



Автономное учреждение дополнительного образования
Ханты-Мансийского автономного округа - Югры
«Мастерская талантов «Сибериус»
(АУ ДО «Мастерская талантов «Сибериус»)

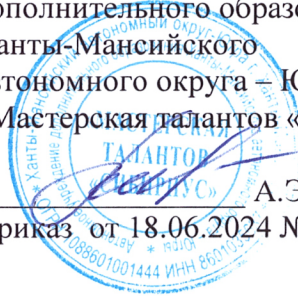
СОГЛАСОВАНО

протокол педагогического совета
автономного учреждения дополнительного
образования Ханты-Мансийского
автономного округа – Югры
«Мастерская талантов «Сибериус»
от 10.06.2024 № 6

УТВЕРЖДАЮ

Директор автономного учреждения
дополнительного образования
Ханты-Мансийского
автономного округа – Югры
«Мастерская талантов «Сибериус»

А.Э. Шишкина
приказ от 18.06.2024 № 228-о



ДОПОЛНИТЕЛЬНАЯ

ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

«Введение в программирование на Python»

Направленность: техническая

Уровень: ознакомительный

Возраст обучающихся: с 12 до 17 лет

Срок реализации программы: 4 месяца

Объем: 90 академических часов

Автор-составитель:

Буш Сергей Витальевич
педагог дополнительного образования,
АУ ДО «Мастерская талантов «Сибериус»,
Региональный центр выявления,
поддержки и развития способностей
и талантов у детей и молодежи

СОДЕРЖАНИЕ

I ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	3
1.1. Нормативно-правовая основа программы	3
1.2. Направленность программы	4
1.3. Уровень освоения программы.....	4
1.4. Актуальность, основные идеи программы	4
1.5. Новизна программы	5
1.6. Цель программы.....	6
1.7. Задачи программы.....	6
1.8. Адресат программы	7
1.9. Объем и срок реализации программы.....	7
1.10. Особенности организации учебного процесса	7
II. СОДЕРЖАНИЕ ПРОГРАММЫ	9
2.1. Календарный учебный график	9
2.2. Учебный план.....	9
2.3. Календарно-тематический план	9
2.4. Общее содержание программы	12
III. СИСТЕМА ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ.....	15
3.1. Формы и порядок проведения аттестации обучающихся	15
3.2. Планируемые результаты освоения программы	15
3.3. Оценочные материалы, формирующие систему оценивания.....	16
3.4. Форма документа фиксации образовательных результатов	17
3.5. Форма документа об обучении.....	17
IV УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ.....	17
4.1. Кадровое обеспечение программы.....	17
4.2. Методическое обеспечение программы	18
4.3. Материально-техническое обеспечение программы	18
V ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ	18
Список литературы для педагога	18
Приложение 1	19
Приложение 2.....	24

І ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1. Нормативно-правовая основа программы

Дополнительная общеобразовательная общеразвивающая программа «*Введение в программирование на Python*» (далее – программа) составлена и разработана на основе следующих нормативных документов:

1. Федеральный Закон от 29.12.2012г. № 273-ФЗ «Об образовании в Российской Федерации».

2. Концепция развития дополнительного образования детей до 2030 (распоряжение Правительства Российской Федерации от 31 марта 2022 г. № 678-р).

3. Стратегия развития воспитания в Российской Федерации на период до 2025 года (распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р).

4. Постановление Главного государственного санитарного врача РФ от 28.09.2020 N 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».

5. Постановление Главного государственного санитарного врача РФ от 28.01.2021 № 2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685- 21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» (раздел VI. Гигиенические нормативы по устройству, содержанию и режиму работы организаций воспитания и обучения, отдыха и оздоровления детей и молодежи»).

6. Паспорт федерального проекта «Успех каждого ребенка» (утвержден на заседании проектного комитета по национальному проекту «Образование» 07 декабря 2018 г., протокол № 3).

7. Приказ Минпросвещения России от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

8. Приказ Министерства образования и науки Российской Федерации от 23.08.2017 г. № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ».

9. Приказ Министерства труда и социальной защиты Российской Федерации от 22.09.2021 № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых» (Зарегистрировано в Минюсте России 17.12.2021 № 66403).

