

Министерство образования Кировской области
Кировское областное государственное образовательное автономное учреждение дополнительного
образования «Центр технического творчества»
Структурное подразделение «Детский технопарк «Кванториум» в г. КировоЧепецке»

Рассмотрено на заседании
педагогического совета,
протокол №3 от 28.05.2024

УТВЕРЖДЕНО
Приказ №139 от 29.05.2024
Директор



Я.А.Пивоваров

**Дополнительная общеобразовательная общеразвивающая программа
технической направленности по информационным технологиям**

«Основы программирования на языке Python»

Возраст детей: 12-18 лет
Срок реализации: 2 года

Составитель:
Рябчук Даниил Аксентьевич
педагог дополнительного образования

Кирово-Чепецк
2024

ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПРОГРАММЫ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Нормативно-правовая база

Дополнительная общеобразовательная общеразвивающая программа технической направленности «Основы программирования на языке Python» разработана в соответствии с нормативными документами:

- Федеральный закон от 29 декабря 2012 г. № 273ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями от 28 февраля 2023 года);
- Приказ Министерства просвещения РФ от 27 июля 2022г. № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.364820 «Санитарноэпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Распоряжение Правительства РФ от 31 марта 2022 г. № 678р «Концепция развития дополнительного образования детей до 2030 года»;
- Устав, Лицензия на образовательную деятельность, нормативные документы и локальные акты Кировского областного государственного образовательного автономного учреждения дополнительного образования «Центр технического творчества».

Актуальность программы обусловлена современным информационным этапом развития общества, характеризующимся ускоренными темпами освоения большого количества информации, потребностью общества в технически грамотных специалистах в области информационных технологий, а также необходимостью повышения мотивации к выбору современных высокотехнологичных профессий и созданию системы непрерывной подготовки будущих квалифицированных кадров, обладающих практическими знаниями и профессиональными компетенциями для развития приоритетных направлений отечественной науки и техники. Знание языка программирования общего назначения является полезным навыком даже для человека, не являющегося профессиональным программистом. Умение создавать алгоритмы, строить логические зависимости позволяет писать программы облегчающие рутинные задачи. В программе «Основы программирования на языке Python» рассматриваются элементы языка программирования Python, используемый сотнями тысяч разработчиков по всему миру в таких областях, как создание вебсценариев, системное программирование, создание пользовательских интерфейсов, настройка программных продуктов под пользователя, численное программирование и в других. Данная программа представляет собой совокупность междисциплинарных занятий, интегрирующих в себе командную работу и

программирование, основанных на активном обучении детей. Это способствует формированию у обучающихся целостного представления о мире информационных технологий, значении информации, способах обработки и хранения данных. Кроме того, реализация данного направления помогает развитию коммуникативных навыков у обучающихся за счёт активного взаимодействия детей в ходе групповой проектной деятельности.

Отличительные особенности программы

Дополнительной общеразвивающей программы «Основы программирования на языке Python» является использование реальных кейсов и проектной деятельности в качестве основной образовательной технологии, возможность реализации детскими командами реальных программных продуктов, а также возможность организации образовательного процесса, исходя из интересов и способностей обучающихся и геймификация образовательного процесса.

Учебный материал строится с опорой на реальные проекты. Особое внимание при изучении уделяется вариативному решению проблем и задач. Такой подход позволяет развить у обучающихся способность находить оптимальный способ решения задач не только в программировании на языке Python, но и в жизненных ситуациях, требующих выбора. Данная программа формирует профессиональные компетенции и знания в области информационных технологий и математики, которые позволят обучающимся в будущем успешно конкурировать на рынке рабочей силы в сфере программирования.

Новизна программы

Программа отличается тем, что позволяет более глубоко погрузиться в мир программирования. Ученики получают знания в области программирования через практическое применение, что позволяет им изучать теоретические концепции на практике.

Педагогическая целесообразность

Программа направлена на создание условий для гармоничного развития личности ребенка. Освоение основных разделов программы способствует формированию высокого интеллекта. Формы и методы обучения, используемые при реализации программы, полностью соответствуют возрастным и психологическим особенностям детей. Ознакомление с основами языка программирования python, они учатся анализировать и оценивать свою работу, реализуют свое право на выбор. Программа способствует формированию социальных и коммуникативных навыков, расширяет возможности общения, формирует инженерное мышление.

Возрастные особенности группы:

Содержание программы учитывает возрастные и психологические особенности подростков 12–18 лет, которые определяют выбор форм проведения занятий с обучающимися. Подростки этого возраста отличаются внутренней уравновешенностью, стремлением к активной практической деятельности, поэтому основной формой проведения занятий выбраны практические занятия. Ребятам также увлекает совместная, коллективная деятельность, так как резко возрастает значение коллектива, общественного мнения, отношений со сверстниками, оценки

поступков и действий подростка не только со стороны старших, но и со стороны сверстников. Подросток стремится завоевать в глазах сверстников авторитет, занять достойное место в коллективе. Поэтому в структуру содержания программы включены практические задания соревновательного характера. Такие задания позволяют каждому проявить себя и найти своё место в детском коллективе.

