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

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

УТВЕРЖДАЮ:
Директор ГБУ ДО «ДЮТТ
Челябинской области»
_____ В.Н. Халамов
Приказ № от «348» 28 июня 2023 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

«Программирование на Python»

Направленность: техническая
Уровень программы: продвинутый
Срок освоения программы: 1 год
Возрастная категория обучающихся: 12 – 16 лет

Автор-составитель:
Юдина Наталья Владимировна,
педагог дополнительного образования

г. Снежинск,
2023

ОГЛАВЛЕНИЕ

TOC \F \H \Z \T "РАЗДЕЛ;1;ЗАГОЛОВОК;2" РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ 2

1.1 Пояснительная записка 3

1.2 Сведения о программе 5

1.3 Цель и задачи программы 6

1.4. Содержание программы 7

1.5 Учебный план 12

1.6. Планируемые результаты 14

РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ 14

2.1 Календарный учебный график 14

2.2 Условия реализации программы 15

2.3 Формы аттестации 15

2.4 Оценочные материалы 16

2.5 Методические материалы 16

2.6 Воспитательный компонент 17

2.7 Информационные ресурсы и литература 19

ПРИЛОЖЕНИЯ 20

Приложение 1. Входной контроль 20

Приложение 2. Материалы текущего контроля 23

Приложение 3. Материалы промежуточной аттестации 25

Приложение 4. Материалы аттестации по итогам освоения программы 26

Приложение 5. Примерный перечень мероприятий 30

Приложение 6. Оформление листа внесения изменений в программу 31

РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

1.1 Пояснительная записка

Программа разработана в соответствии со следующими нормативно-правовыми документами:

1.1.1. Федеральный закон "Об образовании в Российской Федерации" от 29.12.2012 N 273-ФЗ (последняя редакция).

1.1.2. Приказ Министерства Просвещения РФ № 629 от 27.07.2022 г. "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам".

1.1.3. 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".

1.1.4. Приказ Министерства труда и социальной защиты РФ № 652-н от 21.09.2021 г «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых».

1.1.5. 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).

1.1.6. Приказ Минпросвещения России от 21.04.2023 г. №302 «О внесении изменений в Целевую модель развития региональных систем дополнительного образования».

1.1.7. Локально-нормативные акты ГБОУ ДО ДЮТТ Челябинской области.

1.1.8. Постановление Главного государственного санитарного врача РФ от 28.09.2020 г. № 28 «Об утверждении санитарных правил СП 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».

Актуальность программы

Своевременность программы обусловлена тем, что 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.

Актуальная необходимость данной программы связана с тем, что изучение основ программирования развивает целый ряд умений и навыков (организация деятельности, ее планирование и т.д.), которые носят общеинтеллектуальный характер и формирование которых – одна из приоритетных задач в образовании.

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.

Соответствие потребностям времени обусловлено тем, что 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, моделирования научных процессов, в разработке игр и для работы с массивами данных.

Педагогическая целесообразность программы выражена в подборе интерактивных и практико-ориентированных форм занятий, способствующих формированию основных компетенций (информационных, коммуникативных, компетенций личного развития и др.). Данная дополнительная общеразвивающая программа направлена на развитие логического и пространственного мышления слушателя, способствует раскрытию творческого потенциала личности, формированию усидчивости и трудолюбия, приобретению практических умений и навыков в области компьютерных технологий, способствует интеллектуальному развитию обучающегося. Так же целесообразность программы заключается в создании такой методики изучения современных технологий программирования, которая даст почву для самообразования и практической, исследовательской, самостоятельной научной деятельности. Обучение нацелено на раннее выявление и становление талантливых детей как через приобретение знаний и умений, так и через развитие творческих навыков посредством участия в творческих конкурсных состязаниях, популяризации науки, научной, изобретательской деятельности.

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 муниципального, регионального, областного и всероссийского уровней. 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 12-16 лет, проявляющие интерес к информационным технологиям, и имеющие фундаментальные знания по математике, информатике и английскому языку.

На обучение принимаются все желающие, без предварительной подготовки, по заявлению родителей или лиц, их заменяющих.

Содержание программы учитывает возрастные и психологические особенности обучающихся 12-16 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. 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 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.

Срок реализации и объем программы определяется содержанием программы и составляет 1 год (144 академических часа).

Направленность (профиль) программы: техническая.

