

Управление образования администрации ЗАТО Александровск
муниципальное бюджетное учреждение
дополнительного образования
«Дом детского творчества «Дриада»

СОГЛАСОВАНО
Педагогическим советом
МБУДО «ДТТ «Дриада»
Протокол № 6 от 14.12.2023 г.



Дополнительная общеобразовательная общеразвивающая программа
технической направленности

«Программирование на Python»

Возраст обучающихся: 14-17 лет

Срок реализации: 1 год

Уровень программы: базовый

Автор-составитель:
Кудашева Светлана Геннадьевна,
методист

ЗАТО Александровск
г.Снежногорск
2023 г.

АННОТАЦИЯ

В настоящее время мы переживаем большие изменения в развитии общества. Информационные технологии стали неотъемлемой частью нашей жизни. Большое значение приобретает умение работать с компьютером, притом на уровне профессионала, а не обывателя. Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке.

В данной программе выбран язык программирования Python. Выбор обусловлен синтаксисом языка, который является достаточно простым и интуитивно понятным, что понижает порог вхождения и позволяет сосредоточиться на логических и алгоритмических аспектах программирования, а не на выучивании тонкостей синтаксиса. При этом, Python является очень востребованным языком; он отлично подходит для знакомства с различными современными парадигмами программирования и активно применяется в самых разных областях от разработки веб-приложений до машинного обучения.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1. Нормативно-правовая база разработки и реализации программы.

Данная программа составлена на основе дополнительной общеобразовательной общеразвивающей программы технической направленности «Программирование на Python», Комаровой Т.В, Кузнецовой К.В., Борисова Н.А., ЦЦОД «IT-куб», г.Мурманск, 2023 г.

Программа разработана на основе следующих нормативных документов:

- Федерального закона от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденного приказом Министерства просвещения Российской Федерации от 27.07.2022 г. № 629;
- Постановления Главного государственного санитарного врача Российской Федерации от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Постановления Главного государственного санитарного врача Российской Федерации от 28.01.2021 № 2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания»;

– Приказа Минобрнауки России №882, Минпросвещения России №391 от 05.08.2020 «Об организации и осуществлении образовательной деятельности при сетевой форме реализации образовательных программ» (вместе с «Порядком организации и осуществления образовательной деятельности при сетевой форме реализации образовательных программ»);

– Письмо Минпросвещения России от 29.03.2023 г. №АБ-1339/02 «О направлении методических рекомендаций по созданию и функционированию центров цифрового образования «IT-куб»;

– Распоряжение Минпросвещения России №Р-5 от 12.01.2021 г. «Об утверждении методических рекомендаций по созданию и функционированию центров цифрового образования ITкуб»»;

– Письмо Министерства просвещения РФ от 10 ноября 2021 г. N ТВ-1984/04 «О направлении методических рекомендаций»;

– Стратегии развития воспитания в Российской Федерации на период до 2025 года (Утверждена распоряжением Правительства РФ от 29.05.2015 № 996- 4 р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»);

– Концепции развития дополнительного образования детей до 2030 года, утвержденной распоряжением Правительства Российской Федерации от 31 марта 2022 г. № 678-р.

– Устава МБУДО «ДДТ «Дриада», локальных нормативных актов.

2. Новизна, актуальность, педагогическая целесообразность программы.

Данная программа составлена с учётом современных потребностей рынка в специалистах в области информационных технологий, особенно в области программирования. Научившись программировать на языке Python, учащиеся получают мощный и удобный инструмент для решения как учебных, так и прикладных задач.

Знания и умения, приобретенные в результате освоения курса, могут быть использованы обучающимися при сдаче ЕГЭ, при 5 участии в олимпиадах по программированию, при решении задач по физике, химии, биологии, лингвистике и другим наукам, а также они являются фундаментом для дальнейшего совершенствования мастерства программирования.