10. Приказ Департамента образования и науки Ханты-Мансийского автономного округа – Югры от 04.07.2023 г. №10-П-1649 «О внесении изменений в приказ Департамента образования и науки Ханты-

Мансийского автономного округа – Югры от 04.08.2016 г. №1224 «Об утверждении Правил персонифицированного финансирования дополнительного образования детей в Ханты-Мансийском автономном округе – Югре».

11. Приказ Министерства образования и науки Российской Федерации и министерства просвещения Российской Федерации от 05.08.2020 г. № 882/391 «Об организации и осуществлении образовательной деятельности по сетевой форме реализации образовательных программ»

12. Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (разработанные Минобрнауки России совместно с ГАОУ ВО «Московский государственный педагогический университет», ФГАУ «Федеральный институт развития образования», АНО ДПО «Открытое образование», 2015 г.) (Письмо Министерства образования и науки РФ от 18.11.2015 № 09-3242).

13. Устав автономного учреждения дополнительного образования Ханты-Мансийского автономного округа – Югры «Мастерская талантов «Сибирюс».

14. Положение о структуре, порядке разработки и утверждения дополнительных общеобразовательных общеразвивающих программ автономного учреждения дополнительного образования Ханты-Мансийского автономного округа – Югры «Мастерская талантов «Сибирюс».

1.2. Направленность программы

Дополнительная общеобразовательная общеразвивающая программа «*Введение в программирование на Python*» по содержанию является программой технической направленности, с уклоном на удовлетворение познавательного интереса обучающихся, расширение их информированности в области IT.

1.3. Уровень освоения программы

Дополнительная общеобразовательная общеразвивающая программа «*Введение в программирование на Python*» относится к ознакомительному уровню.

1.4. Актуальность, основные идеи программы

Актуальность программы «*Введение в программирование на Python*» заключается в том, что реализация данной программы создает условия для формирования у обучающихся навыков программирования, что становится всё более востребованным в различных областях. Программа способствует

приобретению знаний в области компьютерных технологий, робототехники, системного анализа и обработки данных, а также развития навыков работы с современными инструментами и библиотеками программирования, такими как `numpy`, `matplotlib` и др.

В рамках реализации программы также создаются условия для вовлечения обучающихся в проекты и исследования, направленные на разработку инновационных решений и совершенствование существующих технологий в областях, где используется язык Python. Это способствует развитию критического мышления, креативности и предпринимательских навыков, а также обеспечивает подготовку высококвалифицированных специалистов, готовых к работе в быстро меняющейся технологической среде.

Таким образом, программа «*Введение в программирование на Python*» актуальна и важна для подготовки будущих профессионалов, способных эффективно использовать современные технологии и решать актуальные задачи в различных областях с применением Python.

1.5. Новизна программы

Новизна программы заключается в интеграции передовых технологий и методик обучения, направленных на развитие ключевых компетенций у учащихся. Программа выделяется тем, что акцентирует внимание на практическом применении знаний в области программирования, что включает использование современных инструментов и технологий для решения реальных задач.

Основная особенность подхода к реализации программы заключается в создании интерактивной образовательной среды, где учащиеся погружаются в процесс проектной деятельности, работающей с реальными сценариями и задачами. Это позволяет не только освоить теоретические основы, но и применять их на практике, что способствует формированию алгоритмического и системного мышления.

Программа охватывает разнообразные аспекты, связанные с IT и обработкой данных, что позволяет учащимся получать комплексные знания и навыки. Используемые в программе методы, такие как кейсовый метод и работа с различными уровнями ограничений, способствуют развитию критического мышления, умения анализировать информацию, аргументировать собственные выводы и предлагать эффективные решения задач.

Программа также предполагает регулярную оценку результатов работы учащихся, что позволяет отслеживать динамику усвоения знаний и улучшать образовательный процесс. Постоянное взаимодействие с высокотехнологичным оборудованием и работа в команде обеспечивают развитие как *hard-*, так и *soft-*компетенций, необходимых для успешной профессиональной деятельности в области программирования.

1.6. Цель программы

Целью программы является создание условий для выявления, обучения и развития, поддержки и сопровождения одаренных детей, их самореализации, а также создание условий для всестороннего развития личности обучающихся, формирование у них базовых знаний и навыков, необходимых для разработки программного обеспечения. Программа нацелена на освоение основ программирования, а также на развитие умений по созданию и тестированию программных решений для решения различных задач в разных сферах деятельности.

