ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
«ДОМ ЮНОШЕСКОГО ТЕХНИЧЕСКОГО ТВОРЧЕСТВА ЧЕЛЯБИНСКОЙ ОБЛАСТИ»
ЦЕНТР ЦИФРОВОГО ОБРАЗОВАНИЯ «IT-КУБ» г. СНЕЖИНСК

ПРИНЯТО на заседании
педагогического совета
ГБУ ДО «ДЮТТ Челябинской области»
протокол № 135 от 15 мая 2023г.

УТВЕРЖДАЮ:
Директор ГБУ ДО «ДЮТТ»
Челябинской области»
В.Н. Халамов
Приказ № 54 «28» июня 2023 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«Программирование на Python»

Направленность: техническая
Срок освоения программы: 1 год
Уровень освоения: базовый
Возрастная категория обучающихся: 13-17 лет

Автор-составитель:
Столбов В.Н.,
педагог дополнительного образования

г. Снежинск
2023

СОДЕРЖАНИЕ

РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ 3

1.1 Пояснительная записка 3

1.2 Сведения о программе.....5

1.3 Цель и задачи программы 7

1.4 Содержание программы 8

1.5 Учебный план 11

1.6. Планируемые результаты 14

РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ ...15

2.1 Календарный учебный график 15

2.2 Условия реализации программы.....15

2.3 Формы аттестации 16

2.4 Оценочные материалы 17

2.5 Методические материалы 18

2.6 Воспитательный компонент.....18

2.7 Информационные ресурсы и литература 19

ПРИЛОЖЕНИЯ 20

Приложение 1. 20

Приложение 2. 24

Приложение 3. 26

Приложение 4 28

РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

1.1 Пояснительная записка

Программа разработана в соответствии со следующими нормативными документами:

- Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребенка в Российской Федерации»;
- Распоряжение Правительства РФ от 12.11.2020 № 2945-р «Об утверждении плана мероприятий по реализации в 2021 2025 г. г. Стратегии развития воспитания в Российской Федерации на период до 2025 года»;
- Концепция развития дополнительного образования детей до 2030 года и плана мероприятий по ее реализации, утвержденной распоряжением Правительства РФ от 31.03.2022 № 678-р•,
- Указ Президента Российской Федерации «Стратегия научно технологического развития Российской Федерации» (редакция от 15.03.2021г. № 143);
- Постановление Главного государственного санитарного врача РФ от
- 28.09.2020 N. 28 «Об утверждении санитарных правил СП 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Паспорт приоритетного проекта «Доступное дополнительное образование для детей», утвержденного президиумом Совета при Президенте РФ по стратегическому развитию и приоритетным проектам 30 ноября 2016 г;
- Приказ Министерства просвещения Российской Федерации от 27 июля 2022г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Методические рекомендации по проектированию дополнительных общеразвивающих программ, разработанных Министерством образования и науки России совместно с ГАОУ ВО «Московский государственный педагогический университет», ФГАУ «Федеральный институт развития образования», АНО дополнительного профессионального образования «Открытое образование»;
- Письмо Минобрнауки РФ от 18.112015 г. № 09-3242 «Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»;
- Распоряжение Правительства ЧО № 901-рп от 20.09.2022 г. «Об утверждении регионального плана мероприятий на 2022 — 2024 годы по реализации Концепции развития дополнительного образования детей до 2030 года»;
- Приказ Министерства труда и социальной защиты РФ № 652-н от 21.09.2021 г «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;
- Приказ Минпросвещения России от 03.09.2019 N 467 (ред. от 21.04.2023) «Об утверждении Целевой модели развития региональных систем дополнительного образования детей» (Зарегистрировано в Минюсте России 06.12.2019 N 56722);
- Закон Челябинской области от 29.08.2013 № 515-30 «Об образовании в Челябинской области»;
- Устав ГБУ ДО «Дом юношеского технического творчества Челябинской области».

Современный мир невозможно представить без программирования и его продуктов во всём их многообразии. Автоматизация коренным образом изменила большую часть отраслей человеческой деятельности. Программисты, инженеры по автоматизации и системные администраторы неизменно присутствуют в списке самых востребованных специальностей на рынке образования и труда. В настоящее время мы переживаем большие изменения в развитии общества. В современную жизнь человека всё больше внедряются компьютеры и

информационные технологии. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём зачастую не на пользовательском уровне, а на уровне начинающего программиста.

Российское образование в этой области не уступает в качестве зарубежным конкурентам, но фундаментальные основы изучения информатики в школе не достаточны для конкурентоспособности подрастающего поколения. Ситуация на рынке программирования постоянно меняется. Если в начале 2000-х веб-разработчики думали, что лучше всего учить PHP и Ruby, то в 2020 году быстрее всего развивается и растет Python. Одна из причин популярности Python– более простое оформление, чем в других языках. Его проще изучить с нуля даже новичку в программировании. Это самый популярный язык общего назначения: он используется для машинного обучения, аналитики, разработке игр и в науке о данных.

Дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» относится к технической направленности.

По степени освоения предполагается разделение программы по полугодиям на 2 уровня – ознакомительный и базовый.

По форме организации содержания и процесса педагогической деятельности программа является интегрированной.

Актуальность программы

Изучение основ программирования связано с развитием целого ряда умений и навыков (организация деятельности, ее планирование и т.д.), которые носят общеинтеллектуальный характер и формирование которых – одна из приоритетных задач в образовании.

Программа «Программирование на Python» позволяет посредством формирования начальных навыков программирования подготовить платформу для изучения в дальнейшем более сложных языков программирования. Данный учебный курс позволит обучающемуся самостоятельно моделировать алгоритмические конструкции. Одним из средств достижения запланированных результатов является язык программирования «Python».

Особенности языка Python: язык очень легок в изучении; язык является объектно-ориентированным, поэтому его реализация является самой понятной; на Python можно написать любую программу; широкий спектр применения; Python работает во всех операционных системах, так как это интерпретируемый язык; Python распространяется свободно на условиях лицензии GNU 2.0. Данная лицензия разрешает бессрочно использовать Python на любом количестве компьютеров в любых целях без оформления каких-либо дополнительных документов. Поэтому целесообразно использовать этот язык при изучении программирования школьников. В связи с этим становится очевидным актуальность предлагаемой программы.