Особенность программы «Программирование на Python» заключается в изучении основ программирования на языке Python, основных приёмов написания программ на современном языке программирования, развитие алгоритмического мышления учащихся, творческих способностей, аналитических и логических компетенций.

Направленность программы: техническая.

3. Адресат программы.

Адресатом программы являются дети в возрасте от 14 до 17 лет. Содержание и объем стартовых знаний, необходимых для начального этапа освоения программы: понимание основных алгоритмических структур.

4. Срок реализации программы.

Срок реализации программы составляет 1 год.

Уровень программы: базовый.

Программа 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.

5. Форма реализации программы.

Форма обучения – очная.

Образовательные технологии: информационные технологии, проектная технология, здоровьесберегающие технологии, технология проблемного обучения.

Форма организации содержания и процесса педагогической деятельности – комплексная.

Тип организации работы учеников: групповая работа, индивидуальная, коллективная.

Виды занятий: лекции и практические занятия.

Наполняемость группы: от 10 до 12 человек.

При сетевой форме реализации программы дополнительно заключается договор о сетевом взаимодействии, в котором закрепляется правовой статус сторон и условия реализации программы.

6. Объём программы и режим работы.

Объем программы: 72 часа.

Режим занятий: 2 академических часа в неделю.

Продолжительность часа – 40 минут.

7. Цель программы.

Целью программы является создание условий для изучения основных парадигм программирования и внедрения библиотек для работы с базами данных и визуализацией информации.

8. Задачи программы.

Программа направлена на решение следующих задач:

- изучение принципов и методов объектно-ориентированного программирования;

- изучение базовых и сложных структур данных;
- изучение способов организаций функций;
- изучение основ реляционных баз данных;
- изучение способов визуализации информации;
- формирование навыков работы с базами данных;
- развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
- способствовать формированию познавательных и регулятивных универсальных учебных действий;
- формировать коммуникативные навыки (диалогическая и монологическая речь при защите проектов);
- развитие навыков поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;
- планировать и регулировать собственную деятельность по реализации проекта, доводить начатые проекты до конца;
- формировать коммуникативные навыки (диалогическая и монологическая речь при защите проектов);
- воспитание умения работать в коллективе с учетом личностных качеств учащихся, психологических и возрастных особенностей;
- развитие рефлексивную деятельность учащихся;
- воспитание трудолюбия, целеустремленности, уважения к труду;
- воспитание информационной культуры.

9. Планируемые результаты освоения программы.

В результате освоения программы, учащиеся будут:

знать:

- принципы объектно-ориентированного программирования;
- базовые и сложные структуры данным;
- способы организации функций в языке программирования Python.

Уметь:

- разрабатывать программы для решения задач в рамках ООП;
- использовать инструменты визуализации;
- разрабатывать приложения для работы с реляционными базами данных;
- планировать и регулировать собственную деятельность по реализации проекта, доводить начатые проекты до конца;

- самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата;

- критически оценивать правильность решения задачи;

- корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями.

Владеть:

- навыками решения задач, построения алгоритмов решения;

- навыками разработки эффективных алгоритмов и программ на основе изучения языка программирования Python.

Метапредметные результаты:

- развитие навыков алгоритмического и логического мышления, грамотной разработки программ;

- развитие навыков поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;

- развитие коммуникативных навыков и навыков работы в группе.

Личностные результаты:

– воспитание умения работать в коллективе с учетом личностных качеств учащихся, психологических и возрастных особенностей;

– развитие рефлексивной деятельности учащихся;

– воспитание познавательных и регулятивных универсальных учебных действий;

– воспитание трудолюбия, целеустремленности, уважения к труду;

– воспитание информационной культуры.

10. Формы представления результатов.

Формы аттестации: беседа, демонстрация решения, наблюдение, защита проектов.

Входной контроль осуществляется в начале реализации программы в форме беседы и наблюдения и имеет диагностические задачи. Цель входной диагностики – зафиксировать начальный уровень подготовки обучающихся, имеющиеся знания, умения и навыки, связанные с предстоящей деятельностью. Наблюдение осуществляется в течение реализации программы.