Язык реализации программы: государственный язык РФ – русский.

Особенности реализации программы: программа разработана с учётом модульного построения содержания. Каждый такой модуль охватывает отдельную информационную технологию или её часть. Внутри модуля разбивка по времени изучения производится педагогом самостоятельно, но с опорой на рекомендованный календарно-тематический план. С учётом

регулярного повторения ранее изученных тем продолжительность изучения отдельных разделов модуля определяется субъективными и объективными факторами. Модули реализуются по принципу «от простого к сложному».

Модули 1 и 2 по сложности представляют собой стартовый уровень. Стартовый уровень предполагает использование и реализацию общедоступных и универсальных форм организации материала, минимальную сложность предлагаемого для освоения содержания программы.

Модули 3, 4, 5 и 6 по сложности представляют собой базовый уровень. Базовый уровень предполагает использование и реализацию таких форм организации материала, которые допускают освоение специализированных знаний и языка, гарантированно обеспечивают трансляцию общей и целостной картины в рамках содержательно-тематического направления программы.

Уровень освоения программы: по степени освоения предполагается разделение программы по полугодиям на 2 уровня: 1 полугодие – стартовый и 2 полугодие – базовый.

Форма обучения – очная, с возможностью применения дистанционных технологий.

Формы организации: группа 10-12 человек

Режим занятий: 4 академических часа в неделю.

2 раза по 2 часа (академический час – 45 мин.).

Через каждые 45 минут занятия следует 10-минутный перерыв, согласно требованиям СанПиН.

Форма организации занятий: индивидуально–групповая.

Методы обучения: объяснительно-иллюстративный, практический и проблемно поисковый.

1.2 Сведения о программе

Название программы	Программирование на Python
Возраст обучающихся	12-16 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	Предложенная программа разработана с учётом модульного построения содержания. Модули реализуются по принципу «от простого к сложному». Модуль 1 «Вводный» – знакомство с обучающимися, вводный инструктаж по ТБ и ознакомление с программой, приёмами и формами работы. Модуль 2 «Введение в программирование». В модуле рассматриваются основные типы данных, принципы и понятия структурного программирования. Используется версия языка Python ветки 3.x. Модуль 3 «Базовые конструкции языка Python». Приобретаются базовые навыки работы с основными конструкциями языка программирования; знакомство с комплексом базовых понятий и

	принципов функционального и объектно-ориентированного программирования. Модуль 4 «Работа с файлами в Python». Приобретают базовые знания по созданию файла и записи, редактированию и извлечению информации из этого файла с помощью Python. Модуль 5 «Tkinter – графический интерфейс в Python». В модуле обучающиеся приобретают знания по созданию графического интерфейса на Python. Модуль 6 «Создание проекта». Приобретаются навыки самостоятельной работы при создании собственного проекта. Используются и закрепляются на практике, полученные знания.
Первичные знания, необходимые для освоения программы	Базовые знания, полученные при изучении школьной программы информатики, математики и английского языка
Результат освоения программы	Обучающиеся получат практические навыки программирования на языке Python. 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	В муниципальных, в областных и региональных, всероссийских конкурсах и олимпиадах по данному направлению.
Перечень основного оборудования, необходимого для освоения программы	Ноутбук, WEB-камера, МФУ (принтер, сканер, копир), 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 муниципального, регионального, областного и всероссийского уровней. 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.

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:

Образовательные задачи:

- 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.

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.

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.

1.4 Содержание программы

Модуль 1. Вводный.

Тема 1.1. Знакомство, инструктаж по ТБ, опрос, введение в образовательную программу.

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

Модуль 2. Введение в программирование

Тема 2.1 Понятия кода, интерпретатора, программы.

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 2.2. Переменные, основные операторы линейных алгоритмов

Теоретическая часть: Основные понятия программирования: исполнитель, система команд, математическая модель, алгоритм, программа, среда разработки. Основные операторы для линейных алгоритмов, ввод-вывод в программе.

Практическая часть: Создание линейных алгоритмов и программ.

Тема 2.3. Условный оператор if, if-else

Теоретическая часть: Условный оператор. Переменные и арифметика. Погружение в условия. Логические операторы AND, OR и NOT.

Практическая часть: Создание программ с использованием условного и логических операторов.

Тема 2.4. Цикл for

*Теоретическая часть**:*** Оператор цикла for. Алгоритм действия. Применение.