1.7. Задачи программы

Предметные:

- изучить базовые концепции программирования на Python, такие как синтаксис, структуры данных, алгоритмы;
- познакомить с основами работы с базами данных и файловыми системами;
- научить разрабатывать алгоритмы для решения задач при помощи Python.

Метапредметные:

- повышение мотивации к изучению предмета;
- развитие вариативного подхода к получению данных;
- приобретение опыта использования ТРИЗ при формировании собственных идей и решений;
- формирование понимания прямой и обратной связи проекта и среды его реализации, заложение основ социальной и экологической ответственности
- приобретение и углубление знаний основ проектирования и управления проектами;
- формирование проектного мировоззрения и творческого мышления;
- развитие фантазии и профессионального мышления.

Личностные:

- развитие психофизиологических качеств учащихся: памяти, внимания, способности логически мыслить, анализировать, концентрировать внимание на главном.
- формирование творческого отношения по выполняемой работе;
- развитие творческой инициативы и самостоятельности;
- воспитание уважения к чужому мнению;
- воспитание умения работать в коллективе и самостоятельно.

1.8. Адресат программы

Программа адресована детям возраста 12-17 лет.

Набор детей осуществляется независимо от их способностей и умений, без специального отбора. Для обучения по программе важен интерес ребенка к технической деятельности, а также желание изучать программирование.

1.9. Объем и срок реализации программы

Программа реализуется в 2024 учебном году, рассчитана на 4 месяца (сентябрь – декабрь).

Общая продолжительность реализации программы 90 академических часа.

Режим занятий, у обучающихся регламентируется календарным учебным графиком и расписанием занятий.

Единицей измерения учебного времени и основной формой организации образовательной деятельности является учебное занятие.

Учебные занятия проводятся в течение всего календарного года, включая каникулярное время.

1.10. Особенности организации учебного процесса

Форма обучения – очная, с применением электронного обучения и дистанционных образовательных технологий.

Формы организации деятельности: групповые.

Виды занятий (в зависимости от целей занятия и его темы), включая учебные занятия, направленные на проведение текущего контроля освоения программы:

- групповые;
- индивидуальные;
- комбинированные (для решения нескольких учебных задач);
- контрольные мероприятия (защита проекта).

Наполняемость группы – до 20 человек.

Дистанционные занятия проводятся в пространстве для учёбы и общения – Сферум.

Единицей измерения учебного времени и основной формой организации образовательной является учебное занятие.

Продолжительность учебных занятий составляет 2 академических часа.

Для всех видов аудиторных учебных занятий академический час устанавливается продолжительностью 40 минут. Если обучение по программе реализуется с применением дистанционных образовательных технологий, академический час устанавливается продолжительностью 30

минут.

Периодичность занятий – 3 раза в неделю.

II. СОДЕРЖАНИЕ ПРОГРАММЫ

2.1. Календарный учебный график

Учебный период	
Продолжительность учебного года	Начало учебного года: <i>январь, 2024 г.</i> Окончание учебного года: <i>декабрь, 2024 г.</i>
Период реализации программы	Начало учебного года: <i>сентябрь, 2024 г.</i> Окончание учебного года: <i>декабрь, 2024 г.</i>
Количество учебных недель	<i>16 учебных недель</i>
Продолжительность учебной недели	<i>5 дней (понедельник - пятница)</i>
Промежуточная аттестация	<i>декабрь, 2024 год</i>

2.2. Учебный план

№ п/п	Наименование блоков	Количество часов			Формы аттестации/ Контроля
		Всего	Теория	Практика	
1.	Изучение программирования на Python	62	27	35	Практическая задача
2.	Решение кейсов	26	5	21	Практическая задача
3.	Промежуточная аттестация	2	0	2	Тест
	ИТОГО по программе	90	32	58	

2.3. Календарно-тематический план

№ п/п	Дата	Наименование темы	Объем часов			Форма занятия	Место проведения	Форма контроля
			Всего	В том числе				
				Теория	Практика			
1	2	3	4	5	6	7	8	9
Изучение программирования на Python								
1.	09.09.	Введение. Установка и настройка Python. Написание	2	2	0	лекция	ул. Промышленная, 19, аудитория 212	опрос