Промежуточный контроль осуществляется в целях диагностики теоретических знаний и практических умений и навыков по итогам освоения одного из разделов курса. Проводится в форме тестирования.

Итоговый контроль проводится с целью определения степени достижения результатов обучения и получения сведений для совершенствования программы и методов обучения – представляет из себя защиту проекта.

11. Оценочные материалы, формирующие систему оценивания.

Промежуточный контроль осуществляется в целях диагностики теоретических знаний и практических умений и навыков по итогам освоения одного из разделов курса. Проводится в форме тестирования. Максимальный балл за промежуточное тестирование: 55 баллов. Оценивание тестирования осуществляется по следующим уровням: высокий уровень – учащийся набрал не менее 80% от максимально возможного количества баллов (от 44 балла). средний уровень – учащийся набрал не менее 50% от максимально возможного количества баллов (от 27 баллов). низкий уровень – учащийся набрал менее 50% от максимально возможного количества баллов (менее 27 баллов).

Итоговый контроль представляет из себя защиту проекта. Проект представляет собой программную реализацию на языке программирования Python, по предложенной теме. В проекте реализуются основные функции автоматизирующие процессы предметной области и направленные на оптимизацию рабочих процессов.

Критерии оценки проекта

№	Название критерия	Максимальный балл
1	Актуальность и проработанность проблемы	До 5 баллов
2	Четкость формулировки целей и задач	До 5 баллов
3	Технологическая сложность проекта. Наличие нестандартных структур данных и алгоритмов, специфика которых направлена на решение конкретной задачи.	До 10 баллов
4	Новизна и оригинальность решения	До 5 баллов
5	Качество разработанного продукта. Наличие удобного и понятного интерфейса, как абстракции между реализацией и пользователем.	До 5 баллов
6	Защита проекта: <ul style="list-style-type: none"> – качество презентации; – четкость и ясность изложения, умение взаимодействовать с аудиторией, отвечать на вопросы 	До 5 баллов

7	Наличие самооценки и перспектив дальнейшей разработки проекта	До 5 баллов
	ИТОГО	40 баллов

Оценивание проекта осуществляется по следующим уровням:

Высокий уровень – учащийся набрал не менее 32 баллов по итогам защиты проекта.

Средний уровень – учащийся набрал от 20 до 31 балла по итогам защиты проекта.

Низкий уровень – учащийся набрал менее 20 баллов по итогам защиты проекта.

Общими критериями оценки результативности обучения являются:

- оценка уровня теоретических знаний: свобода восприятия теоретической информации, осмысленность и свобода использования специальной терминологии, свобода ориентации в теоретическом материале;

- оценка уровня практической подготовки: соответствие развития уровня практических умений и навыков программным требованиям, свобода владения специальным оборудованием и оснащением;

- оценка уровня достижения личностных результатов: культура организации самостоятельной деятельности, культура работы с информацией, аккуратность и ответственность при работе.

Оценка итоговых результатов освоения программы осуществляется по трем уровням:

Высокий уровень – достижение 80- 100% показателей освоения программы.

Средний уровень – достижение 50- 79% показателей освоения программы.

Низкий уровень – достижение менее чем 50% показателей освоения программы.

Достигнутые обучающимся знания, умения и навыки заносятся в сводную таблицу результатов обучения.

