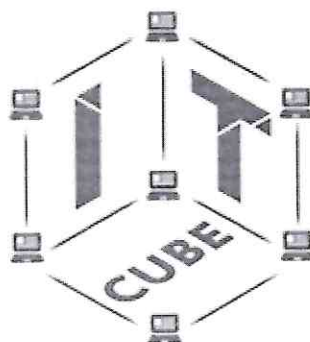


Министерство образования и науки Мурманской области
Государственное автономное профессиональное образовательное учреждение
Мурманской области «Мурманский колледж экономики и информационных технологий»

Центр цифрового образования детей

IT – CUBE.МУРМАНСК



СЕТЬ ЦЕНТРОВ ЦИФРОВОГО
ОБРАЗОВАНИЯ ДЕТЕЙ «IT-КУБ»

Дополнительная общеобразовательная общеразвивающая программа

ПРОГРАММИРОВАНИЕ НА PYTHON

Направленность: инженерно-техническая

Срок реализации: 1 год

Возраст детей: 14-16 лет

г. Мурманск

2021 г.

Программа разработана на основе рекомендаций Министерства просвещения Российской Федерации по реализации дополнительной общеобразовательной программы по направлению «Программирование на Python». Программа разработана для школьников от 14 до 16 лет.

Организация-разработчик: Центр цифрового образования детей IT - CUBE.МУРМАНСК

Разработчик: Егорычева С. В. , преподаватель ГАПОУ МО «МКЭиИТ»,
Кузнецова К. В., преподаватель ГАПОУ МО «МКЭиИТ»

Рабочая программа рассмотрена методическим советом ГАПОУ МО «МКЭиИТ».

Рабочая программа утверждена приказом от «30» 08 2021 г. № 98

Директор ГАПОУ МО «МКЭиИТ» _____ (В.В. Малков)



СОДЕРЖАНИЕ

1 Пояснительная записка.....	4
2 Учебно-тематический план	13
3 Содержание программы	15
4 Методическое обеспечение программы	17
5 Диагностика эффективности образовательного процесса.....	18
6 Материально-техническое обеспечение	24
Список рекомендуемой литературы	22

1 Пояснительная записка

Введение: В настоящее время мы переживаем большие изменения в развитии общества. В современную жизнь человека все больше и больше внедряются компьютеры и информационные технологии. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причем зачастую не на пользовательском уровне, а на уровне начинающего программиста.

В обязательном школьном курсе информатики программирование нередко представлено лишь на элементарном уровне, на это выделяется недостаточное количество часов. Лишь немногие школы могут себе позволить преподавать программирование на достойном уровне. Следствием этого является формальное восприятие учащимися основ современного программирования и неумение применять полученные знания на практике.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке. В данной программе выбран язык программирования Python. Данный выбор обусловлен тем, что синтаксис языка достаточно прост и интуитивно понятен, а это 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.

Программа разработана на основе:

- Конституции Российской Федерации (принята всенародным голосованием 12.12.1993 с изменениями, одобренными в ходе общероссийского голосования 01.07.2020),

- Федерального закона от 29.12.2012 № 273-ФЗ (ред. от 31.07.2020) «Об образовании в Российской Федерации» (с изм. и доп., вступ. в силу с 01.09.2020),

- Паспорта национального проекта «Образование» (утв. президиумом Совета при Президенте РФ по стратегическому развитию и национальным проектам, протокол от 24.12.2018 № 16),

- Государственной программы Российской Федерации «Развитие образования» (Утверждена Постановлением Правительства РФ от 26.12.2017 № 1642 (ред. от 22.02.2021) «Об утверждении государственной программы Российской Федерации «Развитие образования»,

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

- Концепции развития дополнительного образования, утвержденной распоряжением Правительства Российской Федерации от 4 сентября 2014 г. № 1726-р,

- Методических рекомендаций по созданию и функционированию центров цифрового образования «IT-куб» (утверждены распоряжением Министерства просвещения Российской Федерации от 12 января 2021 г. № Р-5),

- Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденным приказом Министерства просвещения Российской Федерации № 196 от 09.11.2018г,

Постановления 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».

Направленность программы: инженерно-техническая

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 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.

Цель программы: создание условий для изучения методов программирования на языке Python; рассмотрение различных парадигм программирования, предлагаемых этим языком (процедурная, функциональная, объектно-ориентированная); подготовка к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях.

