейсов.

Кейс – описание проблемной ситуации понятной и близкой обучающимся, решение которой требует всестороннего изучения, поиска дополнительной информации и моделирования ситуации или объекта, с выбором наиболее подходящего.

Преимущества метода кейсов:

практическая направленность. Кейс-метод позволяет применить теоретические знания к решению практических задач;

интерактивный формат. Кейс-метод обеспечивает более эффективное усвоение материала за счет высокой эмоциональной вовлеченности и активного участия обучаемых. Участники погружаются в ситуацию с головой: у кейса есть

главный герой, на место которого ставит себя команда и решает проблему от его лица. Акцент при обучении делается не на овладение готовым знанием, а на его выработку;

конкретные навыки. Кейс-метод позволяет совершенствовать универсальные навыки, которые оказываются крайне необходимы в реальном рабочем процессе.

Условно можно выделить следующие **виды кейсов**:

инженерно-практический;

инженерно-социальный;

инженерно-технический;

исследовательский (практический или теоретический).

Построение занятия включает в себя фронтальную, индивидуальную и групповую работу, а также некоторый соревновательный элемент.

Программой предусмотрено проведение комбинированных занятий: занятия состоят из теоретической и практической частей, причём большее количество времени занимает именно практическая часть.

Использование дистанционных образовательных технологий при работе с обучающимися

Главным принципом обучения является принцип доступности, который на практике успешно реализуется в ходе использования дистанционных образовательных технологий и электронного обучения. Такой подход позволяет выстроить индивидуальную образовательную траекторию обучающегося, учесть его образовательные потребности, максимально визуализировать процесс обучения.

В режиме дистанционного обучения возможны следующие **виды работы**:

работа через программу Сферум. Такой урок максимально приближен к обычному уроку, поскольку позволяет общаться с учеником в режиме реального времени (выслушать ответ, оценить ученика, построить диалог);

с использованием сервисов, построенных на основе чат-технологий, где дети имеют возможность обмениваться мнениями, вести переписку, участвовать в обсуждении проблемы при выполнении, например, проекта. Чат-технологии полезны для организации групповых форм работы, рассчитанных на длительный период.

Таким образом, применение в практике обучения дистанционных образовательных технологий способствует расширению образовательных возможностей детей, оптимизирует процесс обучения.

Также при организации процесса обучения с использованием дистанционных образовательных технологий необходимо уделять много внимания использованию здоровьесберегающих технологий или их элементов (проведение физкультурминуток, гимнастики для глаз и т.д.).

2.6. Воспитательный компонент программы

Реализация программы невозможна без осуществления воспитательной работы с обучающимися. Воспитательная работа ведётся на протяжении всего учебного процесса.

Приоритетные направления в организации воспитательной работы:

воспитание нравственных качеств (трудолюбия, настойчивости, целеустремленности) происходит непосредственно в процессе обучения во время совместной деятельности;

духовно-нравственное воспитание формирует ценностные представления о морали, об основных понятиях этики (добро и зло, истина и ложь, смысл жизни, справедливость, милосердие, проблеме нравственного выбора, достоинство, любовь и др.), о духовных ценностях народов России, об уважительном отношении к традициям, культуре и языку своего народа и др. народов России;

трудовое и профориентационное воспитание формирует знания, представления о трудовой деятельности; выявляет творческие способности и профессиональные направления обучающихся;

воспитание познавательных интересов формирует потребность в приобретении новых знаний, интерес к творческой деятельности;

экологическое воспитание формирует ценностные представления и отношение к окружающему миру.

Основные задачи воспитательной работы:

формирование мировоззрения и системы базовых ценностей личности;

организация инновационной работы в области воспитания и дополнительного образования;

организационно-правовые меры по развитию воспитания и дополнительного образования обучающихся;

приобщение обучающихся к общечеловеческим нормам морали, национальным устоям и традициям образовательного учреждения;

обеспечение развития личности и её социально-психологической поддержки, формирование личностных качеств, необходимых для жизни;

воспитание внутренней потребности личности в здоровом образе жизни, ответственного отношения к природной и социокультурной среде обитания;

развитие воспитательного потенциала семьи;

поддержка социальных инициатив и достижений обучающихся.