		первого скрипта						
2.	11.09, 13.09	Основы синтаксиса: переменные, типы данных, базовые операции	4	2	2	лекции/практические занятия	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
3.	16.09, 18.09	Условные операторы(if, elif, else)	4	2	2	лекции/практические занятия	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
4.	20.09, 23.09	Циклы(for, while) и их применения	4	2	2	лекции/практические занятия	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
5.	25.09, 27.09	Решение задач	4	0	4	практические занятия	ул. Промышленная, 19, аудитория 212	практическая работа
6.	30.09	Работа с функциями: определение и вызов	2	2	2	лекция/практическое занятие	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
7.	02.10	Параметры функций, возвращаемые значения, область видимости переменных	2	1	1	лекция/практическое занятие	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
8.	04.10, 07.10	Модули и пакеты, использование стандартной библиотеки	4	2	2	лекции/практические занятия	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
9.	09.10, 11.10	Создание собственных модулей и работа с ними	4	2	2	лекция/практическое занятие	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
10	14.10, 16.10	Списки: создание, доступ,	4	2	2	лекции/практические	ул. Промышленная, 19,	опрос, практическая работа

		методы и операции				занятия	аудитория 212	
11	18.10	Кортежи и множества: создание, доступ, методы и операции	2	1	1	лекция/практическое занятие	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
12	21.10	Словари: создание, доступ, методы и операции	2	1	1	лекция/практическое занятие	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
13	23.10, 25.10	Решение задач	4	0	4	практическое занятие	ул. Промышленная, 19, аудитория 212	практическая работа
14	28.10, 30.10	Работа с файлами: открытие, закрытие, чтение, запись данных	4	2	2	лекции/практические занятия	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
15	01.11, 06.11, 08.11	Основы ООП	6	3	3	лекции/практическое занятие	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
16	11.11, 13.11, 15.11	Работа с внешними библиотеками	6	3	3	лекции/практические занятия	ул. Промышленная, 19, аудитория 212	опрос, практическая работа
17	18.11, 20.11	Решение задач	4	0	4	практическое занятие	ул. Промышленная, 19, аудитория 212	практическая работа
Решение кейсов								
1.	22.11, 25.11	Кейс 1. Симуляция броска кубиков	4	1	3	практическое занятие	ул. Промышленная, 19, аудитория 212	практическое задание
2.	27.11, 29.11, 02.12	Кейс 2. Программа для анализа текстов	5	1	4	лекция	ул. Промышленная, 19, аудитория	практическое задание

							212	
3.	02.12, 04.12, 06.12	Кейс 3. Программа для анализа результатов опроса	5	1	4	лекции/ практич еские занятие	ул. Промышле нная, 19, аудитория 212	практичес кое задание
4.	09.12, 11.12, 13.12	Кейс 4. Генератор паролей	6	1	5	практич еские занятие	ул. Промышле нная, 19, аудитория 212	практичес кое задание
5.	16.12, 18.12, 20.12	Кейс 5. Простой менеджер паролей	6	1	5	лекция/п рактичес кие занятие	ул. Промышле нная, 19, аудитория 212	практичес кое задание
Промежуточная аттестация								
1.	23.12	Промежуточ ная аттестация	2	0	2	практич еское занятие	ул. Промышле нная, 19, аудитория 212	тест

2.4. Общее содержание программы

Блок 1. Изучение программирования на Python – 52 часа.

Тема 1. Введение. Установка и настройка Python. Написание первого скрипта.

Теория (1 час). Введение в предмет, и общие данные по изучаемому материалу.

Практика (1 час). Установка Python, рассмотрение IDE (PyCharm) и написание первой программы «Hello, World!».

Тема 2. Основы синтаксиса: переменные, типы данных, базовые операции.

Теория (2 часа). Изучение первых команд на Python, таких как объявление переменных, типы данных str, float, int, и арифметических операторов.

Практика (2 час). Решение простейших использованием изученных арифметических операторов, типов данных и переменных.

Тема 3. Условные операторы (if, elif, else).

Теория (2 часа). Будут изучены основы условных операторов if, elif, и else.

Практика (2 час). Учащиеся реализуют программы для проверки различных условий.

Тема 4. Циклы (for, while) и их применения.

Теория (2 часа). Изучить синтаксис циклов for и while, а также различные методы управления циклами, такие как break, continue и else.