Практическая часть: Создание программ с использованием оператора циклов for. Технология разработки циклических программ.

Тема 2.5. Цикл while.

Теоретическая часть: Оператор цикла с предусловием while. Алгоритм действия. Применение

Практическая часть: Создание программ с использованием оператора циклов while. Технология разработки циклических программ с предусловием.

Тема 2.6. Оператор Continue

Теоретическая часть: Понятие оператора. Функция пропуска в цикле. Область применения пропусков.

Практическая часть: Взаимодействие с оператором в циклах. Создание ситуаций с пропуском.

Тема 2.7. Оператор Break

Теоретическая часть: Понятие оператора. Функция остановки и выхода из цикла. Область применения оператора.

Практическая часть: Взаимодействие с оператором в циклах. Создание циклов с необходимостью выхода из него.

Тема 2.8. True и False

Теоретическая часть: Понятие булевских переменных. Значения истинности и ложности выражений в Python. Переменные истинности.

Практическая часть: Создание переменных в булевском формате и работа с ними на практике. Проверка переменной на истинность или ложность.

Тема 2.9. Модуль keyword

Теоретическая часть: Понятие ключевых слов. Изучение ключевых слов в языке Python

Практическая часть: Область применение ключей. Проверка строки на ключевое слово.

Тема 2.10. Числа: целые, вещественные, комплексные

Теоретическая часть: Общая теория вида переменных в языках программирования. Изучения грамматики чисел в Python. Преобразование типов переменных.

Практическая часть: Определение типа переменной. Работа с несколькими переменными. Преобразование одного вида переменной в определенный вид другого типа.

Тема 2.11. Решение задач повышенной сложности по изученным темам

Практическая часть: Решение олимпиадных задач по программированию.

Тема 2.12. Контрольная работа

Практическая часть: выполнение тестовых заданий, составление алгоритмов (блок-схем) и программ по индивидуальным заданиям.

Модуль 3. Базовые конструкции языка Python

Тема 3.1. Строки. Базовые операции

Теоретическая часть: Понятие строки. Литералы строк. Строки в апострофах и кавычках. Экранирование служебных символов в строке. Работы со строками.

Практическая часть: Конкатетанция. Дублирование строк. Длина строки. Доступ по индексу.

Тема 3.2. Другие функции и методы строк

Теоретическая часть: Понятие категории неизменяемых последовательностей. Обозначения для работы с последовательностью строк.

Практическая часть: Работа и форматирование строк при помощи быстрых команд. Применение быстрых команд на практике.

Тема 3.3. Метод format

Теоретическая часть: Использование строки как элемента. Подстановка значений в текст. Синтаксис объекта format. Флаги.

Практическая часть: Редактирование элементов. Взаимодействие с format и флагами.

Тема 3.4. Списки.

Теоретическая часть: Тип данных списки. Использование списков. Списки и литералы. Генераторы списков.

Практическая часть: Создание списков. Использование циклов для создания списков. Генерация циклом for.

Тема 3.5. Функции списков

Теоретическая часть: Встроенные функции списков. Работа над списками. Таблица «методы списков».

Практическая часть: Редактирование списков. Быстрые команды. Строковые методы.

Тема 3.6. Индексы и срезы

Теоретическая часть: Что такое Индекс. Понятие и грамматика среза. Индексы в списках. Выделение информации срезом.

Практическая часть: Определение необходимого индекса. Работа с списком при помощи индекса. Нахождение объекта методом среза.

Тема 3.7. Кортежи

Теоретическая часть: Тип данных Кортеж. Применение Кортежей. Защита.

Практическая часть: Использование кортежей в качестве ключей.

Тема 3.8. Операции с кортежами

Теоретическая часть: Пустой кортеж. Строка в кортеже. Создание кортежа.

Практическая часть: Сложение, умножение на число, методы index() и count().

Тема 3.9. Словари. Методы словарей

Теоретическая часть: Тип данных словарей. Ассоциативные массивы. Функции dict(). Метод fromkeys. Генератор словарей. Методы словарей и их список.

Практическая часть: Работа со словарем. Добавление записей. Извлечение значения ключей. Работа с методами.

Тема 3.10. Решение задач повышенной сложности по изученным темам

Практическая часть: Решение олимпиадных задач по программированию.