Основные воспитательные мероприятия:

просмотр обучающимися тематических материалов и их обсуждение;

тематические диспуты и беседы;

участие в конкурсах, соревнованиях, олимпиадах различного уровня.

Работа с коллективом обучающихся:

формирование практических умений по организации органов самоуправления этике и психологии общения, технологии социального и творческого проектирования;

обучение умениям и навыкам организаторской деятельности, самоорганизации, формированию ответственности за себя и других;
развитие творческого культурного, коммуникативного потенциала ребят в процессе участия в совместной общественно – полезной деятельности;
содействие формированию активной гражданской позиции;
воспитание сознательного отношения к труду, к природе, к своему городу.

Работа с родителями:

организация системы индивидуальной и коллективной работы с родителями (тематические беседы, собрания, индивидуальные консультации);

содействие сплочению родительского коллектива и вовлечение в жизнедеятельность кружкового объединения (организация и проведение открытых занятий для родителей в течение года);

оформление информационных уголков для родителей по вопросам воспитания учащихся.

Успешная работа детского объединения во многом зависит от степени участия в ней родителей обучающихся. В большинстве родители заинтересованно относятся к занятиям своих детей в объединении, радуются их успехам и достижениям.

Работа с родителями включает в себя следующие формы деятельности:

родительские собрания;

консультации;

беседы;

работа с семьями, находящимися в трудной жизненной ситуации;

совместные праздники обучающихся и их родителей;

привлечение родителей к подготовке и проведению мероприятий;

приглашение родителей на мероприятия объединения и всего учреждения.

Такая работа способствует формированию общности интересов учащихся и их родителей, служит развитию эмоциональной и духовной близости.

Результат воспитания

В процессе воспитания происходят изменения в личностном развитии обучающихся, в процессе общения со своими сверстниками по достижению общих целей, у ребят формируются такие качества как взаимопомощь, самостоятельность, ответственность за порученное дело. Несомненно, большую роль в воспитании моральных качеств, обучающихся играет личный пример педагога.

СПИСОК ЛИТЕРАТУРЫ

1. Бизли Д.М., Г. Ван Россум. Язык программирования Python. Справочник. (пер. с англ.) Киев: ДиаСофт., 2000.
2. Вордерман Кэрол. Программирование для детей. 2016. – 357 с.
3. Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017. – 284 с.: ил
4. Информатика: Учебник для 10-11 класса / Н.Д. Угринович - М.: Бином, 2009. - 512 с.
5. Лутц М. Программирование на Python. (пер. с англ.) СПб.: Символ- Плюс., 2002.
6. Россум, Г. и др. Язык программирования Python. 2001. 454 с.
7. Сузи Р.А. Python. Наиболее полное руководство. СПб.: БХВ- Петербург., 2002.
8. Сузи Р.А. Язык программирования Python. М: Бином. Лаборатория знаний. - 2006.
9. Фридланд А.Я., Ханамирова Л.С., Фридланд И.А. Информатика и компьютерные технологии. Основные термины. Толковый словарь. М.Издательство Астрель., 2003.
10. Хахаев И.А. Практикум по алгоритмизации и программированию на Python: / И.А. Хахаев - М.: Альт Линукс, 2010. - 126 с.
11. Чаплыгин А. Н. Учимся программировать вместе с Питоном. Revision: 226.