Также следует отметить, что подростки данной возрастной группы характеризуются такими психическими процессами, как изменение структуры личности и возникновение интереса к ней, развитие абстрактных форм мышления, становление более осознанного и целенаправленного характера деятельности, проявление стремления к самостоятельности и независимости, формирование самооценки. Эти процессы позволяют положить начало формированию начального профессионального самоопределения обучающихся.

Цель программы: формирование основ программирования на языке Python и навыков командного взаимодействия

Задачи:

обучающие:

- изучить основы языка программирования Python;
- обучить учащихся математической части, применяемой в программировании современных электронных вычислительных машин;
- изучить базовые концепции в программировании на языках высокого уровня;
- обучить работе в интегрированных средах разработки.

развивающие:

- развить навыки инженерного и творческого мышления,
- развить умение работать по предложенным инструкциям программирования, проектирования и эффективного использования электронного вычислительного оборудования;
- развить внимательность, аккуратность и изобретательность при работе с техническими устройствами, выполнении учебных проектов;
- развить умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений и информационного поиска;

воспитательные:

- сформировать мотивацию учащихся к изобретательству, созданию собственных программ;
- сформировать навык проектного мышления,
- уметь работать в команде, эффективно распределять обязанности при реализации проекта.

Режим занятий: программой предусматривается 72 часа в год на вводном уровне и 144 часа на базовом уровне. Для прохождения программного материала

отводится 2 занятия по 2 академических часа в неделю (академический час 40 мин., перерыв 10 мин.).

Программа состоит из двух уровней:

- Вводный (1 год обучения) – программирование 72 часа
- Базовый (2 год обучения) – программирование 144 часа

Перевод учащихся на последующие года обучения осуществляется по достижении ими определенного уровня овладения знаниями, умениями и способами деятельности. При этом учитываются результаты участия в мероприятиях разного уровня.

Основные формы работы с учащимися:

- учебные занятия, консультации, практикумы, самостоятельные работы;
- конкурсы, викторины;
- участие в шахматных турнирах;
- участие в городских, областных и всероссийских конкурсах по информационным технологиям и программированию.

При осуществлении педагогического процесса используются различные *методы обучения:*

- методы организации и осуществления учебнопознавательной деятельности;
- методы стимулирования и мотивации учебнопознавательной деятельности;
- методы контроля и самоконтроля за эффективностью учебнопознавательной деятельности.

На разных этапах педагогического процесса все методы используются комплексно, однако, в зависимости от ситуации, один из методов может быть применен отдельно. Конкретные формы и методы преподавания подбираются с учетом возраста, уровня развития и психофизиологического состояния учащихся. В образовательном процессе акцент делается на технологии личностноориентированного обучения, которые включают индивидуализацию и дифференциацию учебного процесса.

Формы подведения итогов обучения

- фронтальный опрос, беседа;
- контрольные упражнения и тестовые задания;
- выставка;
- межгрупповые соревнования;
- взаимооценка учащимися работ друг друга;
- защита индивидуального или группового проекта

В основе определения результата обучения и воспитания лежит дифференцированный подход, выход учащихся на различные уровни возможностей, способностей и потребностей. В результате реализации данной программы не только сохраняется стабильность посещения занятий в течение учебного года, но и возрастает результативность, повышается интерес к дальнейшей работе. Оценка творческих достижений учащихся проводится в виде игровых заданий, конкурсов различного уровня.

Планируемые результаты освоения программы

Предметными результатами освоения программы являются

на вводном уровне:

- знание основ языка программирования Python
- знание математической части программирования электронных вычислительных машин;
- знание базовых концепций в программировании на языках высокого уровня;
- умение работать в интегрированных средах разработки;

на базовом уровне:

- знание математической части, применяемой в программировании, современных электронных вычислительных машин и мобильной техники;
- знание новых концепций в программировании на языке Python;
- умение работать в интегрированных средах разработки

Метапредметными результатами освоения программы являются:

- формирование инженерного и творческого мышления;
- умение работать по инструкциям программирования, проектирования, использовать электронное вычислительное оборудование;
- внимательность, аккуратность;
- умение четко излагать мысли, отстаивать собственную точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы.

Личностными результатами освоения программы являются:

- приобретение мотивации к дальнейшему развитию;
- формирование структурированного мышления;
- умение работать в команде, распределять обязанности между участниками.

