

**МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
ГОРОДА РОСТОВА-НА-ДОНУ
«ДВОРЕЦ ТВОРЧЕСТВА ДЕТЕЙ И МОЛОДЕЖИ»**

ЦЕНТР ЦИФРОВОГО ОБРАЗОВАНИЯ ДЕТЕЙ «IT-куб»

Принято
педагогическим советом МБУ ДО ДТДМ
Протокол №1 от 31.08.2023 г.

Одобрено
методическим советом МБУ ДО ДТДМ
Протокол № 11 от 30.08.2023 г.

Утверждаю
Директор МБУ ДО ДТДМ

_____ Е.Э. Жихарцева

Приказ № 789 от 31.08. 2023 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА

«Практика программирования на Python (базовый уровень)»

Возрастная категория: 13-16 лет.

Срок реализации: 1 год.

Разработчик программы:

Литвинова Е.С.,

педагог дополнительного образования

Программу реализует:

Литвинова Е.С.,

педагог дополнительного образования

Методическое сопровождение:

Букатова Е.В., старший методист.

г. Ростов-на-Дону
2023 г.

ОГЛАВЛЕНИЕ

| | | |
|------|-------------------------------------|----|
| I. | ПОЯСНИТЕЛЬНАЯ ЗАПИСКА..... | 3 |
| II. | УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН..... | 8 |
| III. | СОДЕРЖАНИЕ ПРОГРАММЫ..... | 11 |
| IV. | МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ..... | 15 |
| V. | ДИАГНОСТИЧЕСКИЙ ИНСТРУМЕНТАРИЙ..... | 18 |
| VI. | СПИСОК ЛИТЕРАТУРЫ..... | 20 |
| VII. | ПРИЛОЖЕНИЯ..... | 24 |

I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Практика программирования Python (базовый уровень)» способствует приобщению обучающихся старшего школьного возраста к новейшим техническим, информационным технологиям посредством творческой и проектной деятельности.

Программа является практикоориентированной. Рекомендована к освоению обучающимся, изучающими курс по программе «Введение в программирование на Python», для отработки практических навыков.

Изучая программирование, обучающиеся получают глубокое понимание принципов работы компьютера, организации ввода, вывода и хранения информации, принципов построения диалоговых приложений, познают азы профессии программиста.

Современный период развития общества характеризуется масштабными изменениями в окружающем мире, влекущими за собой пересмотр социальных требований к образованию, предполагающими его ориентацию не только на усвоение обучающимся определенной суммы знаний, но и на развитие его личности, а также овладение метапредметными компетенциями. Большими возможностями в развитии школьников обладает подготовка в области информационных технологий и программирования. Программа отвечает потребностям общества, формированию творческих способностей и развитию личности.

Актуальность программы обусловлена потребностью общества в технически грамотных специалистах в области программирования, а также необходимостью повышения мотивации к выбору инженерных профессий и создания системы непрерывной подготовки будущих квалифицированных инженерных кадров, обладающих профессиональными компетенциями для развития отечественной науки и техники.

Отличительные особенности программы

Программа предназначена для учащихся, проявляющих повышенный интерес к программированию. Программа имеет практическую направленность с ориентацией на реальные потребности, соответствующие возрасту ученика.

Ключевым элементом обучения является проектная деятельность, которая ориентирована на использование знаний, умений и навыков, полученных в ходе обучения, для постановки и решения практических задач, которые носят прикладной характер. Она позволяет учащимся участвовать в создании конкретного результата и научиться работать в условиях ограниченного времени, под руководством заказчика, презентовать проект, а также обрести навыки профессиональной коммуникации с контрагентами.

Новизна программы заключается в следующем:

- программа интегрированная и построена с использованием межпредметных связей. Она объединяет в себе такие направления деятельности, как техническое моделирование и проектирование, современные компьютерные технологии. Обучающиеся работают с оборудованием и

программным обеспечением (Hard skills) и приобретают навыки, которые важны как для участия в командных проектах, так и для жизни в социуме (Soft skills);

- использование в учебном процессе проектных и исследовательских технологий способствует мотивации и приобретению нового опыта познавательной деятельности; использование в обучении уникального оборудования даёт возможность реального изготовления спроектированных моделей;

- в рамках программы созданы условия для развития навыков самообразования и исследования, построения индивидуальной траектории обучения, формирования познавательных интересов, интеллектуальной и ценностно-смысловой сферы обучающихся; предоставлены возможности участия в конкурсах, выставках и фестивалях различного уровня.

Педагогическая целесообразность программы обусловлена творческо-практической направленностью, которая является стратегически важным направлением в развитии и воспитании учащихся.

Посредством метода проектной работы и с использованием кейс-технологий в рамках базового модуля обучающиеся углубят свои навыки и базовые знания в сфере IT. Для реализации образовательной программы используются технологии развивающего, исследовательского и проектного обучения, которые обеспечивают выполнение поставленных целей и задач образовательной деятельности.

Программа имеет **техническую** направленность, ориентирована на развитие технических и творческих способностей и умений обучающихся, организацию их научно-исследовательской деятельности и профессионального самоопределения.

Тип программы - общеразвивающая.

Вид программы – модифицированная. Её содержание разработано в соответствии с требованиями актуальных нормативно-правовых документов в образовании: Национальной доктрины образования в Российской Федерации, Концепции дополнительного образования детей, Приказа Министерства просвещения РФ от 9 ноября 2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам», Концепции развития системы образования в Ростовской области на период до 2025 г. и др.

При разработке содержания данной программы использована дополнительная общеобразовательная программа: «Программирование на Python»/Степанов Д.И., Елисеева А.Г.

Уровень освоения – базовый.

Целью программы является привлечение детей к исследовательской и изобретательской деятельности через обучение программированию.

Задачи:

предметные:

- формирование умения организации собственной учебной деятельности;

- формирование умения использовать базовые понятия программирования при разработке компьютерных программ;
- формирование подхода к написанию программ: этика программирования, стиль программирования;
- владение методами, используемыми в программировании;
- владение основами языка программирования Python и принципами современного программирования;
- создание приложений на языке программирования Python;
- создание условий для получения первоначального практического опыта проектной работы.

метапредметные:

- обучение различным способам решения проблем творческого и поискового характера для дальнейшего самостоятельного создания способа решения проблемы;
- развитие образного, технического и аналитического мышления;
- формирование умения анализировать поставленные задачи, планировать и применять полученные знания при реализации творческих проектов;
- формирование навыков использования информационных технологий;

воспитательные:

- воспитание личностных качеств: самостоятельности, уверенности в своих силах, креативности;
- воспитать мотивацию учащихся к изобретательству, созданию собственных программных реализаций;
- привить стремление к получению качественного законченного результата в проектной деятельности;
- привить информационную культуру: ответственное отношение к информации с учетом правовых и этических аспектов её распространения, избирательного отношения к полученной информации;
- формировать правильное восприятие системы ценностей, принципов, правил информационного общества;
- формировать потребность в самостоятельном приобретении и применении знаний, потребность к постоянному саморазвитию;
- воспитывать социально-значимые качества личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность.