Приложение 1

Календарно-тематический учебный график на 2023 – 2024 учебный год Место проведения занятий: Центр цифрового образования детей «IT-Куб» г. Тамбов, ул. Монтажников 1

№п/п	Наименование темы, раздела	Кол-во часов	Дата	Форма занятия	Форма контроля
	Вводное занятие	2		Групповая	Стартовая диагностика
Раздел 1 «История создания языка Python и игры Minecraft»					
1.1.	История создания языка программирования Python	2		Групповая	Беседа, практическое задание
1.2.	Установка и настройка необходимых компонентов для программирования в Minecraft с помощью языка Python.	2		Групповая	Тест-опрос, практическое задание
1.3.	Запуск spigot и создания профиля игры, создания нового мира в режиме креатив. Тестирования и запуск.	2		Групповая	Беседа, практическое задание
1.4.	Знакомство с IDLE средой разработки python и установка PyCharm. Запуск редактора кода PyCharm и создание первого проекта. Загрузка модулей mcpi и Minecraft с помощью Pycharm. Проверка работы программы python в Minecraft.	2		Групповая	Опрос, практическое занятие
Раздел 2 «Телепортация с помощью переменных»					
2.1	Что такое переменные и как они устроены?	4		Групповая	Опрос, практическое задание
2.2.	Какие типы данных бывают?	4		Групповая	Опрос, практическое задание
2.3.	Проект: «Телепортация игрока с помощью переменных»	2		Групповая	Контрольное занятие
Раздел 3 «Математика, моментальное строительство и супер прыжки»					

3.1	Выражения и команды, операции	2		Групповая	Беседа, практическое задание
3.2	Сложение	2		Групповая	Опрос, творческая работа
3.3	Проект «Башенка из блоков»	2		Групповая	Беседа, самостоятельная работа
3.4	Проект «Суперпрыжок	2		Групповая	Практическое задание, взаимооценки обучающимися работ друг друга
3.5	Вычитание	2		Групповая	Опрос, практическое задание
3.6	Проект «Изменение блока под ногами игрока», математические операции и аргументы	2		Групповая	Опрос, практическое задание
3.7	Умножение и деление	3		Групповая	Контрольное занятие
Раздел 4 «Общение в чате при помощи строк»					
4.1	Понятие типа string, что такое строки?	2		Групповая	Лекция, творческая работа
4.2	Функции ввода и вывода: print() и input()	2		Групповая	Опрос, самостоятельная работа
4.3	Проект «Отправление сообщение в чат»	2		Групповая	Тест-опрос, практическое задание
4.4	Склейка строк и преобразования числа в строку	2		Групповая	Опрос, практическое задание
4.5	Проект «Позвольте пользователю выбрать тип блока»	2		Групповая	Беседа, педагогическое наблюдение
4.6	Проект «Отчет о перемещениях»	2		Групповая	Опрос, контрольное занятие
Раздел 5 Истина и ложь при помощи булевых значений»					
5.1	Булевы значения: основы	3		Групповая	Тест-опрос, творческая работа
5.2	Склейка строки и булевых значения. Проект: «Отставить разрушение блоков»	3		Групповая	Опрос, практическое задание

5.3	Операции сравнения: равно. Проект «Игрок в воде»	2		Групповая	Беседа, самостоятельная работа
5.4	Операции сравнения: неравно. Проект «Игрок в воздухе»	2		Групповая	Опрос, творческая работа
5.5	Операции сравнения: больше и меньше, больше или равно, меньше или равно. Проекты: «Игрок над землёй», «Далеко ли игрок от дома»	2		Групповая	Опрос, практическое задание
5.6	Логические операции: логическое и. Проект: «Игрок под водой»	2		Групповая	Педагогическое наблюдение, практическое задание
5.7	Логические операции: логическое или. Проект: «Игрок на дереве»	2		Групповая	Педагогическое наблюдение, практическое задание
5.8	Логические операции: логическое не. Проект: «Это не тот блок»	2		Групповая	Практическое задание
5.9	Порядок выполнения логических операций	2		Групповая	Опрос, контрольное занятие
Раздел 6 «Конструкция if»					
6.1	Конструкции if и else. Проект: «Как сделать кратер. Можно ли предотвратить разрушения»	3		Групповая	Решение задач повышенной сложности.
6.2	Конструкция elif. Цепочки конструкции elif.	3		Групповая	Решение задач повышенной сложности.
6.3	Вложенные конструкции if. Проект: «Потайная дверь»	3		Групповая	Решение задач повышенной сложности.
6.4	Проверка диапазона значений с помощью if. Проект: «Ограничьте область телепортации»	3		Групповая	Подготовка творческих проектов.
6.5	Логические операции и конструкции if.	2		Групповая	Решение задач повышенной сложности. Подготовка творческих проектов.
Раздел 7 «Цикл while»					
7.1	Проект: «Телепортация в случайные места с помощью цикла»	2		Групповая	Опрос, практическое задание