В рамках изучения программы обучающиеся будут сталкиваться с необходимостью самостоятельной работы над заданиями: обучающиеся учатся решать задачи без помощи преподавателя. Для этого в содержании курса фигурируют задания, в которых для решения задачи необходимо найти информацию в сети Интернет; может потребоваться устранение ошибки, которую не так просто обнаружить; условие сформулировано недостаточно прозрачно и обучающемуся необходимо самостоятельно формализовать его (или задать правильные вопросы преподавателю). Все эти знания, умения и практические навыки решения актуальных задач, полученные на занятиях, готовят обучающихся к самостоятельной проектно-исследовательской деятельности с применением современных технологий.

Отличительные особенности программы заключаются в том, что программа не дублирует школьный курс информатики, а является его дополнением с профориентационными целями. Программа предполагает возможность участия обучающихся в соревнованиях, олимпиадах и конкурсах. Преимущество программы выражено в подборе интерактивных и практико-ориентированных форм занятий, способствующих формированию основных компетенций у обучающихся.

В программе предложен авторский подход в части структурирования учебного материала, определения последовательности его изучения, путей формирования системы знаний, умений и способов деятельности, развития, воспитания и социализации обучающихся.

Адресат программы – школьники, проявляющие интерес к информационным технологиям, и имеющие фундаментальные знания по математике, информатике и английскому языку.

Возраст обучающихся, участвующих в реализации данной программы: 13-17 лет.

На обучение принимаются все желающие, без предварительной подготовки, по заявлению родителей или лиц, их заменяющих. Набор в объединение производится по желанию обучающихся и их родителей.

Содержание программы учитывает возрастные и психологические особенности обучающихся 13-17 лет, которые определяют выбор форм проведения занятий. Обучающиеся этого возраста отличаются эмоциональностью, стремлением к активной практической деятельности, поэтому основной формой проведения занятий выбраны практические занятия. Обучающихся также увлекает совместная, коллективная деятельность, так как резко возрастает значение коллектива, общественного мнения, отношений со сверстниками, оценка поступков и действий обучающегося со стороны не только старших, но и сверстников. Обучающийся стремится завоевать в их глазах авторитет, занять достойное место в коллективе. Поэтому в программу включены практические занятия соревновательного характера, которые позволяют каждому проявить себя и найти своё место в детском коллективе.

Также следует отметить, что обучающиеся 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.

Форма обучения – очная, с возможностью применения дистанционных технологий.

Уровень освоения: базовый

Форма организации: группы 10-12 человек.

Режим занятий: 2 академических часа в неделю. 1 раз – по 2 часа (академический час – 45 мин.) Через каждые 45 минут занятия следует 15-минутный перерыв, согласно санитарным правилам СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи», утверждённых Постановлением Главного государственного санитарного врача РФ от 28.09.2020 № 28.

Форма организации занятий: групповая, индивидуально-групповая.

Методы обучения: словесный, наглядный, практический, объяснительно-иллюстративный, игровой, аналитический и др**.**

1.2 Сведения о программе

«Программирование на Python» на 2023-2024 уч. год

Название программы	Программирование на Python
Возраст обучающихся	13-17 лет
Длительность программы (в часах)	144 часа
Количество занятий в неделю	4 академических часа в неделю 2 раза - по 2 часа (академический час – 45 мин.)
Цель, задачи	Целью программы является формирование творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы,

	написанной на языке программирования Python. Задачи направлены на достижение цели и включают в себя обучающие, развивающие, воспитательные
Краткое описание программы	Программа «Программирование на Python» составлена в виде двух модулей Модуль 1 «Введение в программирование». В модуле рассматриваются основные типы данных, принципы и понятия структурного программирования. Используется версия языка Python ветки 3.x. Модуль 2 «Базовые конструкции языка Python». В модуле обучающиеся приобретают базовые навыки работы с основными конструкциями языка программирования; знакомство с комплексом базовых понятий и принципов функционального и объектно-ориентированного программирования (изучение структур данных, базовые принципы их обработки); получение опыта работы в интегрированной среде разработки на языке Python; формирование необходимых навыков работы с информацией (поиск, анализ, использование информации в сети Интернет)
Первичные знания, необходимые для освоения программы	Базовые знания, полученные при изучении школьной программы информатики, математики и английского языка
Результат освоения	Обучающиеся будут иметь практические навыки программирования на языке Python, научатся решать задачи на анализ статистических данных, обработке графической информации (построение стереоизображений, простых графических фильтров), построят задачи на использование простого искусственного интеллекта. Научаться работать в команде, разовьют личностные качества (активность, инициативность, волю, любознательность и т. п.). У обучающихся разовьется интеллект, внимание, память, восприятие, образное мышление и творческие способности; разовьются навыки анализа и оценки получаемой информации, разовьется мотивация к профессиональному самоопределению, сформируются навыки самоорганизации, воспитается самостоятельность, инициатива, творческая активность
Перечень соревнований, в которых обучающиеся смогут принять участие	Конференция «Юные техники и изобретатели» Хакатон Научно-инженерная олимпиада «Кванториада» Олимпиада HTIjunior
Перечень основного оборудования, необходимого для освоения программы	Ноутбук, WEB-камера, наушники, моноблочное интерактивное устройство, напольная мобильная стойка для интерактивных досок или универсальное настенное крепление, доска магнитно-маркерная настенная, флипчарт магнитно-маркерный на треноге, сетевой фильтр
Преимущества данной программы (отличия от других подобных курсов)	Программа предполагает возможность участия обучающихся в соревнованиях, олимпиадах и конкурсах. Преимущество программы выражено в подборе интерактивных и практико-ориентированных форм занятий, способствующих формированию основных компетенций у обучающихся

программы	

1.3 Цель и задачи программы

Целью программы является формирование творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

Задачи:

Обучающие:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- объяснить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);
- научить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

Метапредметные (развивающие):

- формирование умения ориентироваться в системе знаний;
- формирование умения выбирать наиболее эффективные способы решения задач на компьютере в зависимости от конкретных условий;
- формирование приёмов проектной деятельности, включая умения видеть проблему, формулировать тему и цель проекта, составлять план своей деятельности, осуществлять действия по реализации плана, результат деятельности соотносить с целью, классифицировать, наблюдать, проводить эксперименты, делать выводы и заключения, доказывать, защищать свои идеи, оценивать результаты своей работы;
- формирование умения распределять время;
- формирование умений успешной самопрезентации;
- развитие способности к самореализации.