Практика (2 часа). Решение задач, требующих итераций.

Тема 5. Решение задач.

Практика (4 часа). Решение задач по пройденным темам.

Тема 6. Работа с функциями: определение и вызов.

Теория (1 час) Синтаксис функций, параметры и возвращаемые значения.

Практика (1 час). Написание функции для выполнения конкретных задач.

Тема 7. Параметры функций, возвращаемые значения, область видимости переменных.

Теория (1 час). Изучение различных видов параметров (позиционные, ключевые, произвольные) и возвращаемые значения функций, а также область видимости переменных и их влияние на функции.

Практика (1 час). Создание функции с разными типами параметров и возвращаемыми значениями, а также реализация примеров, в которых области видимости влияют на доступ к переменным.

Тема 8. Модули и пакеты, использование стандартной библиотеки.

Теория (2 часа) Как импортировать и использовать модули и пакеты, а также как работает стандартная библиотека Python?

Практика (2 часа) Импортирование модулей из стандартной библиотеки, такие как `math` или `datetime`, и использование их функции в задачах.

Тема 9. Создание собственных модулей и работа с ними

Теория (2 час). Научиться как создавать и структурировать собственные модули, включая организацию файлов и использование директивы `import`.

Практика (2 час) Написать несколько собственных модулей с функциями и классами, затем импортировать и использовать их в других скриптах.

Тема 10. Списки: создание, доступ, методы и операции.

Теория (2 часа). Изучение первой структуры данных `list`.

Практика (2 часа). Использование списков при решении задач.

Тема 11. Кортежи и множества: создание, доступ, методы и операции.

Теория (1 час). Изучение структуры данных `tuple`.

Практика (1 час). Рассмотрение задач где можно применять кортежи.

Тема 12. Словари: создание, доступ, методы и операции.

Теория (1 час). Как создавать словари, обращаться к их элементам, а также использовать методы, такие как `keys()`, `values()`, и `items()`.

Практика (1 час). Создать словари для хранения информации, выполнять операции с ключами и значениями, а также реализовать задачи, такие как подсчет частоты слов в тексте.

Тема 13. Решение задач.

Практика (4 часа) Решение задач с использованием изученного материала.

Тема 14. Работа с файлами: открытие, закрытие, чтение, запись данных.

Теория (2 часа) Изучить методы работы с файлами, такие как `open()`, `read()`, `write()`, и `close()`.

Практика (2 часа) Выполнить запись и чтение текстового файла, и создание функций для обработки данных из файлов.

Тема 15. Основы ООП.

Теория (3 часа). Изучить концепции классов и объектов.

Практика (3 часа). Определить класс с атрибутами и методами, создать несколько объектов этого класса и взаимодействовать с ними.

Тема 16. Работа с внешними библиотеками

Теория (3 часа). Как устанавливать и обновлять библиотеки при помощи `pip`, а также как их подключать и использовать в своем коде.

Практика (3 часа). Установить библиотеки `numpy` и `pandas`, а также их подключить.

Тема 17. Решение задач

Практика (2 часа). Решение задач по пройденному материалу.

Блок 2. Решение кейсов – 26 часов.

Кейс 1. Симуляция бросков кубиков.

Описание. Разработать программу, которая имитирует бросок одного или нескольких кубиков и выводит результат. Программа должна учитывать возможность выбора типа кубика (D6, D20 и т.д.)

Теория (1 часа). Получение задания.

Практика (3 часа). Разработка алгоритма и написание программы.

Кейс 2. Программа для анализа текстов.

Описание: Создать программу, которая принимает текстовый файл, подсчитывает частоту появления каждого слова и выводит наиболее часто встречающиеся слова.

Теория (1 час). Получение задания и файлов для анализа. Разработка алгоритма решения.

Практика (4 часов). Разработка программы.

Кейс 3. Программа для анализа результатов опроса.

Описание: Создать программу, которая принимает результаты опроса в виде списка ответов, подсчитывает количество ответов каждого типа и выводит статистику (процентное соотношение каждого ответа используя диаграммы `matplotlib`).

Теория (1 час). Получение задания и файлов для анализа. Разработка алгоритма решения.

Практика (4 час). Разработка программы.

Кейс 4. Генератор паролей.