Прогнозируемые результаты освоения программы обучающимися по уровням

Предметные результаты:

- обучающийся знает и эффективно использует язык программирования Python;
- обучающийся понимает принципы и приёмы разработки программ на Python, умеет их отлаживать и запускать на выполнение;
- обучающийся умеет искать информацию в свободных источниках и

структурировать ее, а также комбинировать, видоизменять и улучшать идеи;

- обучающийся умеет: оперировать основными терминами, используемыми при разработке приложений; следовать требованиям; организовывать индивидуальное информационное пространство;

- обучающиеся приобретают навыки работы для создания собственных и групповых проектов.

Метапредметные результаты

- обучающийся находит решение проблемы;

- обучающийся использует различные источники информации: интернет, книги и журналы, мнение экспертов;

- обучающийся сотрудничает и оказывает взаимопомощь, доброжелательно и уважительно строит своё общение со сверстниками и взрослыми;

- обучающийся продуктивно участвует в проектной деятельности;

- обучающийся умеет самостоятельно планировать пути достижения целей, соотносит свои действия с планируемыми результатами, осуществляет контроль и коррекцию своей деятельности в процессе достижения результата.

Личностные результаты

- обучающийся во время обсуждения (беседы, мозгового штурма) выдвигает собственные идеи;

- обучающийся не нуждается в постоянной помощи педагога, умеет следовать инструкциям;

- обучающийся умеет работать в группе;

- обучающийся демонстрирует осведомлённость и интерес к программированию;

- обучающийся готов и способен к саморазвитию и личностному самоопределению, мотивирован к целенаправленной познавательной деятельности с целью приобретения профессиональных навыков в IT-сфере;

- обучающийся соблюдает ТБ, бережно относится к оборудованию и техническим устройствам.

Объем и срок освоения программы:

Содержание программы рассчитано на 72 часа. Срок освоения программы – 1 год.

Режим занятий: 1 раз в неделю по 2 часа, с учетом возрастных требований, всего 72 часа в год.

Формы занятий.

Обучение проводится в очной форме с возможностью использования дистанционных технологий.

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

Лекции с выполнением практического задания повторяемого учащимися на аналогичном оборудовании, обсуждения, практические занятия по решению заданий, аналогичных рассмотренным на лекции, метод кейсов и проектов, игровые и кибер-спортивные формы. Также программа курса включает

групповые и индивидуальные формы работы обучающихся (в зависимости от темы занятия).

По типу организации взаимодействия педагогов с обучающимися при реализации программы используются личностно-ориентированные технологии, технологии сотрудничества.

Реализация программы предполагает использование здоровьесберегающих технологий: создание безопасных технических условий, благоприятного психологического климата, наличие динамических пауз, периодическая смена деятельности.

Наполняемость группы: 12-15 человек.

Основной тип занятий – комбинированный, сочетающий в себе элементы теории и практики. Большинство заданий курса выполняется самостоятельно с помощью персонального компьютера и необходимых программных средств.

Адресат программы. Программа ориентирована на обучение детей с 13 – 16 лет. В группу принимаются учащиеся, у которых имеется желание дальнейшего развития по этой программе обучения.

Представленная программа рассчитана на любой социальный статус учащихся, имеющих различные интеллектуальные, технические, творческие способности.

Краткое описание возрастных психофизиологических особенностей детей, которым адресовано содержание программы.

Содержание программы разработано с учетом возрастных психофизиологических особенностей обучающихся подросткового и юношеского возраста.

В этом возрасте 13 – 16 лет у детей перестраиваются познавательные процессы (мышление, память, восприятие), которые позволяют успешно осваивать научные понятия и оперировать ими, что позволяет в рамках программы ставить перед детьми сложные задачи, а также использовать сложное оборудование, специализированные компьютерные программы. Учащиеся этого возраста, имеющие достаточную базовую подготовку уже интересуются конструированием, моделированием, созданием дизайна с применением компьютерных технологий, поэтому содержание программы адаптировано к данному возрасту.

II. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

| № п/п | Модуль программы | Часов всего | Количество часов | | Форма контроля |
|--|--|----------------|------------------|-----------|-------------------------------|
| | | | Теория | Практика | |
| Кейс №1 «Симулятор броска игральных костей» | | 12 | 5 | 7 | |
| 1 | Введение в курс. Инструктаж по ТБ. Знакомство со средой программирования PyCharm IDE. | 2 | 1 | 1 | тестирование |
| 2 | Основы ввода-вывода данных. Типы данных. | 2 | 1 | 1 | тестирование |
| 3 | Переменные. Строковые методы. | 2 | 1 | 1 | тестирование |
| 4 | Генерация псевдослучайных чисел. Цикл while. | 2 | 1 | 1 | тестирование |
| 5 | Реализация «Симулятора броска игральных костей». | 2 | 1 | 1 | демонстрация работы программы |
| 6 | Презентация результатов выполненного кейса | 2 | | 2 | презентация |
| Кейс №2 «Крестики-нолики» | | 12 | 2 | 10 | |
| 1 | Функции. Параметры и возвращаемые значения. | 2 | 1 | 1 | тестирование |
| 2 | Локальные и глобальные переменные. | 2 | 1 | 1 | тестирование |
| 3 | Создание плана игры «Крестики-нолики» | 2 | | 2 | Контрольные вопросы |
| 4 | Написание функций для игрока. | 2 | | 2 | тестирование |
| 5 | Написание функций для игры компьютера. Запуск и отладка игры. | 2 | | 2 | Демонстрация работы программы |
| 6 | Презентация результатов выполненного кейса | 2 | | 2 | презентация |
| Кейс №3 «Установи соответствие» | | 14 | 5 | 9 | |
| 1 | Что такое Tkinter. Основы Tkinter и создание окон. | 2 | 1 | 1 | тестирование |
| 2 | Виджеты в Tkinter. Кнопки, радиокнопки и флажки. Объект Canvas. | 2 | 1 | 1 | тестирование |
| 3 | Виджеты в Tkinter. Позиционирование виджетов. Методы Pack, Grid, Place | 2 | 1 | 1 | тестирование |
| 4 | Разработка приложения «Установи соответствие». Создание основного окна и импорт библиотек. | 2 | 1 | 1 | демонстрация работы программы |
| 5 | Разработка программы «Установи соответствие». Добавление символов, определение кнопок | 2 | 1 | 1 | демонстрация работы программы |
| 6 | Разработка программы «Установи соответствие». Показ символов. Завершение | 2 | | 2 | демонстрация работы программы |
| 7 | Презентация итогового результата кейса | 2 | | 2 | презентация |