Личностные (воспитательные):

- формирование умения самостоятельной деятельности;
- формирование умения работать в команде;
- формирование коммуникативных навыков;
- формирование навыков анализа и самоанализа;
- формирование целеустремлённости и усидчивости в процессе творческой, исследовательской работы и учебной деятельности.

Предметные (развивающие):

- формирование понятий «алгоритм», «программа»;
- формирование понятий об основных конструкциях языка программирования Python, таких как оператор ветвления if, операторы цикла while, for, вспомогательные алгоритмы;
- формирование понятий о структурах данных языка программирования Python;

- формирование основных приёмов составления программ на языке программирования Python;
- формирование алгоритмического и логического стилей мышления.

1.4 Содержание программы

Введение

Теоретическая часть: Введение в образовательную программу. Ознакомление обучающихся с программой, приёмами и формами работы. Вводный инструктаж по ТБ.

Модуль 1. Ознакомительный. «Введение в программирование»

Тема 1. Среда разработки, интерпретаторы, первая программа

Теоретическая часть: Код. Интерпретатор. Программы. Простейшие программы с выводом на экран.

Практическая часть: Знакомство с виртуальной средой взаимодействия – регистрация, организация личного кабинета, поиск и выкладывание материалов.

Тема 2. Простые операции, типы данных

Теоретическая часть: Интегрированные среды. Исполнение кода. Основные понятия программирования: исполнитель, система команд, алгоритм, программа, среда разработки. Сложение, вычитание, умножение, деление. Число, строка, плавающая точка

Практическая часть: Создание простейших текстовых математических программ

Тема 3. Степень, неполное частное, остаток от деления

Теоретическая часть: Деление с остатком и деление нацело, mod, div. Возведение в степень

Практическая часть: Создание простейших математических алгоритмов.

Тема 4. Операции со строками

Теоретическая часть: Типы данных, простейшие математические операции со строками.

Практическая часть: Создание программ по взаимодействую строк

Тема 5. Переменные

Теоретическая часть: Контейнеры, хранилище данных, память

Практическая часть: Использование переменных в математических операциях.

Тема 6. Ввод, вывод

Теоретическая часть: Функции ввода и вывода информации. Работа с вводимыми переменными

Практическая часть: Создание простейших текстовых программ взаимодействия с пользователем.

Тема 7. Логические выражения и сравнения

Теоретическая часть: Операторы сравнения больше, меньше, равно.

Практическая часть: Создание условий и сравнения.

Тема 8. if, else, elif

Теоретическая часть: Условный оператор. Переменные и арифметика. Погружение в условия. Ввод-вывод в программе. Условный оператор, оператор цикла с предусловием. Простейшие программы с использованием условного оператора if.

Практическая часть: Создание мини-проекта.

Тема 9. Булева логика

Теоретическая часть:Тип данных, True, False

Практическая часть: Создание мини-проекта.

Тема 10. Приоритет операторов

Теоретическая часть: Комбинаторные действия с операторами неравенства

Практическая часть: Создание мини-проекта.

Модуль 2. Базовый.
«Базовые конструкции языка Python»

Тема 11. Списки
Теоретическая часть:Тип данных список, коллекции переменных
Практическая часть: Создание мини-проекта. Работа со списками.
Тема 12. Функции списков
Теоретическая часть: Добавление, удаление,изменение данных
Практическая часть: Создание мини-проекта.
Тема 13. Циклы while
Теоретическая часть: Повторное использование кода, циклы
Практическая часть: Создание мини-проекта.
Тема 14. Циклы for
Теоретическая часть: Циклы по спискам,разница for и while
Практическая часть: Создание мини-проекта.
Тема 15. Функция range
Теоретическая часть: Работа функции, использование range и for
Практическая часть: Создание мини-проекта.
Тема 16. Повторное использование кода, функции
Теоретическая часть: Инструкция def
Практическая часть: Создание мини-проекта.
Тема 17. Аргументы функции
Теоретическая часть: Вход данных в функции
Практическая часть: Создание мини-проекта.
Тема 18. Возврат из функции
Теоретическая часть: Инструкция return. Использование данных из функции
Практическая часть: Создание мини-проекта.
Тема 19. Функции как объекты, модули
Теоретическая часть: Создание собственных библиотек
Практическая часть: Создание мини-проекта.
Тема 20. Стандартные библиотеки (math, random)
Теоретическая часть: Встроенные библиотеки, использование библиотек.
Практическая часть: Создание мини-проекта.
Тема 21. Свойства pip
Теоретическая часть: Использование pip и сторонних библиотек из интернета
Практическая часть: Создание мини-проекта.

Модуль 3. Работа с файлами Python

Тема 22. Исключения, обработка исключений
Теоретическая часть: Инструкции try, except, finally
Практическая часть: Создание мини-проекта.
Тема 23. Вызов исключений, утверждения
Теоретическая часть: Вызов внутренних ошибок, создание утверждений
Практическая часть: Создание мини-проекта.
Тема 24. Открытие и чтение файлов
Теоретическая часть: Работа с файловыми типами данных, открытие и чтение
Практическая часть: Создание мини-проекта.
Тема 25. Запись в файл
Теоретическая часть: Добавление данных в файл
Практическая часть: Создание мини-проекта.

Тема 26. Объект None

Теоретическая часть: Возврат пустого объекта. Использование пустых данных

Практическая часть: Создание мини-проекта.

Тема 27. Словари, функции словарей

Теоретическая часть: Тип данных словарь, ключи, значения, добавление, удаление, изменение данных, поиск по ключу

Практическая часть: Создание мини-проекта.

Тема 28. Кортежи, срезы, списковое включение

Теоретическая часть: Тип данных кортеж, использование срезов, списки через for

Практическая часть: Создание мини-проекта.

Тема 29. Формирование строк, другие полезные функции

Теоретическая часть: Метод format для строк, функции для строк, списков

Практическая часть: Создание мини-проекта.