Описание: Создать программу, которая генерирует случайные пароли заданной длины. Пользователь может выбрать, какие символы включать в пароль (буквы, цифры, специальные символы).

Теория (1 час). Получение задания.

Практика (5 часа). Создание алгоритма и написание программы

Кейс 5. Простой менеджер паролей.

Описание: Создать программу, которая хранит пароли в зашифрованном виде в файле и позволяет пользователю добавлять новые пароли, получать их по ключевому слову и удалять устаревшие пароли.

Теория (1 час). Получение задания.

Практика (5 часов). Создание алгоритма и написание программы.

Промежуточная аттестация – 2 часа.

III. СИСТЕМА ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

3.1. Формы и порядок проведения аттестации обучающихся

Виды и формы контроля:

входной: предназначен для определения стартового уровня возможностей обучающихся. Проводится в форме тестирования.

промежуточный: осуществляется в процессе усвоения каждой изучаемой темы, при этом диагностируется уровень усвоения отдельных блоков программы. В процессе контроля каждого занятия создается возможность выявления уровня усвоения учебного материала, недочетов, положительных и отрицательных моментов применяемых технологий. Используются методы и формы проверки результатов: наблюдение, рефлексия, устный опрос, тестирование.

Формы подведения итогов: тестирование, рефлексия

3.2. Планируемые результаты освоения программы

Образовательные результаты.

На уровне знаний: учащиеся воспроизводят термины по тематике занятий, понятия и правила, связанные с программированием на языке Python.

На уровне понимания: учащиеся объясняют принципы базовых конструкций Python, для написания программ.

На уровне применения: учащиеся пишут программы на языке Python различной сложности.

На уровне синтеза: учащиеся могут публично выступить с докладом, рефератом, сообщением, защитой проекта или кейса; составляют схемы задач.

На уровне оценки: учащиеся оценивают соответствие выводов имеющимся данным; оценивает значимость того или иного продукта деятельности, оценивают логику построения письменного текста в электронных или бумажных образовательных ресурсах.

Предметные результаты:

знания:

- имеет базовые знания о программировании на Python.
- пишет и отлаживает простые программы на Python;
- поиск и устранение ошибок в простейшем программном коде;

Личностные результаты:

- повышение уровня мотивации к получению новых знаний и применению их на практике;
- развитие опыта участия в проектах с решением реальных задач и конкретного практического использования;
- формирование навыков командной работы и коммуникации внутри команды и вне её для достижения поставленных целей;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование ценности жизни своей и окружающих, формирование ответственного отношения к правилам техники безопасности при работе с профессиональными и бытовыми приборами и антропогенной среде.

Метапредметные результаты:

- формирование и развитие начальной профессиональной компетентности в области программирования;
- развитие коммуникативных навыков в рамках общих понятий делового и общественного этикета;
- умение критически и не предвзято оценивать результат собственной и командной работы;
- умение создавать, применять и преобразовывать условные символичные схемы в выбранной сфере деятельности;
- планирование алгоритмов своей деятельности и поведения для достижения целей;
- умение критически и не предвзято относиться к собственному мнению и проявлять аргументированную готовность к отстаиванию своей точки зрения.

3.3. Оценочные материалы, формирующие систему оценивания

Аттестационный тест состоит из 25 вопросов в электронном виде, за каждый правильный ответ 1 балл, за не верный ответ 0 баллов, тест считается успешно пройденным, если тестируемый набрал от 18 баллов и

выше. Тест с автоматическим сбором информации на основе Яндекс форм (Приложение 1).

3.4. Форма документа фиксации образовательных результатов

Результаты промежуточной аттестации фиксируются, согласно Положению «О формах, периодичности и порядке текущего контроля успеваемости, промежуточной и итоговой аттестации обучающихся в отделах развития детских технопарков «Кванториум» АУ ДО «Мастерская талантов «Сибириус», в документе «Протокол результатов аттестации обучающихся» (Приложение 4).

3.5. Форма документа об обучении

Обучающимся, освоившим дополнительную общеобразовательную программу и успешно прошедшим промежуточную аттестацию, по запросу родителей (законных представителей) выдается сертификат в электронном варианте.

IV УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

4.1. Кадровое обеспечение программы

Педагог дополнительного образования, реализующий настоящую программу, должен соответствовать профессиональному стандарту «Педагог дополнительного образования детей и взрослых», утвержденному приказом Министерства труда и социальной защиты Российской Федерации от 22.09.2021 №652 н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых».