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python» рассчитана на двухгодичный срок обучения (вводный уровень, базовый уровень) для учащихся 12-17 лет. В процессе освоения программы учащимся приобретаются знания по программированию.

Перевод учащихся на уровни осуществляется по достижении ими определенного уровня овладения знаниями, умениями и способами деятельности. Также учитываются результаты участия учащихся в конкурсах и мероприятиях разного уровня.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

Первый год обучения (вводный уровень)

№ п/п	Название раздела	Количество часов		
		Всего	Теория	Практика
1.	Основы алгоритмизации	8	4	4
2.	Основы работы с языком программирования Python	54	6	48
3.	Проектная деятельность	10	2	8
	Итого:	72	12	60

СОДЕРЖАНИЕ

1. Основы алгоритмизации.

Теория

- Инструктаж по технике безопасности. Презентация курса. Оценка имеющихся знаний.
- Понятие алгоритм. Средства описания алгоритма.
- Принципы разработки алгоритмов.

Практика

- Решение алгоритмических задач.

2. Основы работы с языком программирования Python.

Теория

- Обзор языков программирования высокого уровня.
- Знакомство с языком программирования Python.

Практика

- Знакомство с программным обеспечением и принципами работы на занятиях.
- Структура написания программ на Python.
- Знакомство с методами хранения информации.
- Функции вводавывода.
- Решение элементарных математических задач с помощью Python.
- Самостоятельное написание программ.
- Условные операторы.
- Самостоятельное написание программ.
- Базовые типы данных.
- Самостоятельное написание программ.
- Циклы.
- Самостоятельное написание программ.

- Списки.
- Самостоятельное написание программ.
- Функции.
- Командная работа над минипроектом.

3. Проектная деятельность

Теория: Введение в проектную деятельность

Практика: Подготовка проектных работ. Защита проектов

Перечень кейсов (минипроектов), предложенных для выполнения

1. Чат-бот текстовый помощник

Разработка простого чат-бота с базовыми функциями ответа на вопросы.

Реализация взаимодействия с пользователем через текстовый интерфейс.

2. Игра "Угадай число"

Написание логики игры, где пользователю нужно угадать загаданное компьютером число.

Реализация системы подсказок (больше/меньше).

3. Менеджер задач (ToDo List)

Создание приложения для управления списком задач.

Функциональность добавления, удаления, редактирования и завершения задач.

4. Генератор паролей

Разработка приложения для генерации надежных случайных паролей.

Опции выбора длины пароля, наличия специальных символов и т.д.

5. Конвертер валют

Создание программы для конвертации валют с использованием актуальных курсов.

Взаимодействие с API для получения данных о курсах валют.

6. Калькулятор индекса массы тела (BMI)

Написание калькулятора для расчета индекса массы тела.

Интерфейс с вводом роста и веса, вывод рекомендаций.

7. Анализатор текста

Приложение для анализа текста (количество слов, символов, частота слов).

Возможность работы с загружаемыми текстовыми файлами.

8. Квизвикторина

Разработка приложения для проведения викторины с вопросами на различные темы.

Система подсчета баллов и отображения результатов.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН Второй год обучения (базовый уровень)

№ п/п	Наименование раздела	Количество часов		
		Всего	Теория	Практика
1.	Введение в курс	8	4	4
2.	Типы данных	34	8	26
3.	Функции	40	8	32
4.	Модульная структура	40	8	32
5.	Защита проекта	22	0	22
	Итого:	144	28	116

СОДЕРЖАНИЕ

1. Введение в курс.

Теория

- Инструктаж по технике безопасности. Презентация курса. Оценка имеющихся знаний и выявление интересов.

- Междисциплинарный анализ понятия «команда», соотнесение его с понятиями «малая группа», «коллектив», возможные стадии развития группы до уровня команды, определяется содержание понятия «командное взаимодействие» и «целеполагание».

- Создание условий для успешного освоения учениками основ проектноисследовательской деятельности.

Практика

- Знакомство с оборудованием. Обзор средств разработки. Интегрированные среды разработки.

- Создание команд.

- Проектная деятельность.

2. Типы данных.

Теория

- Строки. Принцип обработки строк.

- Списки, методы списков, принцип обработки списков

- Вложенные списки.

- Понятие списка и массива.
- Понятие многомерного массива.
- Индексация в двумерном массиве.
- Трехмерный массив и индексация в нем.
- N-мерный массив.
- Текстовые файлы, принцип работы с ним.
- Негласные правила открытия/закрытия файлов.
- Чтение/запись в файл
- Словари, методы словарей.

Практика

- Задачи с циклами и условным оператором над строками.
- Базовые алгоритмы: Поиск подстроки в строке, замена символов и подобное.
- Задачи на «Стек» и «Очередь», задачи на обработку списков.
- Задачи на двумерные массивы. Сортировка, обработка, слияние и разделение.
- Создание нескольких текстовых файлов, работа с ними в коде.
- Простой аналог заметок или приложения дневника.
- Работа со словарями.