Тема 30. Анализатор текста

Теоретическая часть: Программа для подсчета и анализа букв в текста

Практическая часть: Создание мини-программы для анализа текста из файла.

Тема 31. Функционально программирование

Теоретическая часть: Использование функций

Практическая часть: Создание мини-проекта.

Тема 32. Функция lambda

Теоретическая часть: Альтернативное создание функции, лямбда функции

Практическая часть: Создание мини-проекта.

Тема 33. Функция map, filter

Теоретическая часть: Лямбда функции для создания и фильтрации списков

Практическая часть: Создание мини-проекта.

Тема 34. Генераторы, рекурсия

Теоретическая часть: Многократный выход из функции, циклические функции, хвост рекурсии

Практическая часть: Создание мини-проекта.

Тема 35. Множества

Теоретическая часть: Тип данных множество, особенности , использование

Практическая часть: Решение задач с использованием множеств.

Тема 36. ООП, класс

Теоретическая часть:ООП как связь с миром, использование классов

Практическая часть: Создание мини-проекта.

Тема 37. Наследование

Теоретическая часть: Свойства классов, передача признаков от класса к классу

Практическая часть: Создание мини-проекта.

Тема 38. Магические методы, переопределение

Теоретическая часть: Написание инструкций магическими методами, переопределение методов класса

Практическая часть: Создание мини-проекта.

Тема 39. Сокрытие данных

Теоретическая часть: Понятия public, private, protected

Практическая часть: Создание мини-проекта.

Тема 40. Методы класса и статические методы

Теоретическая часть: @classmethod @staticmethod

Практическая часть: Создание мини-проекта.

Тема 41. Свойства

Теоретическая часть: Свойства классов

Практическая часть: Создание мини-проекта.

Тема 42. Простая игра
Теоретическая часть: Использование классов для определения объектов
Практическая часть: Создание консольной игры с открытым миром возможностей.
Тема 43. Регулярные выражения, простые метасимволы, классы символов
Теоретическая часть: Библиотека re, операции над строками
Практическая часть: Создание мини-проекта.
Тема 44. Другие метасимволы, группы, специальные последовательности
Теоретическая часть: Поиск в строке
Практическая часть: Создание мини-проекта.
Тема 45. Дзен языка Python, PEP
Теоретическая часть: Правила написания кода
Практическая часть: Создание мини-проекта.
Тема 46. Дополнительно об аргументах функции, распаковка кортежей, тернарный оператор, дополнительно об инструкции else
Теоретическая часть: Несколько аргументов функции,кортежи return
Практическая часть: Создание мини-проекта.
Тема 47. __main__, библиотеки, пакеты, сборка для пользователей
Теоретическая часть: Запуск главного файла, свои библиотеки и пакеты
Практическая часть: Создание мини-проекта.

Промежуточная аттестация
Теоретическая часть: Тестирование.
Практическая часть: Решение практических задач средствами языка программирования.

Аттестация по итогам освоения программы
Теоретическая часть: Защита проекта.
Практическая часть: Решение практических задач средствами языка программирования

1.5 Учебный план

№ п/п	Название модуля, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1	**Введение**	2	2	-	Входной: Тестирование
2	**Модуль 1. Ознакомительный. «Введение в программирование»**				
2.1	Тема 1. Среда разработки, интерпретаторы, первая программа	2	1	1	Текущий: Наблюдение, беседа
2.2	Тема 2. Простые операции, типы данных	2	1	1	Текущий: Наблюдение, беседа
2.3	Тема 3. Степень, неполное частное, остаток от деления	2	1	1	Текущий: Наблюдение, беседа
2.4	Тема 4. Операции, со строками	2	-	2	Текущий: Наблюдение, беседа
2.5	Решение задач	2	-	2	Текущий: Наблюдение, беседа
2.6	Тема 5. Переменные	2	1	1	Текущий: Наблюдение, беседа

2.7	Тема 6. Ввод, вывод	2	-	2	Текущий: Наблюдение, беседа
2.8	Тема 7. Логические выражения и сравнения	2	1	1	Текущий: Наблюдение, беседа
2.9	Тема 8. If, else, elif	2	1	1	Текущий: Наблюдение, беседа
2.10	Тема 9. Булева логика	2	1	1	Текущий: Наблюдение, беседа
2.11	Тема 10. Приоритет операторов	2	1	1	Текущий: Наблюдение, беседа
2.12	Решение задач	2	-	2	Текущий: Наблюдение, беседа
3	**Модуль 2. Базовый. «Базовые конструкции языка Python»**				
3.1	Тема 11. Списки	2	1	1	Текущий: Наблюдение, беседа
3.2	Тема 12. Функции списков	2	1	1	Текущий: Наблюдение, беседа
3.3	Решение задач	2	-	2	Текущий: Наблюдение, беседа
3.4	Тема 13. Циклы while	2	1	1	Текущий: Наблюдение, беседа
3.5	Решение задач	2	-	2	Текущий: Наблюдение, беседа
3.6	Тема 14. Циклы for	2	1	1	Текущий: Наблюдение, беседа
3.7	Тема 15. Функция range	2	1	1	Текущий: Наблюдение, беседа
3.8	Решение задач	2	-	2	Текущий: Наблюдение, беседа
3.9	Тема 16. Повторное использование кода, функции	2	1	1	Текущий: Наблюдение, беседа
3.10	Решение задач	2	-	2	Текущий: Наблюдение, беседа
3.11	Тема 17. Аргументы функции	2	1	1	Текущий: Наблюдение, беседа
3.12	Тема 18. Возврат из функции	2	-	2	Текущий: Наблюдение, беседа
2.13	Решение задач	2	-	2	Текущий: Наблюдение, беседа
3.14	Тема 19.Функции как объекты, модули	2	-	2	Текущий: Наблюдение, беседа
3.15	Тема 20. Стандартные библиотеки (math, random)	2	1	1	Текущий: Наблюдение, беседа
3.16	Тема 21. Свойства pip	2	1	1	Текущий: Наблюдение, беседа
3.17	Решение задач	2	-	2	Текущий: Наблюдение, беседа
3.18	Решение задач	2	-	2	Текущий: Наблюдение, беседа