| | | | | | |
|---|---|-----------|----------|----------|-------------------------------|
| Кейс №4 «Экранный питомец» | | 12 | 3 | 9 | |
| 1 | Экранный питомец. Создание тела, глаз и ушей. | 2 | 1 | 1 | Демонстрация работы программы |
| 2 | Экранный питомец. Создание рта, языка и щёк. | 2 | 1 | 1 | демонстрация работы программы |
| 3 | Экранный питомец. Создание эффекта моргания глаз. Создание эффекта забавного лица. | 2 | 1 | 1 | демонстрация работы программы |
| 4 | Экранный питомец. Создание эффектов весёлого и печального лица. | 2 | | 2 | демонстрация работы программы |
| 5 | Экранный питомец. Итоговый вариант. Отладка программы, устранение ошибок. | 2 | | 2 | демонстрация работы программы |
| 6 | Презентация итогового результата кейса. | 2 | | 2 | презентация |
| Кейс № 5 «PyGame. Классический space shooter» | | 14 | 6 | 8 | |
| 1 | PyGame. Основные сведения. Игровой цикл. Шаблон для создания игр. Работа с цветом. Работа со спрайтами | 2 | 1 | 1 | Демонстрация работы программы |
| 2 | Спрайт игрока и управление. Спрайты противников. Столкновения с игровыми объектами и выстрелы. | 2 | 1 | 1 | демонстрация работы программы |
| 3 | Добавление графики. Улучшение столкновений с игровыми объектами | 2 | 1 | 1 | демонстрация работы программы |
| 4 | Анимация спрайтов. Отображение счёта в игре. Звуковое и музыкальное оформление. | 2 | 1 | 1 | Демонстрация работы программы |
| 5 | Добавление защитного поля объекту игрока. Добавление анимации взрывов объектов астероидов. | 2 | 1 | 1 | Демонстрация работы программы |
| 6 | Добавление «полоски жизни» объекту игрока. Добавление выпадающих «бонусов», атаки, защиты объекта игрока. Добавление экрана завершения игры. Отладка и запуск итогового варианта кейса. | 2 | 1 | 1 | демонстрация работы программы |
| 7 | Презентация итогового результата кейса | 2 | | 2 | презентация |
| Проектный модуль «Креативное программирование на Python» | | 8 | 1 | 7 | |
| 1 | Проблематизация. Целеполагание. Поиск решения. Планирование. | 2 | 1 | 1 | Контрольные вопросы |
| 2 | Реализация замысла. Начальный этап. | 2 | | 2 | Контрольные вопросы |

| | | | | | |
|---|---|-----------|-----------|-----------|------------------------|
| 3 | Реализация замысла. Основной этап. | 2 | | 2 | Контрольные вопросы |
| 4 | Завершение проекта. Презентация. Рефлексия. | 2 | | 2 | презентация, рефлексия |
| | Итого: | 72 | 22 | 50 | |

III. СОДЕРЖАНИЕ ПРОГРАММЫ

Кейс №1 «Симулятор броска игральных костей»

Цель: знакомство с языком программирования Python и средами программирования на его основе. **Проблемные вопросы:** Какие существуют среды программирования на Python? Как осуществляется ввод-вывод данных? Перечислите типы данных Python? Как инициализировать переменную? Какие существуют методы работы со строками? Какие существуют циклы в Python?

Содержание: в кейсе рассматриваются начальные аспекты работы с языком программирования Python, и средой PyCharm IDE. Особенности создания программ на языке Python, способы их отладки и запуска. Основные типы данных, особенности инициализации переменных в Python. Методы работы со строками. Цикл с неизвестным количеством повторений.

Этапы:

I. Особенности и виды языков программирования, язык программирования Python, особенности разработки программ на Python, виды сред разработки, среда разработки PyCharm IDE. На этом этапе обучающиеся знакомятся с видами и особенностями языков программирования, в частности с особенностями языка программирования Python. Обучающиеся изучают особенности разработки на языке программирования Python.

II. Обучающиеся знакомятся с особенностями ввода-вывода данных на языке программирования Python, с понятием «переменная», понятием «цикл с известным числом повторений», «цикл с неизвестным числом повторений» применительно к языку программирования Python, типами данных и строковыми методами Python.

Soft skills:

- умение находить, анализировать и использовать информацию
- умение высказывать свою точку зрения

Hard skills:

- иметь представление о типах языков программирования
- иметь представление о языке Python и средах разработки на нём
- иметь основные навыки разработки программ в среде разработки PyCharm IDE
- умение использовать переменные в процессе разработки на Python
- умение использовать строковые методы в процессе разработки на Python

Кейс №2 «Крестики-нолики»

Цель: познакомить обучающихся с глобальными и локальными переменными, функциями и способами их реализации на языке программирования Python, а также с реализацией логики программ с искусственным интеллектом на этом языке.

Проблемные вопросы: что такое функция и как она реализуется на языке Python? Что такое переменные? Какие они бывают? Чем отличаются локальные переменные от глобальных?

Содержание: обучающимся предлагается разработать игру «Крестики-

нолики», в ходе которой они познакомятся с понятием «переменная», её типами, с понятием «функция» и особенностями их реализации на языке программирования Python. Дополнительно обучающиеся познакомятся с основами реализации алгоритмов искусственного интеллекта в ходе разработки данной игры.

Этапы:

I. Знакомство обучающихся с понятием «функция», её параметрами, передаваемыми параметрами и возвращаемыми значениями, а также с особенностями их реализации средствами языка программирования Python. Дополнительно обучающиеся знакомятся с понятием «переменная» и их типами: локальная переменная и глобальная переменная, а также особенностями реализации алгоритма искусственного интеллекта на примере игры «Крестики-нолики».

II. Реализация алгоритмов игры «Крестики-нолики» на языке программирования Python.

Soft skills:

- умение находить, анализировать и использовать информацию;
- умение высказывать свою точку зрения;
- умение работать в группе.

Hard skills:

- умение объявлять переменные на языке программирования Python;
- умение создавать функции, передавать в них параметры, а также обрабатывать возвращаемые значения;
- умение реализовывать алгоритм искусственного интеллекта, отвечающего действиям игрока средствами языка программирования Python.

Кейс №3 «Установи соответствие»

Цель: познакомить обучающихся с библиотекой создания графических интерфейсов пользователя (Graphic User Interface, GUI) на Python на примере создания приложения «Установи соответствие». **Проблемные вопросы:** как создаются окна в Tkinter? Что такое «Виджеты», виды виджетов, взаимное расположение виджетов? Как программно реализовать на Python виджеты кнопок?

Содержание: в рамках кейса обучающиеся создают приложение с графическим интерфейсом пользователя «Установи соответствие» на языке программирования Python. В процессе создания, обучающиеся знакомятся с основными элементами библиотеки Tkinter.