3. Функции

Теория

Понятие функции как способа структурирования кода. Введение в функции: новая структура кода. Область видимости переменных: локальные и глобальные переменные. Способы передачи переменных в функции. Встроенные функции и их использование. Рекурсивные функции и их применение. Лямбда функции и их особенности. Функциональное программирование и функции высшего порядка. Декораторы и их применение в Python. Генераторы и корутины в Python. Функциональные интерфейсы и их роль. Мемоизация и оптимизация функций. Тестирование функций и модульность кода. Контекстные менеджеры и их использование с функциями. Функции в структурах данных. Параллельное выполнение функций. Обработка исключений в функциях. Повторное использование функций и принцип DRY. Полиморфизм и перегрузка функций. Функциональные паттерны проектирования. Оптимизация работы функций и профилирование кода. Модули и их взаимодействие с функциями. Оптимизация времени выполнения функций. Функции как объекты в Python.

Практика

Написание своего кода с использованием своих функций. Решение задач. Применение встроенных функций. Разработка рекурсивных функций для простых и сложных задач. Использование лямбдафункций в контексте различных задач. Создание и применение декораторов. Работа с генераторами и корутинами. Оптимизация функций путем мемоизации. Модульное тестирование функций. Использование функций в контексте данных и структур данных. Параллельное выполнение функций для повышения производительности. Управление

исключениями в функциях. Перегрузка функций и применение полиморфизма. Реализация функциональных паттернов проектирования. Взаимодействие модулей и функций. Профилирование и оптимизация времени выполнения функций. Использование функций как объектов.

4. Модульная структура.

Теория

Введение в модульную структуру программирования. Работа с модулем Random для генерации случайных чисел. Изучение модуля Math и его функций для математических операций. Практическое применение доступных функций модуля Math. Основы математических и тригонометрических функций в Python. Особенности графического интерфейса пользователя (GUI) в программировании. Использование модуля PyGame для разработки игр. Обзор основных функций PyGame.

Практика

Решение практических задач с использованием математических функций. Применение тригонометрических функций в практических задачах. Создание графического интерфейса пользователя (GUI) для программы. Разработка GUI с использованием различных элементов и виджетов. Создание игры с использованием модуля PyGame. Программирование элементов игрового процесса с помощью PyGame. Создание игрового окружения и логики для простой игры. Тестирование и отладка GUI и игровых функций. Написание скриптов для решения математических задач с GUI. Разработка интерактивных графических приложений с математическими функциями. Программирование простой игры с использованием GUI элементов. Практические задания по созданию GUI, математическим функциям и разработке игр.

5. Защита проекта.

Практика

- Введение в проектную деятельность
- Подготовка проектных работ
- Защита проектов

ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

1 год обучения

№ п/п	№ темы в учебнотематическом плане в программе	Тема занятия	Количество часов	Форма проведения занятия	Дата		Форма контроля
					по плану	по факту	
1.	Основы алгоритмизации	Инструктаж по технике безопасности. Презентация курса. Оценка имеющихся знаний.	2	Теория			Наблюдение
2.		Понятие алгоритм. Средства описания алгоритма. Принципы разработки алгоритмов.	2	Теория			Наблюдения
3.		Решение алгоритмических задач.	2	Практика			Тестирование
4.		Решение алгоритмических задач.	2	Практика			Устный опрос
5.	Основы работы с языком программирования Python	Обзор языков программирования высокого уровня	2	Теория			Наблюдение
6.		Знакомство с языком программирования Python	2	Теория			Устный опрос
7.		Знакомство с языком программирования Python	2	Теория			Устный опрос
8.		Знакомство с программным обеспечением и принципами работы на занятиях.	2	Практика			Практическое задание
9.		Структура написания программ на Python.	2	Практика			Наблюдение
10.		Знакомство с методами хранения информации.	2	Практика			Наблюдение
11.		Функции вводавывода.	2	Практика			Практическое задание
12.		Решение элементарных математических задач с помощью Python.	2	Практика			Устный опрос
13.		Самостоятельное написание программ.	2	Практика			Самостоятельная работа