3.19	Промежуточная аттестация	2	-	2	Промежуточный: Тестирование, решение практических задач
4	**Модуль 3. Работа с файлами Python**				
4.1	Тема 22. Исключения, обработка исключений	2	1	1	Текущий: Наблюдение, беседа
4.2	Тема 23. Вызов исключений, утверждения	2	1	1	Текущий: Наблюдение, беседа
4.3	Тема 24. Открытие и чтение файлов	2	1	1	Текущий: Наблюдение, беседа
4.4	Тема 25. Запись в файл	2	1	1	Текущий: Наблюдение, беседа
4.5	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.6	Тема 26. Объект None	2	1	1	Текущий: Наблюдение, беседа
4.7	Тема 27. Словари, функции словарей	2	1	1	Текущий: Наблюдение, беседа
4.8	Тема 28. Кортежи, срезы, списковое включение	2	1	1	Текущий: Наблюдение, беседа
4.9	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.10	Тема 29. Формирование строк, другие полезные функции	2	1	1	Текущий: Наблюдение, беседа
4.11	Тема 30. Анализатор текста	2	1	1	Текущий: Наблюдение, беседа
4.12	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.13	Тема 31. Функциональное программирование	2	1	1	Текущий: Наблюдение, беседа
4.14	Тема 32. Функция lambda	2	1	1	Текущий: Наблюдение, беседа
4.15	Тема 33. Функции map, filter	2	1	1	Текущий: Наблюдение, беседа
4.16	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.17	Тема 34. Генераторы, рекурсия	2	1	1	Текущий: Наблюдение, беседа
4.18	Тема 35. Множества	2	1	1	Текущий: Наблюдение, беседа
4.19	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.20	Тема 36. ООП, классы	2	1	1	Текущий: Наблюдение, беседа
4.21	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.22	Тема 37. Наследование	2	1	1	Текущий: Наблюдение, беседа

4.23	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.24	Тема 38. Магические методы, переопределение	2	1	1	Текущий: Наблюдение, беседа
4.25	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.26	Тема 39. Сокрытие данных	2	1	1	Текущий: Наблюдение, беседа
4.27	Тема 40. Методы класса и статические методы	2	1	1	Текущий: Наблюдение, беседа
4.28	Тема 41. Свойства	2	-	2	Текущий: Наблюдение, беседа
4.29	Тема 42. Простая игра	2	-	2	Текущий: Наблюдение, беседа
4.30	Решение задач	2	-	2	Текущий: Наблюдение, беседа
4.31	Тема 43. Регулярные выражения, простые метасимволы, классы символов	2	1	1	Текущий: Наблюдение, беседа
4.32	Тема 44. Другие метасимволы, группы, специальные последовательности	2	1	1	Текущий: Наблюдение, беседа
4.33	Тема 45. Дзен языка Python, PEP	2	1	1	Текущий: Наблюдение, беседа
4.34	Тема 46. Дополнительно об аргументах функции, распаковка кортежей, тернарный оператор, дополнительно об инструкции else	2	1	1	Текущий: Наблюдение, беседа
4.35	Тема 47. __main__, библиотеки, пакеты, сборка для пользователей	2	1	1	Текущий: Наблюдение, беседа
5	**Создание проекта**				
5.1	Работа над проектом №1	4	-	4	Текущий: Наблюдение, Беседа
5.2	Работа над проектом №2	4	-	4	Текущий: Наблюдение, Беседа
5.3	Аттестация по итогам освоения программы	2	-	2	Защита проекта
Итого	144	43	101		

1.6. Планируемые результаты

В процессе занятий по программе к окончанию учебного года обучающиеся будут знать:

- основные конструкции и идиомы языка программирования Python;
- термины «информация», «сообщение», «данные», «алгоритм», «программа»; понимание различий между употреблением этих терминов в обыденной речи и в информатике;

- логические значения, операции и выражения с ними;
- понятия класс, объект, обработка событий;
- методы проектной деятельности.

В процессе занятий по программе к окончанию учебного года обучающиеся будут уметь:

- на практике составить несложную программу для выполнения поставленной аналитической задачи;
- составлять линейные, разветвляющиеся и циклические алгоритмы управления исполнителями на языке программированияPython;
- использовать готовые прикладные компьютерные программы и сервисы;
- выбирать способы представления данных в зависимости от постановленной задачи;
- создавать мини-проекты на основе полученных знаний.

По итогам освоения программы, к окончанию учебного года, обучающиеся приобретут:

- *Метапредметные результаты:*

  - формирование навыков самоорганизации;
  - формирование навыков сотрудничества: работа в коллективе, в команде, микро-группе;
  - воспитание бережного отношение к технике;
  - воспитание самостоятельности, инициативности;
  - развитие навыков анализа и оценки получаемой информации.

- *Личностные:*

  - развитие личностных качеств (активность, инициативность, воля, любознательность и т. п.);
  - развитие внимания, памяти, восприятия, образного мышления;
  - развитие логического и пространственного воображения;
  - развитие творческих способностей и фантазии;
  - развитие мотивации к познанию и творчеству;
  - формирование положительных черт характера: трудолюбия, аккуратности, собранности, усидчивости, отзывчивости;
  - развитие мотивации к профессиональному самоопределению.

РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

2.1 Календарный учебный график

Год обучения	Всего учебных недель	Количество учебных часов	Режим занятий
1 год	36	144	4 академических часа в неделю. 2 раза - по 2 часа (академический час – 45 мин.)

2.2 Условия реализации программы

Материально-техническое обеспечение:

Занятия проходят в помещении с оптимальными условиями, отвечающими требованиям СанПиН, на базе Центра цифрового образования детей «IT-куб»
г. Снежинск.

Для реализации учебных занятий используется следующее оборудование и материалы:

Оборудование и расходные материалы:

- компьютеры и ноутбуки на каждого обучающегося и преподавателя;
- принтер;
- проекционное оборудование (экраны);
- магнитно-маркерная доска;
- интерактивная панель;
- whiteboard маркеры;
- бумага писчая;
- шариковые ручки;
- permanent маркеры.

Информационное обеспечение:

- операционная система Windows, Linux

Методическое обеспечение:

- варианты демонстрационных программ, материалы по терминологии ПО;
- инструкции по настройке оборудования;
- учебная и техническая литература;
- набор цифровых образовательных ресурсов – дидактические материалы, интерактивные тесты, анимационные плакаты.