Оценка уровней освоения программы

Уровни	Параметры	Показатели
Высокий уровень (80-100%)	Теоретические знания	Обучающийся освоил материал в полном объеме. Знает и понимает значение терминов, употребляет их осознанно и в полном соответствии с содержанием. Самостоятельно ориентируется в содержании материала по темам.
	Практические умения и навыки	Обучающийся овладел 80-100% умений и навыков, предусмотренных программой за конкретный период. Умет работать самостоятельно, применяя практические умения и навыки. Правильно и по назначению применяет инструменты. Умеет выполнять основные логические действия (анализ, синтез, установление причинно-

		следственных связей). Способен планировать и регулировать свою деятельность по реализации проекта. Умеет осуществлять поиск информации, в том числе в сети Интернет; выслушивать собеседника и вести диалог; выражать свои мысли в соответствии с задачами и условиями коммуникации.
	Личностные результаты	Обучающийся обладает внутренней мотивацией. Способен самостоятельно организовать собственную деятельность. Сформирована культура работы с информацией. Работу выполняет аккуратно, доводит до конца. Может оценить результаты выполнения своего задания и дать оценку работы своего товарища.
Средний уровень (50-79%)	Теоретические знания	Учащийся освоил базовые знания, ориентируется в содержании материала по темам, иногда обращается за помощью к педагогу. Использует специальную терминологию, однако сочетает её с бытовой.
	Практические умения и навыки	Владеет базовыми навыками и умениями, но не всегда может выполнить задание самостоятельно, просит помощи педагога. В основном выполняет задания на основе образца. Способен разработать проект с помощью преподавателя. Встречаются отдельные случаи неправильного применения инструментов. Делает ошибки в работе, но может устранить их после наводящих вопросов или самостоятельно. Испытывает незначительные затруднения при выполнении основных логических действий (анализ, синтез, установление причинно-следственных связей). Способен планировать и регулировать свою деятельность по реализации проекта с помощью педагога. Испытывает незначительные сложности в осуществлении коммуникации с педагогом и сверстниками.
	Личностные результаты	Внутренняя мотивация к обучению сочетается с внешней. В работе допускает небрежность. Работу не всегда выполняет аккуратно и/или доводит до конца. Оценить результаты своей деятельности может с подсказкой педагога
Низкий уровень (меньше 50%)	Теоретические знания	Владеет минимальными знаниями, ориентируется в содержании материала по темам только с помощью педагога. Избегает употреблять специальные термины.
	Практические умения и навыки	Владеет минимальными начальными навыками и умениями. Учащийся способен выполнять каждую операцию только с подсказкой педагога или товарищей. Часто неправильно применяет необходимый инструмент или не использует его вовсе. В работе допускает грубые ошибки, не может их найти даже после указания преподавателя. В состоянии выполнять лишь простейшие практические задания педагога. Испытывает существенные затруднения при выполнении

		основных логических действий (анализ, синтез, установление причинно-следственных связей). Не способен планировать и регулировать свою деятельность по реализации проекта. Испытывает значительные сложности в осуществлении коммуникации с педагогом и сверстниками.
	Личностные результаты	Преобладает внешняя мотивация к обучению. Работу часто выполняет неаккуратно и/или не доводит до конца. Не способен самостоятельно и объективно оценить результаты своей работы.

Сводная таблица результатов обучения по программе

№ п/п	ФИО	Оценка теоретических знаний	Оценка практических умений и навыков (предметных и метапредметных)	Личностные результаты	Итоговая оценка
1					
2					
3					

УЧЕБНЫЙ ПЛАН

№	Название разделов, тем	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1.	Актуализация знаний языка Python	8	2	6	Опрос
2.	Работа с операционной системой	12	4	8	Опрос
3.	Объектно-ориентированное программирование	12	4	8	Опрос
4.	Работа с базами данных	16	4	12	Практическая работа
5.	Визуализация данных	12	4	8	Опрос
6.	Создание проекта	12	2	10	Демонстрация решения
	ИТОГО	72	20	52	

СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА

Раздел 1. Актуализация знаний языка Python

1.1. Решение задач с использованием стандартных алгоритмических структур

Теория. Инструктаж о правилах поведения на занятиях и технике безопасности. Вредные и опасные факторы во время работы в компьютерном классе. Повторение основных базовых алгоритмических структур, конструкций и правил работы с ними.

Практика. Решение задач по теме «Алгоритмические структуры»

1.2. Решение задач с использованием списков

Теория. Повторение списков и правил работы с ними.