Этапы:

I. Знакомство с библиотекой Tkinter и её основными элементами, виджетами и их видами.

II. Программная реализация алгоритма приложения «Установи соответствие» на языке программирования Python.

Soft skills:

- умение находить, анализировать и использовать информацию;
- умение высказывать свою точку зрения;
- умение работать в группе;

Hard skills:

- умение создавать графические интерфейсы пользователя, используя средства библиотеки Tkinter
- умение составлять и отлаживать программы;

Кейс №4 Экранный питомец.

Цель: познакомить обучающихся с методами объекта Canvas, и его основными методами, через создание графического приложения «Экранный питомец»

Проблемные вопросы: как при помощи объекта Canvas отрисовать тело, уши и глаза питомца? Каким образом осуществляется отрисовка глаз, рта и щёк питомца? Как осуществить реализацию эмоций экранного питомца? Как осуществляется эффект моргания глаз, показа языка, отражения печального, нормального и весёлого лиц экранного питомца?

Содержание: в рамках кейса обучающиеся создают графическое приложение «Экранный питомец» на языке программирования Python. В процессе создания приложения, обучающиеся знакомятся с объектом Canvas, его основными функциями и методами.

Этапы:

I. Программная реализация алгоритма работы приложения «Экранный питомец» на языке программирования Python.

Soft skills:

- умение находить, анализировать и использовать информацию;
- умение высказывать свою точку зрения;
- умение работать в группе

Hard skills:

- умение работать с объектом Canvas средствами языка программирования Python;
- умение создавать и отлаживать программы.

Кейс №5 «PyGame. Классический space shooter»

Цель: познакомить обучающихся с библиотекой PyGame для создания компьютерных игр на языке Python

Проблемные вопросы: как реализуется основной игровой цикл в PyGame? Как в PyGame осуществляется работа с игровыми спрайтами, цветом? Каким образом в игре реализовано столкновение объекта игрока с другими игровыми объектами? Как подсчитываются очки в игре? Как реализовано выпадение «бонусов», повышающих атакующие и защитные возможности спрайта-игрока?

Содержание: в рамках кейса обучающиеся создают 2D-игру «Классический space shooter», используя возможности и средства языка программирования Python и библиотеки PyGame. **Этапы:**

I. Программная реализация алгоритма игры «Классический space shooter»

Soft skills:

- умение находить, анализировать и использовать информацию;

- умение высказывать свою точку зрения;
- умение работать в группе

Hard skills:

- умение работать с библиотекой PyGame;
- умение создавать и отлаживать программы.

Проектный модуль «Креативное программирование на Python»

Цель: разработать и создать приложение на языке программирования Python.

Проблемные вопросы: как эффективно выстроить работу по созданию приложения?

Содержание: обучающимся предлагается создать собственное приложение на языке программирования Python, соответствующее одной из групп: образовательное, игровое, инновационное. Для обучающихся 13-17 лет можно предложить примерные темы проектов для самостоятельной разработки.

Примерный перечень проектов:

1. Программа для тестирования знаний по выбранному предмету.
2. Приложение-тренажёр для отработки каких-либо навыков.
3. Игровое приложения в виде текстового квеста или игры.

В процессе работы, обучающиеся знакомятся с основами Scrum-метода организации эффективной работы над проектом.

Этапы:

I. Обучающиеся организуются в команды, проводят мозговой штурм, выбирают тему проекта. Проводят поиск информации, целеполагание и планирование, проектируют сюжет/сценарий исполнения проекта, распределяют работу по проекту

II. Обучающиеся создают сценарий приложения.

III. Обучающиеся создают программный код, проводят тестирование и отладку приложения.

IV. Обучающиеся готовят презентацию проекта и демонстрируют все возможности приложения.

V. Проведение рефлексии. Работа с листом самооценивания.

IV.МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Дополнительная общеобразовательная общеразвивающая программа «Практика программирования Python (базовый уровень)» разработана с учетом современных требований к организации образовательного процесса.

Программа основана на постулатах личностно-ориентированного образования, особенностью которого является актуализация гуманитарных, человекообразующих функций обучения: сохранение и восстановление телесного и духовного здоровья, личной свободы, нравственности; гуманный подход направлен не на формирование ребенка, а на обеспечение поддержки в его развитии, создание механизмов саморегуляции, саморазвития, адаптации, самозащиты, самовоспитания личности; акцентирование внимания не на вооружении ребенка знаниями и умениями, а его становлении, обретении им неповторимой индивидуальности, духовности, творческого начала; построение учебного процесса осуществляется на основе образовательных технологий, основными признаками которых является диалогичность, деятельностный творческий характер, поддержка индивидуального развития ребенка, предоставление ему необходимого пространства свободы, творчества и др.

При реализации программы используются следующие 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 ребёнка, 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, партнёрство в отношениях педагога и обучающегося, совместная выработка цели, содержания, оценивание, 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.

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.

– Конкретные навыки. Кейс-метод позволяет совершенствовать «гибкие навыки» (soft skills), которым не учат в университете, но которые оказываются крайне необходимы в реальном рабочем процессе.

В ходе работы над кейсом целесообразно использовать следующие методы, приемы, средства и формы организации, внесённые в таблицу.

| № | Формы организации | Методы и приемы | Дидактический материал | Формы контроля |
|----------|--|---|---|--|
| 1 | Лекция с разбором решения практического задания | устное изложение с визуальным рядом, позволяющий в доступной форме донести до обучающихся обучающийся материал; | Видео, приложения, шаблоны файлов, веб-доски и веб-плакаты. | Проверка синхронного выполнения материала лекции. |
| 2 | Обсуждение, рефлексия рассмотренных тем | устный опрос в ходе демонстрации видеоряда | Видео-презентация | рефлексивный самоанализ, контроль и самооценка обучающихся |
| 3 | Практическое задание, сходное с разбиравшимся на лекции; | репродуктивный практический метод; частично-поисковый | Видео, шаблоны исходных файлов, веб-плакаты | Просмотр хода выполнения; обсуждение итогов |
| 4 | Проект | исследовательский метод; практический метод частично-поисковый | Веб-доски и веб-документы, видео, инструкции по работе над проектом, шаблоны файлов | Защита проекта |
| 5 | Соревнование | практический метод | Веб-доски и веб-документы | Подведение итогов. |
| 6 | Исследование | исследовательский метод | Презентация, видео, описание хода исследования и т.д. | Конференция |

Учебно-методический комплекс программы

Для реализации программы используются:

– авторские учебные видео и презентации, раздаточный материал к каждой теме;

– специализированная литература по программированию, подборка журналов;

- наборы технической документации к применяемому оборудованию;
- образцы игровых приложений, выполненных обучающимися и педагогом,
- плакаты, фото и видеоматериалы,
- учебно-методические пособия для педагога и обучающихся, включающие дидактический, информационный, справочный материалы на различных носителях, компьютерное и видео оборудование.