14.		Условные операторы.	2	Практика		Наблюдение
15.		Самостоятельное написание программ.	2	Практика		Письменный опрос
16.		Базовые типы данных.	2	Практика		Наблюдение
17.		Самостоятельное написание программ.	2	Практика		Самостоятельная работа
18.		Циклы.	2	Практика		Наблюдение
19.		Самостоятельное написание программ.	2	Практика		Самостоятельная работа
20.		Списки.	2	Практика		Наблюдение
21.		Самостоятельное написание программ.	2	Практика		Практическое задание
22.		Функции	2	Практика		Устный опрос
23.		Командная работа над мини проектом	2	Практика		Практическое задание
24.		Командная работа над мини проектом	2	Практика		Практическое задание
25.		Командная работа над мини проектом	2	Практика		Практическое задание
26.		Командная работа над мини проектом	2	Практика		Практическое задание
27.		Командная работа над мини проектом	2	Практика		Практическое задание
28.		Командная работа над мини проектом	2	Практика		Практическое задание
29.		Командная работа над мини проектом	2	Практика		Практическое задание
30.		Командная работа над мини проектом	2	Практика		Практическое задание
31.		Командная работа над мини проектом	2	Практика		Практическое задание
32.	Проектная деятельность	Введение в проектную деятельность	2	Теория		Наблюдение тестирование
33.		Подготовка проектных работ к презентации.	2	Практика		Практическое задание
34.		Подготовка проектных работ к презентации.	2	Практика		Практическое задание
35.		Подготовка проектных работ к презентации.	2	Практика		Итоговый тест
36.		Защита проектов	2	Практика		Защита проекта

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

2 год обучения

№ п/п	№ темы в учебнотематическом плане в программе	Тема занятия	Количество часов	Форма проведения занятия	Дата		Форма контроля
					по плану	по факту	
1	Введение в курс	Основы техники безопасности и введение в курс	2	Теория			Наблюдение
2		Понимание командного взаимодействия и целеполагания	2	Теория			Наблюдение
3		повторение работы с оборудованием и средствами разработки	2	Практика			Устный опрос
4		Формирование и работа в командах, повторение понятия «проектная деятельность»	2	Практика			Устный опрос
5	Типы данных	Основы работы со строками	2	Теория			Наблюдение
6		Манипуляции с типом данных "список"	2	Практика			Устный опрос
7		Вложенные списки	2	Теоретико-практическая			Наблюдение
8		Определение списков и массивов	2	Теоретико-практическая			Устный опрос
9		Работа с многомерными массивами	2	Теоретико-практическая			Практическое задание
10		Индексация в двумерных массивах	2	Теоретико-практическая			Практическое задание
11		Трехмерные массивы и их индексация	2	Теоретико-практическая			Практическое задание
12		Работа с n-мерными массивами	2	Теоретико-практическая			Практическое задание

13		Чтение и запись в текстовые файлы	2	Теоретико практическая			Практическое задание
14		Понятие и правила работы с файлами	2	Теоретико практическая			Практическое задание
15		Операции с файлами	2	Теоретико практическая			Практическое задание
16		Использование словарей	2	Теоретико практическая			Практическое задание
17		Использование словарей	2	Теоретико практическая			Письменный опрос
18		Работа с базовыми алгоритмами	2	Теоретико практическая			Практическое задание
19		Задачи на "Стек" и "Очередь"	2	Теоретико практическая			Тест Самостоятельная работа
20		Задания на работу с двумерными массивами	2	Теоретико практическая			Практическое задание
21		Работа с текстовыми файлами и создание приложения дневника. Обработка данных через словари.	2	Теоретико практическая			Практическое задание
22	Функции	Встроенные функции и их использование.	2	Теоретико практическая			Устный опрос
23		Рекурсивные функции и их применение.	2	Теоретико практическая			Практическое задание
24		Лямбдафункции и их особенности.	2	Теоретико практическая			Практическое задание
25		Функциональное программирование и функции высшего порядка.	2	Теоретико практическая			Практическое задание

26	Декораторы и их применение в Python.	2	Теоретико практическая			Устный опрос
27	Генераторы и корутины в Python.	2	Теоретико практическая			Практическое задание
28	Функциональные интерфейсы и их роль.	2	Теоретико практическая			Наблюдение
29	Мемоизация и оптимизация функций.	2	Теоретико практическая			Практическое задание
30	Тестирование функций и модульность кода.	2	Теоретико практическая			Практическое задание
31	Контекстные менеджеры и их использование с функциями.	2	Теоретико практическая			Наблюдение
32	Функции в структурах данных.	2	Теоретико практическая			Наблюдение
33	Параллельное выполнение функций.	2	Теоретико практическая			Наблюдение
34	Обработка исключений в функциях.	2	Теоретико практическая			Наблюдение
35	Повторное использование функций и принцип DRY.	2	Теоретико практическая			Практическое задание
	Полиморфизм и перегрузка функций.	2	Теоретико практическая			Практическое задание
37	Функциональные паттерны проектирования.	2	Теоретико практическая			Практическое задание
38	Оптимизация работы функций и профилирование кода.	2	Теоретико практическая			Практическое задание