Практика. Решение задач по теме «Списки»

1.3. Решение задач с использованием строк

Теория. Повторение строк и правил работы с ними.

Практика. Решение задач по теме «Строки»

1.4. Решение задач с использованием функций

Теория. Повторение функций и правил работы с ними.

Практика. Решение задач по теме «Функции»

Раздел 2. Работа с операционной системой

2.1. Операции с директориями. Абсолютный и относительный пути

Теория. Знакомство с библиотеками `os`, `sys`. Обзор основных функций для работы с директориями

Практика. Создание, копирование, удаление и перемещение файлов и каталогов

2.2. Работа с CSV-файлами

Теория. Знакомство с CSV-файлами. Методы по работе с CSV-файлами.

Практика. Создание, запись и чтение из CSV-файла.

2.3. Работа с PDF-файлами

Теория. Знакомство с PDF-файлами. Методы по работе с PDF-файлами.

Практика. Создание, запись и чтение из PDF-файла.

2.4. Работа с DOCX-файлами

Теория. Знакомство с DOCX-файлами. Методы по работе с DOCX-файлами.

Практика. Создание, запись и чтение из DOCX-файла.

2.5. Работа с JSON-файлами

Теория. Знакомство с JSON-файлами. Методы по работе с JSON-файлами.

Практика. Создание, запись и чтение из JSON-файла.

2.6. Подпроцессы. Запуск приложений из скрипта

Теория. Обзор библиотеки `subprocess`. Работа с процессами. Запуск процессов из скрипта

Практика. Запуск стандартных приложений из Python-скрипта по условию

2.7. Самостоятельное решение задач по теме работа с ОС

Практика. Самостоятельная демонстрация полученных навыков по работе с операционной системой

Раздел 3. Объектно-ориентированное программирование

3.1. ООП. Основные понятия

Теория. Объектно-ориентированное программирование. История появления.
Обоснование использования. Основные парадигмы.

3.2. ООП. Наследование

Теория. Знакомство с принципом наследования и правилами его использования

Практика. Создание наследуемого класса, вызов родительских методов.

3.3. ООП. Инкапсуляция

Теория. Знакомство с принципом инкапсуляции и правилами его использования

Практика. Создание приватных полей, метод `__init__`

3.4. ООП. Полиморфизм

Теория. Знакомство с принципом полиморфизм и правилами его использования

Практика. Создание методов и их перегрузка.

3.5. Специальные методы для работы с классами. Метакласс

Теория. Обзор стандартных методов класса. Понятие метакласса

Практика. Создание класса, метод `__super__`

3.6. Функция как объект. Декораторы

Теория. Представление функции как объекта. Декораторы

Практика. Передача функции как параметр. Использование декораторов на простейших функциях

3.7. Аттестация по пройденным темам

Практика. Тестирование по темам предыдущих разделов

Раздел 4. Работа с базами данных

4.1. Основы теории реляционных баз данных. Синтаксис языка SQL

Теория. Базы данных. Принцип работы реляционных БД. Язык запросов SQL

Практика. Синтаксис и стиль языка SQL. Функции по созданию базы данных и таблиц в БД.

4.2. Функции языка SQL

Теория. Функции CONCAT, SUBSTRING, DESCRIBE

Практика. Использование функций в реальных запросах

4.3. Агрегатные функции SQL

Теория. Функции MAX, MIN, AVG, SUM

Практика. Использование функций в реальных запросах

4.4. Основы языка SQL. CRUD-функции

Теория. Запросы по созданию, выбору, редактированию и удалению записей

Практика (4 часа). Разработка CRUD-функций

4.5. Реализация базы данных

Теория. Разбор примера работы с базой данных

4.6 Реализация вывода информации из БД

Практика. Написание SELECT-запросов для вывода информации из базы данных

4.7 Реализация добавления информации в БД

Практика. Написание INSERT-запросов для добавления информации в базу данных

4.8 Реализация редактирования информации в БД

Практика. Написание UPDATE-запросов для редактирования информации в базе данных

4.9 Реализация добавления информации в БД

Практика. Написание DELETE-запросов для удаления информации из базы данных

4.10. Самостоятельное решение задач по теме БД

Практика. Самостоятельная демонстрация полученных навыков по работе с базами данных