Требования к квалификации педагога дополнительного образования, работающего по данной программе:

– высшее образование или среднее профессиональное образование в рамках укрупненных групп направлений подготовки высшего образования и специальностей среднего профессионального образования «Образование и педагогические науки»

– высшее образование либо среднее профессиональное образование в рамках иного направления подготовки высшего образования и специальностей среднего профессионального образования при условии его соответствия дополнительной общеразвивающей программе и получение при необходимости после трудоустройства дополнительного профессионального образования по направлению подготовки «Образование и педагогические науки».

К квалификационной категории по должности «педагог дополнительного образования» – требований нет. По программе могут

работать педагоги дополнительного образования высшей или первой квалификационной категории или педагоги, не имеющие квалификационной категории.

4.2. Методическое обеспечение программы

Педагогические технологии: технология группового обучения, технология коллективного взаимообучения, технология развивающего обучения, технология проблемного обучения, технология дистанционного обучения (при необходимости), технология исследовательской деятельности, технология проектной деятельности, технология развития критического мышления через дискуссии, технология решения изобретательских задач, здоровьесберегающая технология.

4.3. Материально-техническое обеспечение программы

Перечень технических средств обучения	<ul style="list-style-type: none"> • компьютер преподавателя; • графический планшет • выход в интернет
Учебный комплект на каждого обучающегося	<ul style="list-style-type: none"> • персональный компьютер с периферийным оборудованием. • специализированное программное обеспечение
Программное и информационное обеспечение	<ul style="list-style-type: none"> • Python 3.12.5 • редактор кода PyCharm • Microsoft Word • Microsoft Excel

V ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ

Список литературы для педагога

1. Маттес Э. Курс по Python: Руководство для начинающих. — СПб.: Питер, 2019. — 544 с.
2. Свейгарт А. Автоматизация рутинных задач с помощью Python. — СПб.: Питер, 2016. — 480 с.
3. Лутц М. Изучение Python. — СПб.: Питер, 2013. — 1600 с.
4. Макинни У. Python и анализ данных. — СПб.: Питер, 2017. — 512 с.
5. Рамальо Л. Эффективный Python: Написание устойчивого, читаемого кода. — СПб.: Питер, 2016. — 792 с.

Тест по основам Python

1. Какой метод используется для добавления элемента в конец списка?
A) append()
B) add()
C) insert()
D) extend()
2. Какой оператор используется для проверки равенства двух значений?
A) =
B) ==
C) !=
D) <=
3. Что выведет следующий код?
x = [1, 2, 3]
y = x
y.append(4)
print(x)
A) [1, 2, 3]
B) [1, 2, 3, 4]
C) [1, 2, 4]
D) [4, 1, 2, 3]
4. Какой результат выполнения следующего кода?
print(3 * 'ab')
A) ababab
B) 3ab
C) ababab
D) ab3
5. Какой тип данных возвращает функция input()?
A) int
B) float
C) str
D) list
6. Что делает функция len()?
A) Возвращает длину строки
B) Возвращает последний элемент списка
C) Возвращает количество уникальных элементов в списке
D) Возвращает наибольший элемент в строке
7. Какое значение будет у переменной a после выполнения следующего кода?
a = [1, 2, 3]
a[1] = 5
A) [1, 5, 3]
B) [1, 2, 5]
C) [5, 2, 3]
D) [1, 5]
8. Какой метод используется для удаления элемента из списка по значению?
A) remove()
B) pop()
C) discard()
D) del()

9. Что выведет следующий код?

```
def func(x, y=10):  
    return x + y
```

```
print(func(5))
```

- A) 5
- B) 10
- C) 15
- D) TypeError

10. Какой результат будет у следующего кода?

```
x = 'hello'  
y = x.upper()  
print(y)
```

- A) HELLO
- B) hello
- C) Hello
- D) HELLO WORLD

11. Какой оператор используется для логического "И"?

- A) &
- B) |
- C) and
- D) &&

12. Какой из следующих типов данных является изменяемым?