39		Модули и их взаимодействие с функциями.	2	Теоретико практическая			Практическое задание
40		Оптимизация времени выполнения функций.	2	Теоретико практическая			Практическое задание
41		Функции как объекты в Python.	2	Теория			Наблюдение
42	Модульная структура	Введение в модульную структуру программирования.	2	Теория			Устный опрос
43		Работа с модулем Random для генерации случайных чисел.	2	Теоретико практическая			Практическое задание
44		Изучение модуля Math и его функций для математических операций.	2	Практическое			Практическое задание
45		Практическое применение доступных функций модуля math.	2	Теоретико практическая			Практическое задание
46		Основы математических и тригонометрических функций в Python.	2	Теоретико практическая			Практическое задание
47		Особенности графического интерфейса пользователя (GUI) в программировании.	2	Практика			Тестирование
48		Использование модуля PyGame для разработки игр.	2	Практика			Практическое задание
49		Обзор основных функций PyGame.	2	Теоретико практическая			Практическое задание
50		Решение практических задач с использованием математических функций.	2	Практика			Практическое задание
51		Применение тригонометрических функций в практических задачах.	2	Теоретико практическая			Практическое задание
52		Создание графического интерфейса пользователя (GUI) для программы.	2	Практика			Практическое задание

53		Разработка GUI с использованием различных элементов и виджетов.	2	Практика			Практическое задание
54		Создание игры с использованием модуля PyGame.	2	Практика			Практическое задание
55		Программирование элементов игрового процесса с помощью PyGame.	2	Практика			Практическое задание
56		Создание игрового окружения и логики для простой игры.	2	Практика			Практическое задание
57		Тестирование и отладка GUI и игровых функций.	2	Практика			Практическое задание
58		Написание скриптов для решения математических задач с GUI.	2	Практика			Практическое задание
59		Разработка интерактивных графических приложений с математическими функциями.	2	Практика			Практическое задание
60		Программирование простой игры с использованием GUI элементов.	2	Практика			Практическое задание
61		Практические задания по созданию GUI, математическим функциям и разработке игр.	2	Теоретико-практическая			Самостоятельная работа
62	Защита проекта	Основы проектной деятельности: этапы, задачи, требования.	2	Теория			Устный опрос
63		Планирование проекта: определение целей, задач, сроков, ресурсов и бюджета.	2	Теоретико-практическая			Практическое задание
64		Процесс подготовки проектной работы: выбор темы, составление плана, разработка концепции.	2	Практика			Наблюдение
65		Работа над проектом.	2	Практика			Самостоятельная работа

66		Работа над проектом.	2	Практика			Самостоятельная работа
67		Работа над проектом.	2	Практика			Самостоятельная работа
68		Работа над проектом.	2	Практика			Самостоятельная работа
69		Основные принципы и правила защиты проектов.	2	Теория			Устный опрос
70		Подготовка команды для защиты проекта: распределение ролей, обучение участников.	2	Теоретико-практическая			Устный опрос
71		Техники и методы эффективного выступления при защите проекта.	2	Теоретико-практическая			Итоговое тестирование
72		Защита итогового проекта	2	Защита проекта			Защита проекта

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

В ходе обучения применяются разнообразные педагогические технологии, ориентированные на личность, которые нацелены на индивидуализацию обучения, учитывающие психологические особенности каждого ученика и способствующие его самостоятельности.

– личностноориентированные, которые представляют собой подход, в центре которого находится личность ученика. Основные принципы включают индивидуализацию обучения, учет психологических особенностей каждого ребенка и стимулирование его самостоятельности.

– коммуникативные. Это совокупность способов, механизмов и средств, используемых для сбора, обработки, хранения и передачи информации.

– информационнокоммуникационные. Это методы, производственные процессы и программнотехнические средства, которые применяют, чтобы собрать, обработать, сохранить, распространить, отобразить или использовать информацию в интересах пользователей.

– игровые, которые представляют собой группу методов и приемов организации педагогического процесса в форме различных педагогических игр, которая стимулирует познавательную активность детей, «провоцирует» их самостоятельно искать ответы на возникающие вопросы, позволяет использовать жизненный опыт детей, включая их обыденные представления.

Формы организации учебных занятий:

- лекция, объяснение, рассказ, демонстрация, игра
- беседа, дискуссия, практическая работа
- творческое задание
- техническое соревнование;
- индивидуальная защита проектов;
- творческая мастерская;
- творческий отчет;
- рефлексия

Методы образовательной деятельности:

В процессе обучения используются следующие методы:

- словесный,
- наглядный,
- практический,
- игровой,
- объяснительноиллюстративный,
- деятельностный.

Виды контроля/аттестации обучающихся:

- входной контроль не предусмотрен;
- текущий контроль осуществляется в форме наблюдения, демонстрация самостоятельных работ,
- промежуточная аттестация проводится в форме тестирования, демонстрация результата самостоятельной работы.