Раздел 5. Визуализация данных

5.1. Типы визуализации данных и их принадлежность

Теория. Обзор стандартных библиотек для визуализации данных

5.2. Реализация визуализации данных

Теория. Разбор примера визуализации данных

Практика. Самостоятельная реализация приложения с визуализацией

5.3. Визуализация данных из файла

Практика. Создание приложения, строящего визуализацию на основе данных из файла

5.4. Визуализация данных из БД

Практика. Создание приложения, строящего визуализацию на основе данных из базы данных

5.5. Самостоятельное решение задач по теме визуализация данных

Практика. Самостоятельная демонстрация полученных навыков по работе с визуализацией данных

Раздел 6. Создание проекта

6.1. Определение и согласование темы и цели итогового проекта

Теория. Введение в проектную деятельность. Цели, задачи проекта.

Практика. Выбор темы, определение целей и задач проекта.

6.2. Планирование работы над итоговым проектом

Теория. Этапы работы над проектом. Правила эффективного планирования.

6.3. Дизайн итогового проекта

Практика. Создание интуитивно понятного интерфейса приложения.

6.4. Создание макетов приложения

Теория. Правила создания дружественного интерфейса. Правила гармоничного расположения элементов.

Практика. Создание макетов приложения.

6.5. Программирование логики программ.

Практика. Программирование функционала программ для итоговых проектов.

6.6. Тестирование приложения

Практика. Тестирование приложения на наличие ошибок

6.7. Подготовка презентаций

Теория. Правила создания презентаций

Практика. Подготовка презентации для защиты итогового проекта.

6.8. Подготовка речи

Практика. Подготовка речи и контрольного примера для защиты проекта

6.9. Подготовка к защите программы

Практика. Проверка работоспособности программного продукта, финальные правки

6.10. Защита итоговых проектов

Практика. Защита проекта.

6.11. Доработка проектов

Практика. Доработка проектов в соответствии с рекомендациями, полученными обучающимися на защите проектов.

КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

Материально-техническое обеспечение

Комплекс условий реализации программы:

Аппаратное и техническое обеспечение:

- а) Рабочее место учащегося - ноутбук с выходом в сеть Интернет;
- б) Рабочее место наставника - ноутбук с выходом в сеть Интернет; - технические средства обучения (мультимедийное устройство).

Методическое обеспечение программы

Методы обучения:

- объяснительно-иллюстративный (беседы, объяснения);
- репродуктивный (деятельность обучаемых носит алгоритмический характер, выполняется по инструкциям, предписаниям, правилам в аналогичных, сходных с показанным образцом ситуациях);

- метод проблемного изложения;
- эвристический (метод обучения заключается в организации активного поиска решения выдвинутых в обучении (или самостоятельно сформулированных) познавательных задач в ходе подготовки и реализации творческих проектов);
- исследовательский.

В процессе обучения учащиеся работают со средами программирования, поддерживающими язык Python, текстовыми редакторами, тренажерами быстрого набора текста, табличными редакторами.

Педагогические технологии: информационные технологии, проектная технология, здоровьесберегающие технологии, технология проблемного обучения.

Проектная технология дает возможность самостоятельно конструировать свои знания, ориентироваться в информационном пространстве, развивает критическое и творческое мышление, создаёт условия для формирования и развития внутренней мотивации учащихся к более качественному овладению знаниями, повышения мыслительной активности и приобретения навыков логического мышления.

Здоровьесберегающие технологии позволяют создать максимально возможные условия для сохранения, укрепления и развития эмоционального, интеллектуального и физического здоровья, в том числе в условиях работы с компьютерной техникой.