- A) tuple
- B) list
- C) str
- D) int

13. Какой результат выполнения следующего кода?

```
a = {1, 2, 3}  
b = {2, 3, 4}  
print(a & b)
```

- A) {1, 2, 3, 4}
- B) {1, 2}
- C) {2, 3}
- D) {3}

14. Что выведет следующий код?

```
for i in range(3):  
    print(i, end=' ')
```

- A) 0 1 2
- B) 1 2 3
- C) 0 1 2 3
- D) 0 1 2 3

15. Какой результат выполнения следующего кода?

```
a = [1, 2, 3]  
b = a.copy()  
b.append(4)  
print(a)
```

- A) [1, 2, 3, 4]
- B) [1, 2, 3]
- C) [4, 1, 2, 3]
- D) [1, 2, 3, 4, 4]

16. Какой метод используется для замены подстроки в строке?
- A) replace()
 - B) sub()
 - C) replace_sub()
 - D) substitute()
17. Какой результат возвращает следующая функция?
- ```
def add(x, y=5):
 return x + y

print(add(3, 2))
```
- A) 5
  - B) 7
  - C) 8
  - D) 3
18. Какой оператор используется для проверки истинности логического выражения?
- A) is
  - B) ==
  - C) or
  - D) not
19. Какое значение будет у переменной x после выполнения следующего кода?
- ```
x = [1, 2, 3]  
x *= 2
```
- A) [1, 2, 3, 1, 2, 3]
 - B) [1, 2, 3, 2, 3]
 - C) [2, 3, 4, 5]
 - D) [2, 3, 4]
20. Какой метод используется для получения ключей словаря?
- A) keys()
 - B) values()
 - C) items()
 - D) get()
21. Какой результат возвращает функция range(1, 5)?
- A) [1, 2, 3, 4, 5]
 - B) (1, 2, 3, 4)
 - C) [1, 2, 3, 4]
 - D) range(1, 5)
22. Какой метод используется для добавления элемента в множество?
- A) add()
 - B) insert()
 - C) append()
 - D) extend()
23. Какой оператор используется для проверки, что элемент находится в коллекции?
- A) in
 - B) not in
 - C) exists
 - D) contains
24. Что выведет следующий код?
- ```
def func(a, b=[]):
 b.append(a)
 return b
```

```
print(func(1))
print(func(2, []))
print(func(3))
```

- A) [1] [2] [3]
- B) [1] [2] [1, 3]
- C) [1, 3] [2] [1, 3]
- D) [1] [2] [1, 3]

25. Какой из следующих методов используется для конвертации строки в нижний регистр?

- A) lower()
- B) upper()
- C) capitalize()
- D) title()

## Ответы:

1. A) `append()`
2. B) `==`
3. B) `[1, 2, 3, 4]`
4. A) `ababab`
5. C) `str`
6. A) Возвращает длину строки
7. A) `[1, 5, 3]`
8. A) `remove()`
9. C) `15`
10. A) `HELLO`
11. C) `and`
12. B) `list`
13. C) `{2, 3}`
14. A) `0 1 2`
15. B) `[1, 2, 3]`
16. A) `replace()`
17. B) `7`
18. C) `or`
19. A) `[1, 2, 3, 1, 2, 3]`
20. A) `keys()`
21. D) `range(1, 5)`
22. A) `add()`
23. A) `in`
24. B) `[1] [2] [1, 3]`
25. A) `lower()`

**Протокол результатов  
промежуточной аттестации обучающихся**

Направление \_\_\_\_\_

Ф.И.О. педагога \_\_\_\_\_

Наименование дополнительной  
общеразвивающей программы \_\_\_\_\_

Группа \_\_\_\_\_

Количество детей \_\_\_\_\_

Дата проведения \_\_\_\_\_

**Результаты промежуточной аттестации**

| <b>№</b> | <b>ФИО обучающегося</b> | <b>Форма проведения<br/>аттестации</b> | <b>Уровень</b> |
|----------|-------------------------|----------------------------------------|----------------|
| 1        |                         |                                        |                |
| 2        |                         |                                        |                |
| 3        |                         |                                        |                |
| 4        |                         |                                        |                |
| 5        |                         |                                        |                |
| 6        |                         |                                        |                |
| 7        |                         |                                        |                |
| 8        |                         |                                        |                |
| 9        |                         |                                        |                |
| 10       |                         |                                        |                |
| 11       |                         |                                        |                |
| 12       |                         |                                        |                |

Подпись педагога \_\_\_\_\_