7.2	Управление циклами с помощью переменной count	2		Групповая	Опрос, практическое задание
7.3	Бесконечный цикл. Проект: «Водяное проклятие»	3		Групповая	Самостоятельная работа
7.4	Логические операции и цикл while	3		Групповая	Опрос, практическое задание
7.5	Вложенные конструкции if и цикл while	2		Групповая	Опрос, практическое задание
7.6	Проверка диапазона значений в условии while	2		Групповая	Опрос, практическое задание
7.7	Выход из цикла while с помощью break. Проект: «Постоянный чат на основе цикла while»	2		Групповая	Подготовка творческих проектов. Зачет.
7.8	Конструкция while-else. Проект: «Горячо или холодно»	2		Групповая	Опрос, практическое задание
Раздел 8 «Функции как источник больших возможностей»					
8.1	Создание и вызов собственных функций.	3		Групповая	Решение задач повышенной сложности.
8.2	Функции принимают аргументы, рефакторинг кода. Проект: «Посадите лес, да здравствует рефакторинг»	3		Групповая	Решение задач повышенной сложности.
8.3	Комментирование с помощью строк документации, переносы строк в списке аргументов. Проект: «Напоминалка типов блоков»	3		Групповая	Решение задач повышенной сложности.
8.4	If и while внутри функции. Проект: «Цвет шерсти»	3		Групповая	Решение задач повышенной сложности.
8.5	Понятие глобальные и локальные переменные	4		Групповая	Контрольное занятие
Раздел 9 «Списка, словаря и удары по блокам»					
9.1	Работа со списками. Доступ к элементам списка, изменение элементов списка.	3		Групповая	Опрос, практическое задание
9.2	Работа со строками как со списками. Основные методы списков. Проект: «Столбик-секундамер»	3		Групповая	Опрос, практическое задание

9.3	Кортежи. Методы кортежей. Проект: «Случайный блок»	3		Групповая	Опрос, практическое задание
9.4	Словари, создание словаря, методы словарей. Проект: «Удары по блокам и таблицы с результатами»	3		Групповая	Подготовка творческих проектов.
Раздел 10 «Цикл for»					
10.1	Простой цикл for. Функция range()	3		Групповая	Педагогическое наблюдение, практическое задание
10.2	Эксперименты с функции range(). Проект: «Постройка колонн и пирамид»	2		Групповая	Педагогическое наблюдение, практическое задание
10.3	Выход из цикла с помощью break. Проект: «Алмазоискатель»	3		Групповая	Решение задач повышенной сложности. Подготовка творческих проектов.
10.4	Вложенные циклы for и многомерные списки. Проект: «Создание пиксель-арта при помощи блоков»	3		Групповая	Решение задач повышенной сложности. Подготовка творческих проектов.
	Итоговое занятие	2		Итоговое занятие	Презентация и защита творческих проектов

Приложение 2

к дополнительной образовательной
общеразвивающей программе

Таблица мониторинга образовательных результатов:

№	Ф.И. Обучающегося	Уровень развития умений и навыков								
		Уровень владения терминологией и теоретическими знаниями по разделам программы			Уровень навыков сборки работа по инструкции.			Уровень навыков создания простейших программ (алгоритмов).		
		Сент.	Дек.	Май.	Сент.	Дек.	Май.	Сент.	Дек.	Май.
1.										
2.										
3.										
4.										
5.										
6.										
7.										
8.										
9.										
10.										
11.										

Оценка результатов.

В таблицу отслеживания образовательных результатов, в которой обучающиеся по каждой теме выходят на следующие уровни шкалы, вносятся оценки:

Высокий результат – полное освоение содержания;

Средний – базовый уровень;

Низкий – освоение материала на минимально допустимом уровне