V. ДИАГНОСТИЧЕСКИЙ ИНСТРУМЕНТАРИЙ

Система контроля знаний и умений обучающихся представляется в виде учёта результатов по итогам выполнения практических заданий и посредством наблюдения, отслеживания динамики развития учащегося.

Критерии и способы определения результативности

Результативность отслеживается методом анализа практических и творческих работ, участия в мероприятиях (викторинах, выставках, олимпиадах).

Оцениваются следующие аспекты:

- идентификация (определение) проблемы;
- целеполагание и планирование деятельности;
- применение технологий;
- планирование ресурсов;
- оценка деятельности;
- оценка результатов (продукта) деятельности;
- рефлексия.

Психолого-педагогический мониторинг результатов образовательного процесса

Психолого-педагогический мониторинг, или текущий контроль, – это систематическая оценка уровня освоения дополнительной программы в течение учебного года.

Текущий контроль складывается из следующих компонентов.

Входная диагностика знаний. В начале учебных занятий педагогом проводится входная диагностика для определения начального уровня Hard skills и Soft skills.

Оперативный контроль усвоения материала осуществляется по завершению изучения каждого кейса с помощью контрольных вопросов, мини-конференций по защите проектов, внутригрупповой конкурс (соревнования), презентаций (самопрезентация) проектов, творческой работы или тестирования.

Итоговая аттестация по программе:

- решение практического задания, основанного на рассмотренном ранее материале;
- защита индивидуального или группового проекта;
- выставка;
- межгрупповые соревнования;
- проведение промежуточного и итогового тестирования;
- взаимооценка обучающимися работ друг друга.

Итоговая оценка развития личностных качеств воспитанника производится по трём уровням:

- «высокий»: положительные изменения личностного качества воспитанника в течение учебного года признаются как максимально возможные для него;
- «средний»: изменения произошли, но воспитанник потенциально был способен к большему;

– «низкий»: изменения не замечены.

Результатом усвоения обучающимися программы по каждому уровню являются: устойчивый интерес к занятиям программированием, результаты достижений в массовых мероприятиях различного уровня.

Для усвоения курса и повышения общей эрудированности обучаемых ключевые понятия **hardskills** модуля и значимую фактологическую информацию предлагать в виде домашних контрольных на самостоятельную подготовку обучающимся.

Для этих целей программа имеет **дистанционную поддержку**, которая реализуется с помощью веб-сервиса Google Класс.

VI. СПИСОК ЛИТЕРАТУРЫ

Нормативная литература

1. Конституция Российской Федерации (принята всенародным голосованием 12.12.1993 с изменениями, одобренными в ходе общероссийского голосования 01.07.2020);
2. Федеральный Закон от 29.12.2012г. № 273-ФЗ «Об образовании в Российской Федерации» (ред.17.02.2023);
3. Федеральный закон РФ от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребенка в Российской Федерации» (с изменениями от 29.12.2022);
4. Концепция развития дополнительного образования детей до 2030г., утвержденная распоряжением Правительства Российской Федерации от 31.03.2022г. №678-р;
5. Стратегия развития воспитания в РФ на период до 2025 года (распоряжение Правительства РФ от 29.05.2015 г. № 996-р);
6. Государственная программа Российской Федерации «Развитие образования» на 2019-2025 г, утвержденная Постановлением Правительства Российской Федерации от 26 декабря 2017 года № 1642 (ред. от 15.03.2021).
7. Приоритетный проект «Доступное дополнительное образование для детей», утвержденный 30.11.2016г. протоколом заседания президиума при Президенте РФ (в ред.27.09.2017);
8. Федеральный проект "Успех каждого ребенка" (утв. на заседании проектного комитета по национальному проекту "Образование" 07.12.2018 г, пр. 3);
9. Приказ Министерства просвещения РФ от 27.07.2022г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
10. Приказ Министерства просвещения Российской Федерации №467 от 03.09.2019 г. «Об утверждении Целевой модели развития региональных систем дополнительного образования».
11. Распоряжение Министерства просвещения Российской Федерации №Р-126 от 21.06.2021 г. «Об утверждении ведомственной целевой программы «Развитие дополнительного образования детей, выявление и поддержка лиц, проявивших выдающиеся способности».
12. Конвенция о правах ребенка (принята резолюцией 44/25 Генеральной Ассамблеи от 20 ноября 1989 г.) — URL: http://www.un.org/ru/documents/decl_conv/conventions/childcon.shtml.
13. Национальный проект «Образование», утвержденный на заседании президиума Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам (протокол от 24 декабря 2018 г. № 16).
14. Приказ Министерства просвещения Российской Федерации от 9 ноября 2018 г. № 196 (ред. от 30.09.2020 г.) «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

15. Приказ Министерства просвещения Российской Федерации от 30 сентября 2020 г. № 533 «О внесении изменений в Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденный приказом Министерства просвещения Российской Федерации от 9 ноября 2018 г. № 196».
16. Приказ Министерства труда и социальной защиты Российской Федерации от 22 сентября 2021 г. N 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых».
17. Распоряжение Правительства Российской Федерации от 23.01.2021г. № 122-р «Об утверждении Плана основных мероприятий, проводимых в рамках Десятилетия детства, на период до 2027 года.
18. Стратегическая инициатива «Новая модель системы дополнительного образования», одобренная Президентом Российской Федерации 27 мая 2015 г
19. Стратегия развития воспитания в Российской Федерации на период до 2025 года, утвержденная Распоряжением Правительства Российской Федерации от 29 мая 2015 г. № 996-р.
20. Указ Президента Российской Федерации от 29 мая 2017 г. № 240 «Об объявлении в Российской Федерации Десятилетия детства».
21. Указ Президента Российской Федерации от 21 июля 2020 г. № 474 «О национальных целях и стратегических задачах развития Российской Федерации на период до 2030 года».
22. Федеральный закон от 29.12.2012 N 273-ФЗ (ред. от 24.03.2021) «Об образовании в Российской Федерации».
23. Федеральный проект «Успех каждого ребенка», утвержденный президиумом Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам (протокол от 3 сентября 2018 года № 10).
24. Приказ Министерства образования и науки РФ от 23.08.2017 г. № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
25. Приказ Министерства просвещения Российской Федерации от 03.09.2019 № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей» (в ред. от 02.02.2021г.);
26. Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
27. Постановление Главного государственного санитарного врача РФ от 28.01.2021 № 2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» (рзд.VI. Гигиенические нормативы по устройству, содержанию и режиму работы организаций воспитания и обучения, отдыха и оздоровления детей и молодежи»);

28. Распоряжение правительства Российской Федерации от 29 ноября 2014г. № 2403р. «Основы государственной молодежной политики Российской Федерации на период до 2025 года.»;
29. Государственная программа Ростовской области «Развитие образования», утверждена постановлением Правительства Ростовской области от 17.10.2018 № 646 (с изменениями на 28 декабря 2020 года).
30. Методические рекомендации по проектированию дополнительных общеобразовательных программ (письмо Минобрнауки России от 18 ноября 2015 г. № 09-3242);
31. Областной Закон Ростовской области от 14.11.2013 №26-ЗС «Об образовании в Ростовской области»;
32. Локальные акты МБУ ДО ДТДМ: Устав, Учебный план, Положения о структурных подразделениях, Правила внутреннего трудового распорядка, инструкции по технике безопасности.