Кадровое обеспечение:

Программа реализуется педагогом дополнительного образования Столбовым Владиславом Николаевичем; уровень образования : высшее (ФГФОУ ВО «Национальный Исследовательский Ядерный Университет «МИФИ» г. Москва, 2023 г. Квалификация – инженер. Специальность – «Применение и эксплуатация автоматизированных систем специального назначения»).

Курсы повышения квалификации – стажировка (40 ч.) по направлению «Информатика продвинутый уровень» 2022 г. ГБУ ДО ДЮТТ.

2.3 Формы аттестации

Система контроля знаний и умений обучающихся представляется в виде:

Входящий контроль осуществляется при комплектовании группы в начале учебного года.

Цель − определить исходный уровень знаний обучающихся, определить формы и методы работы с обучающимися.

Форма контроля: тестирование.

Текущий контроль осуществляется после изучения отдельных тем, раздела программы. В практической деятельности результативность оценивается качеством выполнения практических работ, поиску и отбору необходимого материала, умению работать с различными источниками информации. Анализируются положительные и отрицательные стороны работы, корректируются недостатки. Контроль знаний осуществляется с помощью заданий педагога (решение практических задач средствами языка программирования); взаимоконтроля, самоконтроля и др. Они активизируют, стимулируют работу обучающихся, позволяют более полно проявлять полученные знания, умения, навыки.

Промежуточная аттестация осуществляется в конце I полугодия учебного года.

Форма контроля: тестирование, решение практических задач средствами языка программирования.

Аттестация по итогам освоения программы осуществляется в конце учебного года.

Форма контроля: защита проекта.

Проект является одним из видов самостоятельной работы, предусмотренной в ходе обучения по программе. Педагог-наставник оказывает консультационную помощь в выполнении проекта.

Индивидуальный (групповой) проект оценивается формируемой комиссией. Состав комиссии (не менее 3-х человек): педагог-наставник, администрация учебной организации, приветствуется привлечение IT-профессионалов, представителей высших и других учебных заведений.

Компонентами оценки индивидуального (группового) проекта являются (по мере убывания значимости): качество индивидуального проекта, отзыв руководителя проекта, уровень презентации и защиты проекта. Если проект выполнен группой обучающихся, то при оценивании учитывается не только уровень исполнения проекта в целом, но и личный вклад каждого из авторов. Решение принимается коллегиально.

2.4 Оценочные материалы

В программу входят разнообразные оценочные материалы, в зависимости от темы занятия (Приложение).

Организация образовательного процесса в данной программе происходит в очной форме обучения, с возможностью применения дистанционных технологий, и групповой форме.

При реализации программы используются различные методы обучения:

- объяснительно-иллюстративный (предъявление информации различными способами (объяснение, рассказ, беседа, инструктаж, демонстрация, работа с технологическими картами и др.);
- проблемный (постановка проблемы и самостоятельный поиск её решения обучающимися);
- репродуктивный (воспроизводство знаний и способов деятельности по аналогу);
- поисковый (самостоятельное решение проблем);
- метод проблемного изложения (постановка проблемы педагогам, решение ее самим педагогом, соучастие обучающихся при решении);
- метод проектов (технология организации образовательных ситуаций, в которых обучающийся ставит и решает собственные задачи).

Для оценки результативности обучения и воспитания регулярно используются разнообразные методы: наблюдение за деятельностью; метод экспертной оценки преподавателем, мотивация, убеждение, поощрение, упражнение, стимулирование, создание ситуации успеха. Данные методы используются при анализе деятельности обучающихся, при организации текущей, промежуточной и итоговой аттестации обучающихся.

Перечисленные выше методы обучения используются в комплексе, в зависимости от поставленных целей и задач.

Формы организации учебного занятия по программе

В образовательном процессе помимо традиционного учебного занятия используются многообразные формы, которые несут учебную нагрузку и могут использоваться как активные способы освоения детьми образовательной программы, в соответствии с возрастом обучающихся, составом группы, содержанием учебного модуля:

- беседа;
- лекция;
- мастер-класс;
- практическое занятие;
- защита проектов;
- конкурс;
- викторина;
- диспут;

- круглый стол;
- «мозговой штурм»;
- воркшоп;
- квиз.

Некоторые формы проведения занятий могут объединять несколько учебных групп или весь состав объединения, например, экскурсия, викторина, конкурс и т. д.

В данной программе применяются следующие педагогические технологии:

- технология индивидуализации обучения;
- технология группового обучения;
- технология коллективного взаимообучения;
- технология дифференцированного обучения;
- технология разноуровневого обучения;
- технология проблемного обучения;
- технология развивающего обучения;
- технология дистанционного обучения;
- технология игровой деятельности;
- коммуникативная технология обучения;
- технология коллективной творческой деятельности;
- технология решения изобретательских задач;
- здоровье-сберегающая технология.

2.5 Методические материалы

ДИДАКТИЧЕСКИЕ ПОСОБИЯ - это наглядный вспомогательный методический материал, используемый при обучении актива, закреплении и проверке знаний. К дидактическим пособиям относятся презентации, различные познавательные игры и материалы к ним (пакеты с заданиями, карточки, художественное оформление игры).

Все многообразие методических материалов не может быть представлено в какой-то одной форме.

В рамках программы используются разработанные по темам презентации в формате .pptx, содержащие текстовый и визуальный материал, помогающий в изучении конкретной темы урока. Всего составлено 48 презентаций, а так же план-конспект на каждый урок.

Помимо презентаций, подготовленных к теоретической части календарно-тематического плана, так же существуют пособия по практике, представляющий собой набор задач как из открытых источников, так и авторски задачи, адаптированные к теме урока.