Тема 3.11. Контрольная работа

Практическая часть: Работа со словарем. Добавление записей. Извлечение значения ключей. Работа с методами.

Модуль 4. Работа с файлами в Python

Тема 4.1. Множества (set, frozenset)

Теоретическая часть: Что такое множество. Использование множества для удаления элементов. Другие операции над множеством. Set. Frozenset.

Практическая часть: Операции нахождения, объединения, пересечения и операции непосредственно изменяющие множество.

Тема 4.2. Именные функции, инструкция def

Теоретическая часть: Функции, именные и анонимные. Инструкция def, return и lambda. Обязательные и необязательные аргументы функции. Функция с произвольным числом аргументов.

Практическая часть: Определение функции. Создание различных функций. Возвращение значений из функции.

Тема 4.3. Аргументы функции

Теоретическая часть: Принятие аргументов. Позиционные и именные аргументы. Обязательные и необязательные аргументы. Переменное количество позиционных аргументов.

Практическая часть: Создание функции и направление в неё одного или нескольких аргументов.

Тема 4.4. Анонимные функции, инструкция lambda

Теоретическая часть: Инструкция lambda. Отличие от def func().

Практическая часть: Создание функций через lambda. Примеры работ с Анонимными функциями.

Тема 4.5. Байты

Теоретическая часть: Байтовые строки в Python. Что такое Байт. Бинарная система исчисления. Информация в Байте. Bytearray.

Практическая часть: Функция bytes. Метод decode.

Тема 4.6. Null в Python: None

Теоретическая часть: Концепция ключевого слова null. «Нулевое» поведение.

Практическая часть: Присваивание значений. Использование None. Проверка на None.

Тема 4.7. Работа с файлами

Теоретическая часть: Встроенные средства Python для работы с файлами.

Практическая часть: Открытие. Закрытие. Чтение. Запись.

Тема 4.8. Чтение и запись в файл

Теоретическая часть: Операции чтения и записи.

Практическая часть: Создание файла и редактирование в нем информации.

Тема 4.9. Обработчик исключений

Теоретическая часть: Понятие ошибок. Конструкция «try-expect».

Практическая часть: Создание и отслеживание ошибок. Альтернативные варианты событий при ошибке.

Тема 4.10. Менеджер «with..as»

Теоретическая часть: Работа с файлами при помощи менеджера «with..as». Синтаксис конструкции with ... as.

Практическая часть: Оборачивания выполнения блока инструкций менеджером контекста.

Тема 4.11. Модули в Python

Теоретическая часть: Ознакомление с подключением пакетов через встроенный менеджер «pip». Подключение стандартных модулей в Python.

Практическая часть: Создание своих собственных модулей. Подключение пакетов через встроенный менеджер «pip».

Тема 4.12. Контрольная работа

Практическая часть: выполнение тестовых заданий, написание программ по индивидуальным заданиям.

Модуль 5. Tkinter – графический интерфейс в Python

Тема 5.1. Введение в Tkinter. Окно приложения

Теоретическая часть: Tkinter и графическая библиотека Tk. Преимущества и недостатки Tkinter. Окно, как основа для графического интерфейса. Методы и атрибуты, позволяющие устанавливать различные аспекты окон.

Практическая часть: Создание окон различных размеров, работа с заголовками, иконками и атрибутами окон.

Тема 5.2. Виджеты: Button, Label, Entry.

Теоретическая часть: Введение в виджеты. Основные параметры виджетов Button, Label и Entry.

Практическая часть: Работа по созданию и редактированию кнопок, текстовых меток и однострочных текстовых полей.

Тема 5.3. Виджеты: Checkbutton, Listbox, Combobox, Scrollbar.

Теоретическая часть: Основные параметры виджетов Checkbutton, Listbox, Combobox и Scrollbar.

Практическая часть: Работа по созданию и редактированию списков, выпадающих списков, полос прокруток и кнопок-флажков.

Тема 5.4. Виджет Text.

Теоретическая часть: Основные параметры для создание многострочного текстового поля.

Практическая часть: Работа по созданию и редактированию многострочного текстового поля.

Тема 5.5. Виджет Treeview. Создание таблиц и деревьев

Теоретическая часть: Основные параметры Treeview для отображения иерархических данных в виде дерева или в виде таблицы.

Практическая часть: Работа по созданию и редактированию иерархических данных.

Тема 5.6. Контрольная работа

Практическая часть: Разработка графического интерфейса будущего проекта.