По итогам полного изучения программы проводится диагностика результативности освоения программы учащимися с целью определения степени освоения программы каждым ребёнком. В основе диагностики лежат оцениваемые параметры, результативность освоения программы делится на 3 уровня, выражающимися определённым количеством баллов: низкий 1 балл, средний 2 балла, высокий 3 балла (Приложение 1)

Оценочные материалы

Для оценки уровня усвоения материала учениками по окончании программы были разработаны материалы для педагогического мониторинга: промежуточные и итоговые тесты, практические задания, защита проектов и наблюдения.

Оценивание тестирования:

Для успешного прохождения проверочного тестирования и получения отметки "зачет" необходимо дать как минимум половину правильных ответов из всех вопросов. Таким образом, нижняя граница успешности составляет 50% правильных ответов от общего числа вопросов.

Оценивание устного опроса:

Опрос считается пройденным, если ответы учащегося соответствуют критериям: полнота и правильность ответа (учащийся дает развернутый ответ) степень осознанности, понимания изученного

Результат опроса не засчитывается, если обучающийся проявляет:

незнание большей части соответствующих вопросов, допускает ошибки в формулировке определений, искажающие их смысл, беспорядочно и неуверенно излагает материал.

Оценивание практических работ:

работа считается выполненной: учащийся самостоятельно выполняет работу, быстро и без ошибок, может самостоятельно исправляя ошибки в среднем темпе выполнить работу, может выполнять работу по образцу в медленном темпе
работа считается невыполненной – учащийся не может самостоятельно без помощи педагога выполнить работу, полное отсутствие умения применить знания на практике.

Оценивание самостоятельной работы учащихся:

Работа считается выполненной: учащийся успешно завершил задание, продемонстрировал способность самостоятельно применить полученные знания, навыки и умения, решив поставленную задачу без подсказок или дополнительной помощи. Работа выполнена точно, в соответствии с требованиями.

Работа считается невыполненной: учащийся не смог самостоятельно выполнить задание или применить знания на практике без посторонней помощи. Отсутствие самостоятельности, неспособность к самостоятельному решению задачи.

Оценивание проектных работ:

Оценивание выполнения проектных работ производится исходя из критериев, которые оцениваются определенным баллом. Успешная результативность определяется конкретной суммой баллов. В качестве нижней границы успешности выполнения проектной работы, соответствующей отметке «зачет», можно принять уровень 50% соответствия проектной работы заявленным критериям.

Материальнотехническое обеспечение

Информационное обеспечение: программное обеспечение – среда для программирования IDLE (Python) или аналог. При работе в дистанционном режиме – на рабочие места педагога и обучающихся должны быть предустановлены соответствующие сервисы.

Рекомендуемое учебное оборудование, рассчитанное на группу из 15 учащихся:

Презентационное оборудование

1. Интерактивная панель ICL infoRay 65" 1 шт.

Компьютерное оборудование

1. Персональный компьютер с доступом в сеть Интернет – 15 шт.

Программное обеспечение

1. Операционная система Windows 10 – 15 шт.
2. Офисный пакет программ для обработки текстовой, табличной, презентационной информации – 15 шт.
3. Visual Studio Code – 15 шт.
4. Интерпретатор Python – 15 шт.

ЛИТЕРАТУРА

Литература для педагога

1. Билл Любанович, «Простой Python. Современный стиль программирования», СПб.: Питер, 2021. – 592с.
2. Тарьк Зиаде «Python. Лучшие практики и инструменты», СПб.: Питер, 2021. – 560 с.
3. Агарева, О. Ю. Математическая логика и теория алгоритмов: – учеб. Пособие / О. Ю. Агарева, Ю. В. Селиванов. – М.: МАТИ, 2011 – 80 с.
4. Математика тулкит. Светлана Говор – М.: Фонд новых форм развития образования, 2018 –36 с.

Методические пособия для обучающихся:

1. Кольцов Д.М., Дубовик Е.В. Справочник PYTHON. Кратко, быстро, под рукой. М.: Наука и техника, 2021 – 288с.
2. Тимур Машнин «Создание настольных Python приложений с графическим интерфейсом пользователя», 2021, 140с.
3. Буйначев, С. К. Основы программирования на языке Python: учебное пособие / С. К. Буйначев, Н. Ю. Боклаг. – Екатеринбург: Издво Урал. унта, 2014. – 91, [1] с.
4. <https://python.org>

**Диагностика результативности освоения дополнительной общеразвивающей программы
«Основы программирования на языке Python»**