Проблемное обучение — это тип развивающего обучения, содержание которого представлено системой проблемных задач различного уровня сложности, в процессе решения которых учащиеся овладевают новыми знаниями и способами действия, а через это происходит формирование творческих способностей: продуктивного мышления и познавательной мотивации.

Учебно-методические средства обучения:

- специализированная литература по направлению, подборка журналов,
- образцы программ и систем, выполненные обучающимися и педагогом,
- учебно-методические пособия для педагога и обучающихся, включающие дидактический, информационный, справочный материалы на различных носителях, компьютерное и видео оборудование.

Применяемое на занятиях дидактическое и учебно-методическое обеспечение включает в себя электронные учебники, справочные материалы и системы используемых Программ, Интернет, рабочие тетради обучающихся.

Кадровое обеспечение

Программу реализуют педагоги дополнительного образования МБУДО «ДЦТ «Дриада».

Информационное обеспечение

Для реализации программы планируется использование следующих информационных ресурсов:

- курс «Программирование на Python» (<https://stepik.org/course/67>);
- курс «Алгоритмы: теория и практика. Методы» (<https://stepik.org/course/217>).

СПИСОК ЛИТЕРАТУРЫ

Список литературы для педагогов

1. Бэрри П. Изучаем программирование на Python. — М., 2017. — 624 с.
2. Лутц М. Изучаем Python, пер. с англ. 3-е изд. — СПб.: Символ Плюс, 2009. — 848 с.
3. Мэттиз Э. Изучаем Python. 3-издание. — СПб.: Питер, 2021 – 511 с.
4. Стивенс Р. Алгоритмы. Теория и практическое применение. – М.: Эксмо, 2022 – 547 с.
5. Фёдоров Д. Ю. Программирование на языке высокого уровня Python: учебное пособие для прикладного бакалавриата. — М.: Издательство Юрайт, 2019. — 161 с.

Список литературы для учащихся

1. Буйначев С. К. Основы программирования на языке Python: учебное пособие. — Екатеринбург: Изд-во Урал. ун-та, 2014. — 91 с.
2. Бхаргава А. Грокаем алгоритмы: иллюстрированное пособие для программистов и любопытствующих. — СПб.: Питер, 2017. — 288 с.
3. Гэддис Т. Начинаем программировать на Python / пер. с англ. 4-е изд. — СПб.: БХВ-Петербург, 2019. — 768 с.
4. Луридаш П. Алгоритмы для начинающих: теория и практика для разработчика. — М.: Эксмо, 2018. — 608 с.
5. Мюллер Дж. Python для чайников. — СПб.: Диалектика, 2019. — 416 с.

Приложение 1

Примерные задания для промежуточной аттестации

1. Создайте скрипт, который создает директорию с вашим именем (5 баллов).
2. Создайте скрипт, который выводит в CSV-файл информацию о пяти пользователях. За информацию о пользователе следует взять: ФИО, дата рождения, мобильный телефон и email (5 баллов).
3. Создайте класс, реализующий кулинарное блюдо. Сделать поля приватными, а работу с ними через геттеры и сеттеры (10 баллов).
4. Создать класс работника, от которого будут наследоваться бухгалтер, разнорабочий и строитель. Инициализировать поля дочерних классов через родительский (15 баллов).
5. Создайте приложение, которое будет содержать информацию о цветах в саду. Информацию вводит пользователь. Информация хранится в списке классов. Информация должна выводиться в CSV-файл (20 баллов).

Примерные задания для итоговой аттестации

1. Визуализация информации о подземных толчках из базы данных.
2. Чат-бота, формирующий характеристику личности пользователя.
3. Учет заявок на поступление в ДДТ Дриада.
4. Файловый менеджер.
5. Калькулятор.
6. Графический редактор.
7. Учет посещаемости группы ДДТ Дриада
8. Викторина.
9. Чат-бот, оповещающий о занятиях.
10. Визуализация информации об успеваемости группы ДДТ Дриада.