Модуль 6. Реализация итогового проекта

Тема 6.1. Работа над итоговым проектом.

Практика: Разработка итогового проекта, тестирование и отладка. Структурирование элементов проекта, подготовка среды сборки приложений к компиляции, проверка параметров. Компиляция. Оценка целостности и законченности проекта. Доработка приложения, исправление ошибок. Итоговая компиляция.

Тема 6.2. Подготовка к защите VR/AR-**проекта. Разработка презентации для представления VR/AR-проекта.**

Теория: определение цели, задачи, гипотезы и проблемы.

Практика: экспорт необходимых материалов, подготовка презентации, исходников, видеороликов.

Тема 6.3. Аттестация по итогам освоения программы.

Практика: защита итогового проекта.

1.5 Учебный план

1№ п/п	Название модуля, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1	**Модуль 1. Вводный.**	**2**	**2**	**0**	
1.1	Знакомство, инструктаж по ТБ, опрос, введение в образовательную программу.	2	2	0	Входная диагностика: тестирование, опрос
2	**Модуль 2. Введение в программирование**	**38**	**17**	**21**	
2.1	Знакомство со средой, понятия кода, интерпретатора, программы	4	2	2	Текущая аттестация: наблюдение, беседа, опрос, выполнение практических заданий по темам и заданий для самостоятельной работы.
2.2	Переменные, основные операторы линейных алгоритмов	4	2	2	
2.3	Условный оператор if, if-else	4	2	2	
2.4	Цикл for	4	2	2	
2.5	Цикл while	4	2	2	
2.6	Оператор Continue	2	1	1	
2.7	Оператор Break	2	1	1	
2.8	True и False	2	1	1	
2.9	Модуль keyword	4	2	2	
2.10	Числа: целые, вещественные, комплексные	4	2	2	
2.11	Решение задач повышенной сложности по изученным темам	2	-	2	
2.12	Контрольная работа	2	-	2	Тестирование, выполнение заданий
3	**Модуль 3. Базовые конструкции языка Python**	**30**	**12**	**18**	
3.1	Строки. Базовые операции	2	1	1	Текущая аттестация: наблюдение, беседа, опрос, выполнение практических заданий по темам и заданий для самостоятельной работы.
3.2	Другие функции и методы строк	2	1	1	
3.3	Метод format	4	2	2	
3.4	Списки.	2	1	1	
3.5	Функции списков	2	1	1	
3.6	Индексы и срезы	4	2	2	
3.7	Кортежи.	2	1	1	
3.8	Операции с кортежами	2	1	1	
3.9	Словари. Методы словарей	4	2	2	
3.10	Решение задач повышенной сложности по изученным темам	4	-	4	
3.11	Контрольная работа	2	-	2	Тестирование, решение практических задач

4	**Модуль 4. Работа с файлами в Python**	**42**	**20**	**22**	
4.1	Множества (set, frozenset)	4	2	2	Текущая аттестация: наблюдение, беседа, опрос, выполнение практических заданий по темам и заданий для самостоятельной работы.
4.2	Именные функции, инструкция def	4	2	2	
4.3	Аргументы функции	4	2	2	
4.4	Анонимные функции, инструкция lambda	4	2	2	
4.5	Байты	4	2	2	
4.6	Null в Python: None	4	2	2	
4.7	Работа с файлами	2	1	1	
4.8	Чтение и запись в файл	2	1	1	
4.9	Обработчик исключений	4	2	2	
4.10	Менеджер «with..as»	4	2	2	
4.11	Модули в Python	4	2	2	
4.12	Контрольная работа	2	-	2	Тестирование, выполнение заданий
5	**Модуль 5. Tkinter – графический интерфейс в Python**	**20**	**8**	**12**	
5.1	Введение в Tkinter. Окно приложения	2	1	1	Текущая аттестация: наблюдение, беседа, опрос, выполнение практических заданий по темам и заданий для самостоятельной работы.
5.2	Виджеты: Button, Label, Entry.	4	2	2	
5.3	Виджеты: Checkbutton, Listbox, Combobox, Scrollbar.	4	2	2	
5.4	Виджет Text.	2	1	1	
5.5	Виджет Treeview. Создание таблиц и деревьев	4	2	2	
5.6	Контрольная работа	4	0	4	Разработка графического интерфейса будущего проекта (мини-проект)
6	**Модуль 6. Реализация итогового проекта**	**12**	**1**	**11**	
6.1	Работа над итоговым проектом.	8	-	8	Текущая аттестация: наблюдение, консультации.
6.2	Подготовка к защите проекта. Разработка презентации для представления проекта.	2	1	1	
6.3	Аттестация по итогам освоения программы	2	-	2	Защита итогового проекта
Итого		**144**	**60**	**84**	

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 программирования Python;
- 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 помещениис оптимальными условиями, отвечающими требованиям СанПиН, на базе Центра цифрового образования детей «IT-куб» г. Снежинск.