Основная литература

1. Гин, А.А. Приёмы педагогической техники: свобода выбора, открытость, деятельность, обратная связь, идеальность: Пособие для учителей / А.А. Гин. – Гомель : ИПП «Сож», 1999. – 88 с.
2. Григорьев, Д.В. Внеурочная деятельность школьников. Методический конструктор: пособие для учителя / Д.В. Григорьев, П.В. Степанов. – М.: Просвещение, 2011. – 223 с. – (Стандарты второго поколения).
3. Доусон М. Програмируем на Python. – СПб.: Питер, 2014 г. – 416 с
4. Келли С. Разработка игр на PyGame, Raspberry Pi – 2016г. – 200 с
5. Эрик Мэтиз. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. – СПб: Питер, 2020 – 512 с
6. Буйлова, Л.Н. Методические рекомендации по подготовке авторских программ дополнительного образования детей / Л.Н. Буйлова, Н.В. Кленова, А.С. Постников [Электронный ресурс] / Дворец творчества детей и молодежи. В помощь педагогу. – Режим доступа: <http://doto.ucoz.ru/metod/>.
7. Конасова, Н.Ю. Оценка результатов дополнительного образования детей. ФГОС. / Н.Ю. Конасова. - Волгоград: Учитель, 2016. – 121с. – (Образовательный мониторинг).
8. Кучма, В.Р. Гигиена детей и подростков при работе с компьютерными видеодисплейными терминалами. / В.Р. Кучма. - М. : Медицина, 2000. - 160 с.
9. Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы). Письмо Департамента государственной политики в сфере воспитания детей и молодежи Министерства образования и науки РФ № 09-3242 от 18.11.2015 г. [Электронный ресурс] / Самарский дворец детского и юношеского творчества. – Режим доступа: pioner-samara.ru/sites/default/files/docs/metodrek_dop_rf15.doc.
10. Методические рекомендации по разработке дополнительных

общеобразовательных программ. Письмо Министерства образования и науки Самарской области от 03.09.2015 г. № МО-16-09-01/826-ту [Электронный ресурс] / Самарский дворец детского и юношеского творчества. - Режим доступа: <http://pioner-samara.ru/content/metodicheskaya-deyatelnost>.

11. Фомичева, О.С. Воспитание успешного ребенка в компьютерном веке. / О.С. Фомичева. –

М.: Гелиос АРВ, 2000. -192 с.

12. Официальный обучающий ресурс по языку программирования Python <https://docs.python.org/3/tutorial/index.html> [Электронный ресурс] – 14.06.2021

13. Обучающий ресурс по разработке игровых приложений средствами библиотеки PyGame <https://kidscancode.org/lessons/> [Электронный ресурс] – 14.06.2021.

VII. ПРИЛОЖЕНИЕ

Приложение 1

Календарно-тематический план

| № п/п | Тема занятия | Часов всего | Количество часов | | Форма контроля |
|-------|---|-------------|------------------|----------|-------------------------------|
| | | | Теория | Практика | |
| 1 | Введение в курс. Знакомство со средой программирования PyCharm IDE. | 2 | 1 | 1 | тестирование |
| 2 | Основы ввода-вывода данных. Типы данных. | 2 | 1 | 1 | тестирование |
| 3 | Переменные. Строковые методы. | 2 | 1 | 1 | тестирование |
| 4 | Генерация псевдослучайных чисел. Цикл while | 2 | 1 | 1 | тестирование |
| 5 | Реализация «Симулятора броска игральных костей» | 2 | 1 | 1 | демонстрация работы программы |
| 6 | Презентация результатов выполненного кейса | 2 | | 2 | презентация |
| 7 | Функции. Параметры и возвращаемые значения. | 2 | 1 | 1 | тестирование |
| 8 | Локальные и глобальные переменные. | 2 | 1 | 1 | тестирование |
| 9 | Создание плана игры «Крестики-нолики». | 2 | | 2 | контрольные вопросы |
| 10 | Написание функций для игрока. | 2 | | 2 | тестирование |
| 11 | Написание функций для игры компьютера. Запуск и отладка игры. | 2 | | 2 | демонстрация работы программы |
| 12 | Презентация результатов выполненного кейса. | 2 | | 2 | презентация |
| 13 | Что такое Tkinter. Основы Tkinter и создание окон. | 2 | 1 | 1 | тестирование |
| 14 | Виджеты в Tkinter. Кнопки, радиокнопки и флажки. Объект Canvas. | 2 | 1 | 1 | тестирование |
| 15 | Виджеты в Tkinter. Позиционирование виджетов. Методы Pack, Grid, Place. | 2 | 1 | 1 | тестирование |
| 16 | Разработка программы «Установи соответствие». Создание основного окна и импорт библиотек. | 2 | 1 | 1 | демонстрация работы программы |

| | | | | | |
|----|--|---|---|---|-------------------------------|
| 17 | Разработка программы «Установи соответствие». Добавление символов, определение кнопок | 2 | 1 | 1 | демонстрация работы программы |
| 18 | Разработка программы «Установи соответствие». Показ символов. Завершение. | 2 | | 2 | демонстрация работы программы |
| 19 | Презентация итогового результата кейса. | 2 | | 2 | презентация |
| 20 | Экранный питомец. Создание тела, глаз и ушей | 2 | 1 | 1 | демонстрация работы программы |
| 21 | Экранный питомец. Создание рта, языка и щёк. | 2 | 1 | 1 | демонстрация работы программы |
| 22 | Экранный питомец. Создание эффекта моргания глаз, забавного лица. | 2 | 1 | 1 | демонстрация работы программы |
| 23 | Экранный питомец. Создание эффектов весёлого и печального лица. | 2 | | 2 | демонстрация работы программы |
| 24 | Экранный питомец. Итоговый вариант. Отладка программы, устранение ошибок. | 2 | | 2 | демонстрация работы программы |
| 25 | Презентация итогового результата кейса. | 2 | | 2 | презентация |
| 26 | PyGame: основные сведения; игровой цикл; шаблон для создания игр; работа с цветом, со спрайтами. | 2 | 1 | 1 | демонстрация работы программы |
| 27 | Спрайты: игрока и управление; противников. Столкновения с игровыми объектами и выстрелы. | 2 | 1 | 1 | демонстрация работы программы |
| 28 | Добавление графики. Улучшение столкновений с игровыми объектами. | 2 | 1 | 1 | демонстрация работы программы |
| 29 | Анимация спрайтов. Отображение счёта в игре. Звуковое и музыкальное оформление. | 2 | 1 | 1 | демонстрация работы программы |
| 30 | Добавление защитного поля объекту игрока; анимации взрывов объектов астероидов. | 2 | 1 | 1 | демонстрация работы программы |
| 31 | Добавление «полоски жизни» объекта игрока; выпадающих «бонусов», атаки, защиты объекта игрока; экрана завершения игры. Итоговый вариант кейса. | 2 | 1 | 1 | демонстрация работы программы |