2.6 Воспитательный компонент

Общей целью воспитания является формирование у обучающихся духовно-нравственных ценностей, способности к осуществлению ответственного выбора собственной индивидуальной образовательной траектории, способности к успешной социализации в обществе. Задачи воспитания:

- поддерживать и развивать традиции учреждения, коллективные творческие формы деятельности, формировать у обучающихся чувство солидарности и принадлежности к образовательному учреждению; реализовывать воспитательный потенциал общеобразовательных общеразвивающих программ и возможности учебного занятия и других форм образовательных событий; развивать социальное партнерство как один из способов достижения эффективности воспитательной деятельности;

- организовывать работу с семьями обучающихся, их родителями или законными представителями, активно их включать в образовательный процесс, содействовать формированию позиции союзников в решении воспитательных задач; использовать в воспитании детей возможности занятий по дополнительным общеобразовательным общеразвивающим программам как источник поддержки и развития интереса к познанию и творчеству; содействовать приобретению опыта личностного и профессионального самоопределения на основе личностных проб в совместной деятельности и социальных практиках; формировать сознательное отношение обучающихся к своей жизни, здоровью, здоровому образу жизни, а также к жизни и здоровью окружающих людей.

Мероприятия по взаимодействию с родителями (проведение родительских собраний, совместных праздников, мастер-классов и т.д.) А также участие родителей в проектной деятельности, в разработке и защите проектов вместе с ребенком.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ МЕРОПРИЯТИЙ

Сроки	Уровень проведения соревнований	Название
декабрь	внутренний	«Новый год»
февраль	внутренний	«День науки»
май	внутренний	«Ярмарка проектов»

2.7 Информационные ресурсы и литература

Список литературы для педагога:

1. Бонцанини М.Анализ социальных медиа на Python. Извлекайте и анализируйте данные из всех уголков социальной паутины на Python - Издательство "ДМК Пресс" - 2018 - ISBN: 978-5-97060-574-5 - Текст электронный // ЭБС Лань - URL: https://e.lanbook.com/book/108129
2. Основы алгоритмизации и программирования на Python : учеб. пособие / С.Р. Гуриков. -М. : ФОРУМ : ИНФРА-М, 2018. - 343 с. - (Среднее профессиональное образование-Режим доступа:
3. http://znanium.com/catalog/product/961522
4. Язык программирования Python: практикум : учеб. пособие / Р.А. Жуков- М. : ИНФРА-М, 2019. - 216 с. + Доп. материалы [Электронный ресурс; Режим доступа: http://www.znanium.com]. - (Высшее образование: Бакалавриат). - ww.dx.doi.org/10.12737/textbook_5cb5ca35aaa7f5.89424805.

Список литературы для обучающихся:

1. Златопольский Д.М. Основы программирования на языке Python. – М.: ДМК Пресс,2017.–284с.ил."
2. Тарасов Д. Факультативный курс «Основы алгоритмизациии программирования на языке Python» 10-11 классы. Видеоуроки для учителей, родителей и учеников.

Электронные ресурсы:

1. Сообщество IT специалистов https://habr.com/ru/
2. Образовательный портал https://code.org/

ПРИЛОЖЕНИЯ

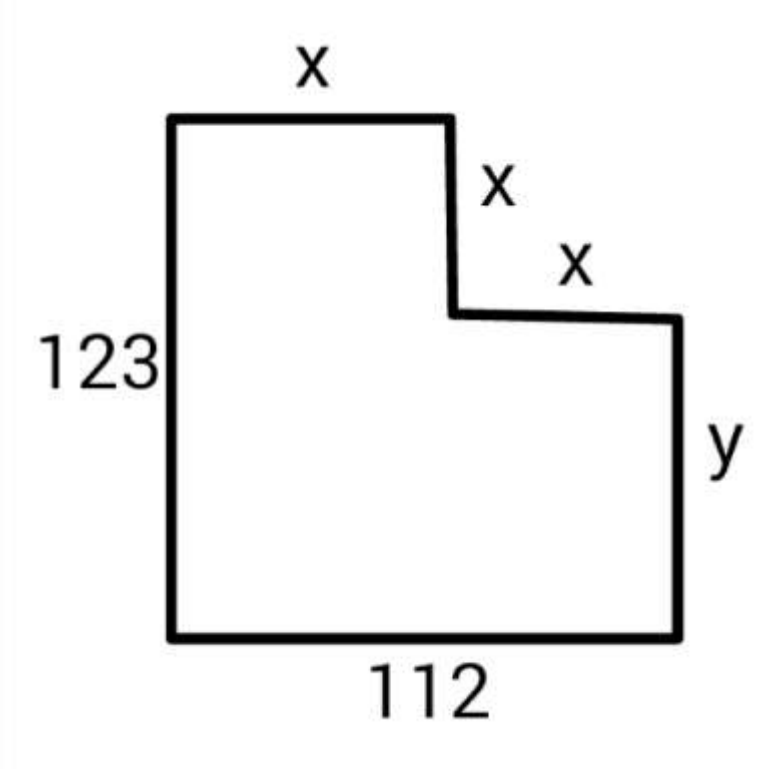
Приложение 1.

Входной контроль.

Для обучающихся – 13-17 лет

Фамилия, Имя	

Из прямоугольника вырезали квадрат со стороной x, получилась фигура как на рисунке. Чему равна сумма цифр?



В алфавите племени мумба-юмба 32 буквы. Любое слово в языке этого племени состоит из пяти букв и должно одинаково читаться справа налево и слева направо, при этом первые две буквы слова обязательно различаются, а третья совпадает с пятой.

Каково максимальное количество слов в этом языке?

Катя наклеила на рулет тонкие поперечные кольца трёх разных цветов. Если разрезать по серым кольцам, получится 25 кусков рулета, если по малиновым – 47 кусков, а если по зеленым – 31 кусок.

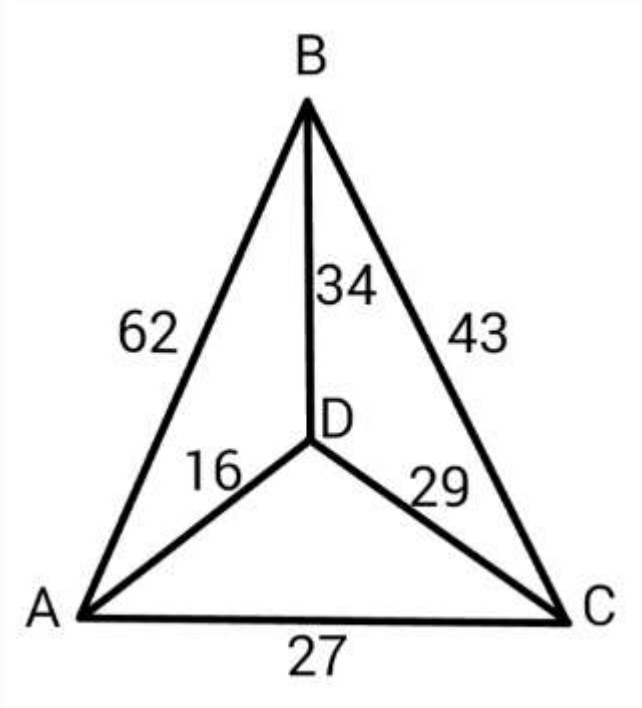
Сколько кусков рулета получится, если разрезать по кольцам всех трёх цветов?