Показатели (оцениваемые параметры)	Критерии	Степень выраженности оцениваемого качества	Число баллов	Методы диагностики
Предметные результаты				
<i>Теоретическая подготовка</i>				
Теоретические знания (по основным разделам учебнотематического плана программы)	Соответствие теоретических знаний учащегося программным требованиям	Учащийся демонстрирует знание теории, но не применяет эти знания в практической деятельности	1	Наблюдение, тестирование
		Учащийся демонстрирует знание теории, применяет эти знания в практической деятельности	2	
		Учащийся демонстрирует знание теоретических фактов, применяет эти знания в практической деятельности, в незнакомых условиях	3	
Владение специальной терминологией в области программирования	Осмысленность и правильность использования терминологии по программированию и проектной деятельности	Учащийся знает отдельные специальные термины, но избегает их употреблять	1	Наблюдение, тестирование
		Учащийся сочетает специальную терминологию с бытовой	2	
		Учащийся специальные термины употребляет осознанно и в полном соответствии с их содержанием	3	
<i>Практическая подготовка</i>				
Практические умения и навыки,	Применение практических умений и навыков полученных	Учащийся демонстрирует полученные умения и навыки, но не применяет их в практической деятельности	1	Наблюдение, практическое задание

предусмотренные программой	при изучении программы	Учащийся способен частично продемонстрировать полученные умения и навыки, сомневается в применении их в практической деятельности	2	защита проекта
		Учащийся демонстрирует полученные умения и навыки, применяет в практической деятельности, в незнакомых условиях	3	
Владение специальным оборудованием и программным обеспечением	Отсутствие затруднений в использовании программ по программированию	Испытывает серьезные затруднения при работе с оборудованием	1	Наблюдение, практическое задание
		Работает с оборудованием с помощью педагога	2	
		Работает с оборудованием самостоятельно, не испытывает особых затруднений	3	
Умение подбирать и анализировать специальную техническую литературу и работать с информацией	Умение пользоваться технической литературой	Учащийся испытывает серьезные затруднения при выборе литературы на техническую тематику, нуждается в постоянной помощи и контроле педагога при работе с ней	1	Наблюдение
		Работает с технической литературой с помощью педагога/родителя, не проявляя самостоятельную заинтересованность в ее изучении	2	
		Умение подбирать и анализировать специальную техническую литературу и работать с информацией сбор, систематизацию, хранение информации	3	
Метапредметные результаты				
Различные виды мышления и способности	аналитическое, креативное и критическое мышление, изобретательность, образное и пространственное видение	Не способен или способен в очень незначительной степени самостоятельно осуществлять логические операции сравнения, анализа, обобщения, классификации, установления аналогий. Обладает рассеянным вниманием и слабой памятью.	1	наблюдение, практическое задание
		Не всегда самостоятельно осуществляет логические операции сравнения, анализа, обобщения, классификации, установления аналогий. Нуждается в помощи и контроле со стороны педагога. Внимание и память на среднем уровне.	2	
		Не испытывает никаких затруднений при осуществлении логических операций сравнения, анализа, обобщения, классификации, установления аналогий. Обладает хорошей кратковременной и долгосрочной памятью, внимателен, сосредоточен.	3	

Планирование деятельности	Умение планировать свою деятельность придерживаясь этапов	Учащийся постоянно не соблюдает план действий, не может выстроить последующие этапы для достижения цели	1	беседа
		Учащийся может периодически отклоняться от изначального плана действий, но достигает поставленной цели	2	
		Учащийся может составить план своих действий в начале работы и следовать ему до конца, до реализации цели	3	
Познавательная активность	включение в различные виды конкурсной и соревновательной деятельности	Не интересуется и не стремится к конкурсной деятельности по направлению	1	результаты конкурсов, соревнований
		Интерес к участию в конкурсах проявляет слабо. Участвует по рекомендации педагога	2	
		Учащийся интересуется и активно принимает участие в конкурсной деятельности	3	
Личностные результаты				
Мотивация	Обладание мотивацией к созданию собственных программных продуктов	Интерес к занятиям и конкурсной деятельности продиктован учащимся извне	1	наблюдение
		Интерес периодически поддерживается учащимся	2	
		Интерес к учебной и конкурсной деятельности постоянно поддерживается учащимся самостоятельно	3	

Командная работа	умение работать в команде, распределять обязанности между участниками команды в ходе реализации проекта	Проявляет готовность к общению, но редко выражает симпатию и доброжелательное отношение к команде, часто конфликтует. Избегает участия в общих делах	1	наблюдение
		Проявляет готовность общаться в команде, но сам проявляет инициативу лишь в некоторых ситуациях, иногда умеет договариваться, слушает не всегда внимательно. Участвует в общих делах при побуждении извне	2	
		Учащийся проявляет сам и поддерживает инициативу другого в общении, умеет договариваться, слушать, владеет навыками коммуникативного поведения. Инициативен в общих делах	3	

Низкий уровень: 10-16 баллов

Средний уровень: 17-23 балла

Высокий уровень: 24-30 баллов