Для реализации учебных занятий используется следующее оборудование и 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.
- пакет приложений Office
- программное обеспечение: Python, Jupyter Notebook в составе дистрибутива Anacodnda, среда разработки PyCharm.

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 Юдиной Натальей Владимировной, уровень образования: высшее - Аспирантура СФТИ НИЯУ МИФИ. Курсы повышения квалификации: «Формирование гибких компетенций у обучающихся: вводный уровень», 2022 г. ФЦДО; «Введение в VR\AR - направление», 2022 г. ФЦДО; «Основы трехмерного моделирования: работа в редакторе Blender», 2023 г. ФЦДО.

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.

Формы контроля: наблюдение, опрос, выполнение практических заданий по темам и заданий для самостоятельной работы. 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 и формы контроля знаний (Приложение). Программа включает следующие формы контроля: входящая диагностика, текущий контроль, промежуточная аттестация и аттестация по итогам освоения программы.

Входящая диагностика проводится в форме устного опроса и анализа выполненного обучающимися теста. Текущий контроль проводится по итогам прохождения каждой темы в форме наблюдения, устного опроса, выполнения практических заданий. Контроль промежуточной аттестации проводится в форме тестирование и решение практических задач. Аттестация по итогам освоения программы проводится в виде защиты итогового проекта.

Оценочные материалы для всех форм контроля программы представлены перечнем используемых диагностик, которые позволяют определить достижение обучающимися планируемых результатов. Это списки вопросов для устных опросов, критерии оценивания работ, тесты, перечень практических заданий по каждой теме.

2.5 Методические материалы

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

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);
- поисковый (самостоятельное решение проблем);
- метод проблемного изложения (постановка проблемы педагогам, решение ее самим педагогом, соучастие обучающихся при решении);
- проектный (технология организации образовательных ситуаций, в которых обучающийся ставит и решает собственные задачи).

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

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

В образовательном процессе помимо традиционного учебного занятия используются многообразные формы организации учебных занятий, 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.

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.

Методическое обеспечение учебного процесса включает разработку преподавателем методических пособий, вариантов демонстрационных программ и справочного материала.

2.6 Воспитательный компонент

Воспитание в дополнительном образовании рассматривается, прежде всего, как организация педагогических условий и возможностей для осознания ребенком собственного личностного опыта, приобретаемого на основе межличностных отношений и обусловленных ими ситуаций, проявляющегося в форме переживаний, смыслотворчества, саморазвития.

При осуществлении настоящей программы необходимо выполнение следующих воспитательных задач:

– поддерживать и развивать традиции учреждения, коллективные творческие формы деятельности, реализовать воспитательные возможности ключевых дел ГБУ ДО ДЮТТ, формировать у обучающихся чувство солидарности и принадлежности к образовательному учреждению;

– формирование активной жизненной позиции, гражданско-патриотической ответственности;

– использовать в воспитании детей возможности занятий по дополнительным общеобразовательным общеразвивающим программам как источник поддержки и развития интереса к познанию и творчеству;

– приобщение обучающихся к российским традиционным духовным ценностям, правилам и нормам поведения в обществе;

– воспитание аккуратности и дисциплинированности при выполнении работ;

– развитие коммуникативных отношений внутри проектных групп и в коллективе в целом;

– воспитание этики групповой работы, отношения делового сотрудничества, взаимоуважения;

– воспитание внимательности, аккуратности и изобретательности при работе с техническими устройствами, разработке приложений и выполнении учебных проектов;

– содействовать приобретению опыта личностного и профессионального самоопределения на основе личностных проб в совместной деятельности и социальных практиках;

– формирование готовности обучающихся к саморазвитию;

– формировать сознательное отношение обучающихся к своей жизни, здоровью, здоровому образу жизни, а также к жизни и здоровью окружающих людей.