| | | | | | |
|---------------|--|-----------|-----------|-----------|---------------------|
| 32 | Презентация итогового результата кейса. | 2 | | 2 | презентация |
| 33 | Проблематизация. Целеполагание. Поиск решения. Планирование. | 2 | 1 | 1 | Контрольные вопросы |
| 34 | Реализация замысла. Начальный этап. | 2 | | 2 | Контрольные вопросы |
| 35 | Реализация замысла. Основной этап. | 2 | | 2 | Контрольные вопросы |
| 36 | Завершение проекта. Презентация. Рефлексия. | 2 | | 2 | презентация |
| Итого: | | 72 | 22 | 50 | |

Диагностика знаний

Диагностическая карта

| № /п | Имя | Входящая Диагностика | Промежуточная диагностика | Итоговая диагностика |
|------|-----------|----------------------|---------------------------|----------------------|
| 1 | | | | |
| 2 | | | | |
| 3+ | | | | |
| | Итого в % | Н С В | | Н С В |

Результативность отслеживается с помощью карты наблюдений, анализа участия детей в совместной продуктивной деятельности, разработки и защите творческих проектов. Усвоение программы возможно по 3-м уровням: низкий (Н), средний (С), высокий (В).

Низкий уровень

Учащиеся должны знать / понимать:

- технику безопасного поведения во время занятий;
- правила поведения в общественных местах,
- понятие программы Python; общую структуру программы;
- основные типы данных;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- основные циклы с условием;
- основные правила записи циклов условием;
- формат записи цикла с параметром;
- понятие функции;
- основные принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- операции со строками;
- способ описания списка;
- способ описания кортежа;
- способ описания словаря;
- основные операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.
- что такое стиль программирования;
- правила именования объектов;

- основные рекомендации при написании программ.
- Учащиеся должны уметь:*
- понимать учебную задачу, сохранять ее содержание в процессе ее выполнения под руководством педагога;
- работать в паре, малой группе;
- выполнить установку программы под руководством педагога;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе под руководством педагога;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием под руководством педагога;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи по наводящим вопросам педагога;
- создавать и использовать основные функции;
- описывать и соединять строки;
- находить подстроку в строке с помощью педагога;
- находить количество слов в строке;
- вводить и выводить элементы списка под руководством педагога;
- приводить примеры использования вложенных списков (матриц) по наводящим вопросам педагога;
- описывать множества под руководством педагога;
- определять принадлежность элемента множеству по наводящим вопросам;
- определять вид ошибок и находить ошибки в программе под руководством педагога;
- составлять элементарные алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python под руководством педагога.

Средний уровень

Учащиеся должны знать / понимать:

- основные положения техники безопасности на занятиях, правила поведения в общественных местах, правила дорожной безопасности;
- основные приемы взаимодействия в группе сверстников;
- понятие программы;
- структуру программы на Python;
- режимы работы с Python.
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- циклы с условием и их виды;
- назначение и особенности использования цикла с параметром;

- формат записи цикла с параметром;
- примеры использования циклов различных типов.
- понятие функции;
- основные способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками;
- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.
- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ;
- основные шаги работы над проектом, его презентации.

Учащиеся должны уметь:

- уважительно относиться к преподавателям и сверстникам;
- применять некоторые приемы логического (абстрактное) мышления;
- концентрировать внимание на одном или двух объектах;
- понимать причины успеха/неуспеха с помощью анализа педагога;
- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи;
- создавать и использовать функции;
- использовать механизм параметров для передачи значений;
- описывать строки;
- соединять строки;
- находить длину строки;

- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке;
- описывать списки;
- вводить элементы списка;
- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.
- определять вид ошибок и находить ошибки в программе.
- составлять алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- отлаживать и тестировать программы, написанные на языке Python;
- понимать и сохранять цели и задачи учебной деятельности в ходе занятия;
- планировать свою деятельность с помощью взрослого;
- сотрудничать с взрослыми и сверстниками в процессе выполнения учебной задачи;
- понимать причины успеха/неуспеха учебной деятельности под руководством взрослого;
- делать выводы в ходе полученных заданий; выстраивать логические цепи рассуждений под руководством педагога;
- выражать творческие идеи, разработать творческий проект на основе образца;
- конструктивно взаимодействовать в составе группы в ходе работы над проектом.

Высокий уровень

Учащиеся должны знать / понимать:

- основные положения техники безопасности на занятиях, правила поведения в общественных местах, правила дорожной безопасности, правила поведения во время чрезвычайных происшествий;
- приемы конструктивного взаимодействия в группе сверстников;
- понятие программы;
- структуру программы на Python;
- режимы работы с Python.
- общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;

- логические операторы or, and, not;
- циклы с условием и их виды;
- правила записи циклов с условием;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.
- понятие функции;
- способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками;
- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.
- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ;
- правила и этапы работы над проектом;
- приемы успешной презентации проекта.

Учащиеся должны уметь:

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи;
- создавать и использовать функции;
- использовать механизм параметров для передачи значений;
- описывать строки;
- соединять строки;

- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке;
- описывать списки;
- вводить элементы списка;
- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.
- определять вид ошибок и находить ошибки в программе.
- составлять алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- отлаживать и тестировать программы, написанные на языке Python;
- понимать учебную задачу, анализировать достижение результата;
- делать выводы в ходе полученных заданий; самостоятельно выстраивать логические цепи рассуждений;
- понимать причины успеха/неуспеха, конструктивно принимать причины неуспеха, исправлять ошибки с минимальным участием педагога;
- планировать свою деятельность, находить оригинальные способы выполнения поставленной творческой задачи;
- создать проектировочную команду и организовать ее деятельность;
- разработать творческий проект по предложенной или самостоятельно выбранной теме в составе творческой группы;
- самостоятельно подготовить презентацию и защитить проект.

Низкий уровень: 0-70% выполненных заданий;

Средний уровень: 70-85% правильно выполненных заданий;

Высокий уровень: 85-100% правильно выполненных заданий.