Примечания

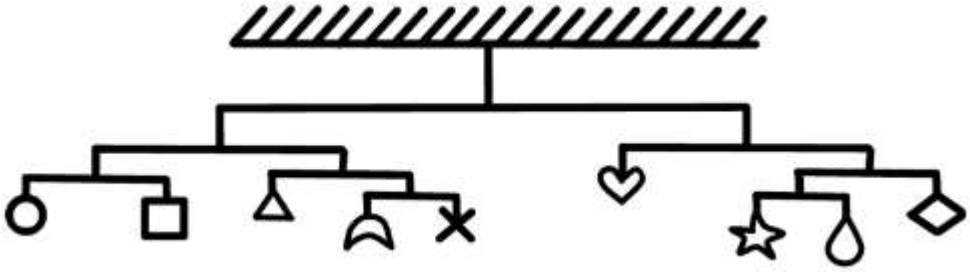
Учтите, что наклеить кольцо одного цвета на кольцо другого нельзя.

На рисунке показано расположение городов A, B, C и D и расстояния между ними. Турист выходит из города B и собирается посетить остальные города, побывав в каждом по разу.

Какова наименьшая возможная длина маршрута, если он хочет закончить свой путь в том же городе?



Фигурки, общей массой 432 грамма, при помощи невесомых нитей и планок собрали в конструкцию, изображённую на рисунке. Оказалось, что все её части находятся в равновесии. Сколько весит ромб?



Люди переезжают в города, за год численность людей удваивается. Если люди заселят весь город за 12 лет, то сколько лет понадобится, чтобы занять лишь половину города?

Сравните пары слов. Сколько среди них полностью идентичных?

O/Sanmarco	O/Samnarco
Ф.Wagonerrte	Ф.Wagonertre
A.S.Schmetterling	A.S.Schnetterling
N.V.Murfreesboroque	N.V.Munfreesboroque
P.S.Splendoursec	P.S.Sqlendoursec

Семь человек выясняли, какой сегодня день недели.
Первый сказал: «Послезавтра – воскресенье».
Второй: «Вчера был понедельник».
Третий: «Завтра будет суббота».
Четвертый: «Завтра будет среда».
Пятый: «Вчера был четверг».
Шестой: «Позавчера было воскресенье».
Седьмой: «Позавчера была среда».
Какой сегодня день недели, если трое ошибаются?

Вам предложены несколько высказываний и следствие из них (выделено жирным).
Согласны ли Вы с этим следствием?
Все клёны — растения.
Некоторые растения быстро желтеют.
Значит, некоторые клёны быстро желтеют.

☐ да

☐ нет

Гусеница прогрызает яблоко диаметром 6 сантиметров насквозь за 16 секунд, вылезая снаружи полностью.
Известно, что середину яблока она начинает грызть уже через 6 секунд после начала пути.
Какова длина гусеницы в сантиметрах?

Для обучающихся 11-13 лет

Фамилия, Имя	

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

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

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

Как с помощью только одной палочки образовать на столе треугольник?

В каком месяце болтливая девочка говорит меньше всего?

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

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

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.

Вопрос: Сколько кусков пиццы получилось у Юры.

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?

Критерии оценивания:

Низкий уровень: 0-70% выполненных заданий;

Средний уровень: 70-85% правильно выполненных заданий;

Высокий уровень: 85-100% правильно выполненных заданий.

Приложение 2.

Вопросы для промежуточного контроля по усвоению материала

Составьте выражение для вычисления в интерпретаторе Python 3 и вставьте в поле ответа результат вычисления:

11111·1111111

— произведение чисел 11111 (5 единиц) и 1111111 (7 единиц)

Запишите число 1.2345e3 в виде десятичной дроби.

Составьте и запишите выражение для вычисления:

2014.0^14(Возвестив 14 степень)

Обратите внимание на запись числа: это вещественное число.

Приведите к целому типу число 2.99

Расставьтескобкиввыражении

a and b or not a and not b

в соответствии с порядком вычисления выражения (приоритетом операций). Всего потребуется 5 пар скобок (внешние скобки входят в их число).

Найдите результат выражения для заданных значений a и b

Учитывайте регистр символов при ответе.

a = True

b = False

a and b or not a and not b

Отметьте выражения, значения которых равны True:

"239" < "30" and 239 < 30

"239" < "30" and 239 > 30

"239" > "30" and 239 < 30

"239" > "30" and 239 > 30

Укажите результат выражения:

"123" + "42"

Какое значение будет у переменной i после выполнения фрагмента программы?

```
i = 0
whilei<= 10:
i = i + 1
ifi> 7:
i = i + 2
```

Сколько итераций цикла будет выполнено в этом фрагменте программы?

```
i = 0
whilei<= 10:
i = i + 1
ifi> 7:
i = i + 2
```

Сколько всего знаков * будет выведено после исполнения фрагмента программы:

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

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

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

Критерии оценивания:

Низкий уровень: 0-70% выполненных заданий;

Средний уровень: 70-85% правильно выполненных заданий;

Высокий уровень: 85-100% правильно выполненных заданий.

Приложение 3.

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×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)**

Критерии оценивания:

Низкий уровень: 0-70% выполненных заданий;

Средний уровень: 70-85% правильно выполненных заданий;

Высокий уровень: 85-100% правильно выполненных заданий.

Приложение 4

Итоговые результаты по программе

Диагностическая карта

№	ФИО	Входная диагностика	Промежуточная диагностика	Итоговая диагностика

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

Обучающиеся 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.

Обучающиеся 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

Обучающиеся 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.

Обучающиеся должны уметь:

уважительно относится 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

Обучающиеся 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 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.

Обучающиеся 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 пдагога;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.