– создавать инновационную среду, формирующую у детей и подростков креативное, критическое мышление через освоение дополнительных общеобразовательных общеразвивающих программ нового поколения в области инженерных и цифровых технологий;

– формирование готовности обучающихся к саморазвитию.

2.7 Информационные ресурсы и литература

Список литературы:

1. Таненбаум Эндрю, Остин Тодд. Архитектура компьютера. – СПб.: Прогресс книга, 2022. – 816 с.
2. Вильям Спрингер. Гид по Computer Science для каждого программиста. – СПб.: Питер, 2020. – 193 с.
3. Тимофеева Е.В. Информатика. – М.: Эксмо, 2021. – 176 с.
4. Никола Лейси. Python, например. – СПб.: Питер, 2021 – 192 с.
5. Ли Воган. «Непрактичный» Python. – СПб.: БХВ-Петербург, 2021. – 457 с.

Литература, рекомендованная обучающимся:

1. Дэвид Копец. Классические задачи Computer Science на языке Python. – СПб.: Питер, 2022. – 224 с.
2. Таненбаум Эндрю, Бос Херберт. Современные операционные системы. – СПб.: Питер, 2022. – 1120 с.
3. Джейми Чан. Python Быстрый старт. – СПб.: Питер, 2022. – 224 с.

Электронные ресурсы:

1. Питонтьютор. Бесплатный курс по программированию с нуля. (Электронный ресурс). – Режим доступа: https://pythontutor.ru/
2. Code Basics: обучение базовым аспектам языков программирования от образовательной платформы Hexlet. (Электронный ресурс). – Режим доступа: https://ru.code- basics.com/
3. Образовательный портал. (Электронный ресурс) – Режим доступа https://code.org/

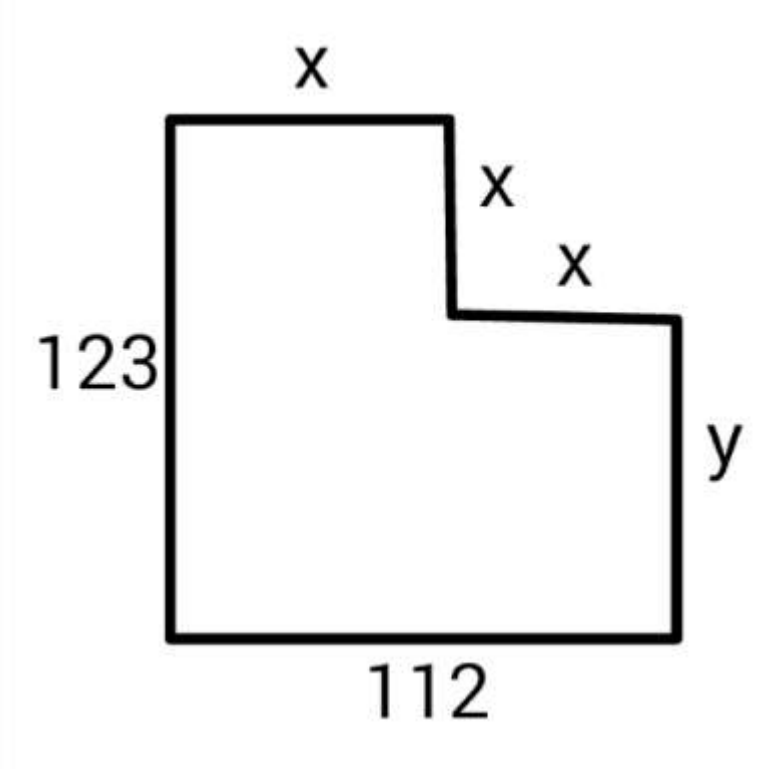
ПРИЛОЖЕНИЕ

Приложение 1. **Входной контроль.**

Для обучающихся – 14-16 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.