Входящий контроль

1. Назовите два числа, у которых количество цифр равно количеству букв, составляющих название каждого из этих чисел.

| | |
|--|--|
| | |
|--|--|

2. Собака была привязана к десятиметровой веревке, а прошла двести метров. Как ей это удалось?

| |
|--|
| |
|--|

3. Есть дорога, по которой может проехать только одна машина. По дороге едут две машины: одна с горы, другая под гору. Как им разъехаться?

| |
|--|
| |
|--|

4. Как с помощью только одной палочки образовать на столе треугольник?

| |
|--|
| |
|--|

5. В каком месяце болтливая девочка говорит меньше всего?

| |
|--|
| |
|--|

6. Что становится на треть больше, если его поставить вверх ногами?

| |
|--|
| |
|--|

7. Представьте себе, что вы кондуктор. Поезд везет сто вагонов, в каждом вагоне 10 купе, в каждом купе 4 пассажира. Сколько лет кондуктору?

| |
|--|
| |
|--|

8. Юра разрезал огромную пиццу на 10 кусков. Затем он взял один из кусков и разрезал его еще на 10. После этого из имеющихся кусков он выбрал два и разрезал каждый из них на 10.

9. Вопрос: Сколько кусков пиццы получилось у Юры.

| |
|--|
| |
|--|

10. В компьютерной игре нужно победить монстра. Изначально у Юры было только 9 выстрелов. Но за каждое попадание он получал дополнительно еще 3 выстрела.

Вопрос: Сколько раз Юра попал в цель, если всего он выстрелил 30 раз, израсходовав все выстрелы?

| |
|--|
| |
|--|

**Вопросы для промежуточного контроля по усвоению
материала**

| |
|--|
| <p>Составьте выражение для вычисления в интерпретаторе Python 3 и вставьте в поле ответа результат вычисления:</p> <p style="text-align: center;">11111·1111111</p> <p>— произведение чисел 11111 (5 единиц) и 1111111 (7 единиц)</p> |
| <p>Запишите число 1.2345e3 в виде десятичной дроби.</p> |
| <p>Составьте и запишите выражение для вычисления:</p> <p style="text-align: center;">2014.0¹⁴(Возвестив 14 степень)</p> <p>Обратите внимание на запись числа: это вещественное число.</p> |
| <p>Приведите к целому типу число 2.99</p> |
| <p>Расставьте скобки в выражении</p> <p style="text-align: center;">a and b or not a and not b</p> <p>в соответствии с порядком вычисления выражения (приоритетом операций). Всего потребуется 5 пар скобок (внешние скобки входят в их число).</p> |
| <p>Найдите результат выражения для заданных значений <i>a</i> и <i>b</i></p> <p>Учитывайте регистр символов при ответе.</p> <p>a = True b = False a and b or not a and not b</p> |
| <p>Отметьте выражения, значения которых равны True:</p> <ul style="list-style-type: none">• "239" < "30" and 239 < 30• "239" < "30" and 239 > 30• "239" > "30" and 239 < 30• "239" > "30" and 239 > 30 |
| <p>Укажите результат выражения: "123" + "42"</p> |
| <p>Какое значение будет у переменной <i>i</i> после выполнения фрагмента программы?</p> <pre>i = 0 while i <= 10: i = i + 1 if i > 7: i = i + 2</pre> |

Сколько итераций цикла будет выполнено в этом фрагменте программы?

```
i = 0  
while i <= 10:  
    i = i + 1  
    if i > 7:  
        i = i + 2
```

Сколько всего знаков * будет выведено после исполнения фрагмента программы:

```
i = 0  
while i < 5:  
    print('*')  
    if i % 2 == 0:  
        print('*')  
    if i > 2:  
        print('***')  
    i = i + 1
```

Определите, какое значение будет иметь переменная i после выполнения следующего фрагмента программы:

```
i = 0  
s = 0  
while i < 10:  
    i = i + 1  
    s = s + i  
    if s > 15:  
        break  
    i = i + 1
```

Определите, какое значение будет иметь переменная i после выполнения следующего фрагмента программы:

```
i = 0  
s = 0  
while i < 10:  
    i = i + 1  
    s = s + i  
    if s > 15:  
        continue  
    i = i + 1
```

Задачи для промежуточного контроля по усвоению материала

Напишите простой калькулятор, который считывает с пользовательского ввода три строки: первое число, второе число и операцию, после чего применяет операцию к введённым числам ("первое число" "операция" "второе число") и выводит результат на экран.

Поддерживаемые операции: +, -, /, *, mod, pow, div, где mod — это взятие остатка от деления, pow — возведение в степень, div — целочисленное деление.

Если выполняется деление и второе число равно 0, необходимо выводить строку "Деление на 0!".

Обратите внимание, что на вход программе приходят вещественные числа.

Жители страны Малевии часто экспериментируют с планировкой комнат. Комнаты бывают треугольные, прямоугольные и круглые. Чтобы быстро вычислять жилплощадь, требуется написать программу, на вход которой подаётся тип фигуры комнаты и соответствующие параметры, которая бы выводила площадь получившейся комнаты. Для числа π в стране Малевии используют значение 3.14.

Формат ввода, который используют Малевийцы:

Треугольник

a

b

c

где a, b и c — длины сторон треугольника

прямоугольник

a

b

где a и b — длины сторон прямоугольника

круг

r

где r — радиус окружности

Напишите программу, которая получает на вход три целых числа, по одному числу в строке, и выводит на консоль в три строки сначала максимальное, потом минимальное, после чего оставшееся число.

На ввод могут подаваться и повторяющиеся числа.

Паша очень любит кататься на общественном транспорте, а получая билет,

сразу проверяет, счастливый ли ему попался. Билет считается счастливым, если сумма первых трех цифр совпадает с суммой последних трех цифр номера билета.

Однако Паша очень плохо считает в уме, поэтому попросил вас написать программу, которая проверит равенство сумм и выведет "Счастливый", если суммы совпадают, и "Обычный", если суммы различны.

На вход программе подаётся строка из шести цифр. (Пример:123321)

Выводить нужно только слово "Счастливый" или "Обычный", с большой буквы.

Напишите программу, которая считывает с консоли числа (по одному в строке) до тех пор, пока сумма введённых чисел не будет равна 0 и **сразу после этого** выводит сумму квадратов всех считанных чисел.

Гарантируется, что в какой-то момент сумма введённых чисел окажется равной 0, **после этого считывание продолжать не нужно**.

В примере мы считываем числа 1, -3, 5, -6, -10, 13; в этот момент замечаем, что сумма этих чисел равна нулю и выводим сумму их квадратов, не обращая внимания на то, что остались ещё не прочитанные значения.

Напишите программу, которая выводит часть последовательности 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5 ... (число повторяется столько раз, чему равно). На вход программе передаётся неотрицательное целое число n — столько элементов последовательности должна отобразить программа. На выходе ожидается последовательность чисел, записанных через пробел в одну строку.

Например, если $n = 7$, то программа должна вывести 1 2 2 3 3 3 4.

Выведите таблицу размером $n \times n$, заполненную числами от 1 до n^2 по спирали, выходящей из левого верхнего угла и закрученной по часовой стрелке, как показано в примере (здесь $n=5$)