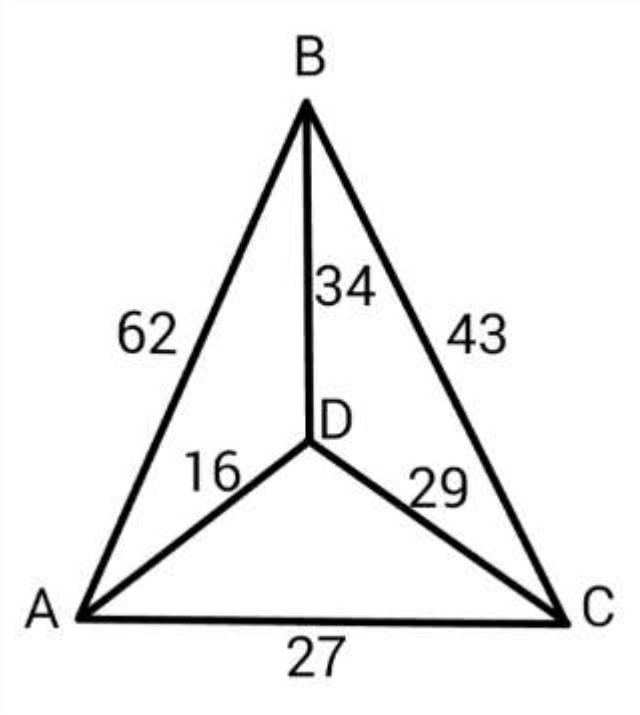
Сколько кусков рулета получится, если разрезать по кольцам всех трёх цветов?

Примечания

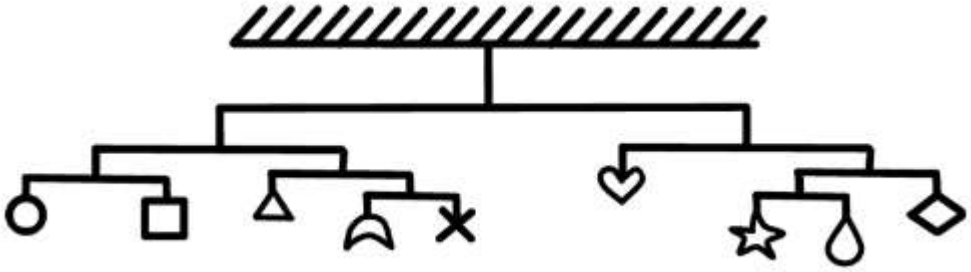
Учтите, что наклеить кольцо одного цвета на кольцо другого нельзя.

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.

☐ а

☐ ет

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 12-13 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. **Материалы текущего контроля**

Вопросы для текущего контроля

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. **Материалы промежуточной аттестации**

Задачи для промежуточной аттестации

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:

1) Треугольник: a, b, c

где a, b и c — длины сторон треугольника

2) Прямоугольник: a, b

где a и b — длины сторон прямоугольника

3) 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. **Материалы аттестации по итогам освоения программы**

Аттестация по итогам освоения программы проводится в форме защиты созданных обучающимися проектов. Список предлагаемых тем проектов для выполнения обучающимися согласуется с учебной частью не позднее, чем за 3 месяца до проведения аттестации по итогам освоения программы.

Допускается организация детей в группы до 3-х человек с распределением ролей между ними. После разработки обучающимся необходимо подготовить презентацию и устный рассказ об их проекте (суть проекта, практическая применимость и полезность).

Требования к содержанию проектной работы:

- Проекты должны решать актуальные проблемы;
- Проект должен представлять исследовательскую, практикоориентированную работу в соответствии с возрастными особенностями обучающихся;
- Проект должен иметь практическую направленность, востребованность, возможность применения в той или иной сфере деятельности.

После завершения работы над проектом обучающиеся презентуют результаты его выполнения.

Результативность отслеживается с помощью диагностической карты, 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).

Диагностическая карта

№	ФИО	Входная диагностика	Промежуточная диагностика	Итоговая диагностика

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 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.

Приложение 5.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ МЕРОПРИЯТИЙ

Сроки	Уровень проведения соревнований	Название соревнований, конкурсов, мероприятий

Приложение 6.

ОФОРМЛЕНИЕ ЛИСТА ВНЕСЕНИЯ ИЗМЕНЕНИЙ В ПРОГРАММУ

«СОГЛАСОВАНО»
Заведующий учебной части/методист _____ / /

«_____» _____ 202___ г.

Лист изменений в программе на 202____ г.

№	Раздел программы	Внесённые изменения
1	Титульный лист	
2	Пояснительная записка	
3	УП и содержание программы	
4	Календарный учебный график	
5	Условия реализации программы	
6	Формы аттестации. Оценочные материалы	
7	Методическое обеспечение	
8	Список литературы	