Задачи:

Обучающие:

- знакомство с принципами и методами фундаментального программирования;
- знакомство с принципами и методами объектно-ориентированного программирования;
- формирование навыков работы в интегрированной среде разработки на языке Python;
- изучение конструкции языка программирования Python;
- формирование навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;

- формирование представления о значении информационных технологий в развитии общества и в изменении характера труда человека.

Развивающие:

- 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 познавательных способностей ребенка, памяти, внимания, пространственного мышления, аккуратности и изобретательности;

- формирование творческого подхода к поставленной задаче;

- развитие навыков инженерного мышления, умения работать как по предложенным инструкциям, так и находить свои собственные пути решения поставленных задач;

- развитие навыков эффективной деятельности в проекте;

- развитие стрессоустойчивости;

- развитие способностей к самоанализу, самопознанию;

- формирование навыков рефлексивной деятельности.

Воспитательные:

- воспитание мотивации учащихся к изобретательству, созданию собственных программных реализаций;

- формирование стремления к получению качественного законченного результата в проектной деятельности;

- формирование информационной культуры: ответственного отношения к информации с учетом правовых и этических аспектов её распространения, избирательного отношения к полученной информации;

- формирование правильного восприятия системы ценностей, принципов, правил информационного общества;

- формирование потребностей в самостоятельном приобретении и применении знаний, потребности к постоянному саморазвитию;
- воспитание социально-значимых качеств личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность.

Отличительные особенности программы. 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.

Уровень программы: базовый

Возраст обучающихся, участвующих в реализации программы: 14-16 лет

Содержание и объем стартовых знаний, необходимых для начального этапа освоения программы: особых знаний не требуется

Срок реализации программы (модуля): 1 год

Объем программы: 144 часа

Режим занятий: 2 раза в неделю по 2 академических часа.

Формы организации учебной деятельности: групповая, индивидуальная, коллективная.

Количество обучающихся в группе: 10-12 человек.

Прогнозируемые результаты и способы их проверки

Личностные результаты:

- 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.

Метапредметные результаты:

- 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.

Предметные результаты:

В результате освоения программы обучающиеся должны

Знать:

- правила работы с компьютером и технику безопасности;
- основные предметные понятия («информация», «алгоритм», «исполнитель», «модель») и их свойства;
- основы современных языков программирования;
- особенности работы с интегрированной средой разработки;
- базовые и сложные конструкции, способы организации процедур и функций в языке программирования Python;
- технику ведения проектной деятельности и принципов тайм-менеджмента.

Уметь:

- работать с информацией: находить с применением правил поиска в компьютерных сетях, оценивать и использовать информацию из различных источников при выполнении заданий и проектов по различным темам;

- определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей, создавать на их основе программы анализа данных, читать и понимать программы, написанные на языке программирования высокого уровня Python;

- выполнять пошагово алгоритмы управления исполнителями и анализа числовых и текстовых данных;

- разрабатывать и использовать компьютерно-математические модели, оценивать числовые параметры моделируемых объектов и процессов, интерпретировать результаты, получаемые в ходе моделирования реальных процессов, анализировать готовые модели на предмет соответствия реальному объекту или процессу;

- самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;

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

- критически оценивать правильность решения учебно-исследовательской задачи;

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

- организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Владеть:

- навыками исследовательской, проектной и социальной деятельности, строить логическое доказательство;

- навыками использования, создания и преобразования различных символьных записей, схем и моделей для решения познавательных и учебных

задач в различных предметных областях, исследовательской и проектной деятельности;

- навыками разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- навыками использования специальных средств и библиотек языка Python
- навыками работы в интегрированной среде разработки на языке программирования Python;
- навыками алгоритмического и логического мышления, грамотной разработки программ;
- навыками проектирования, разработки, документирования и представления собственных проектов;
- навыками самообразования - периодической оценкой своих успехов и собственной работы самими обучающимися..

Формы подведения итогов реализации дополнительной программы

Формы аттестации: наблюдение, опрос, защита проекта, беседа, тест, участие в мероприятия различного уровня.

В конце каждого раздела проводится промежуточное тестирование. Итоговый контроль проводится в конце года с целью определения степени достижения результатов обучения и получения сведений для совершенствования программы и методов обучения.

Итоговая оценка развития личностных качеств воспитанника производится по трём уровням:

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

Результатом усвоения обучающимися программы по каждому уровню являются: устойчивый интерес к занятиям по разработке программ на языке Python, результаты достижений в массовых мероприятиях различного уровня.

2 Учебно-тематический план

№ п/п	Название разделов, тем	Количество часов			Форма аттестации/контроля
		Всего	Теория	Практика	
1	Введение в программирование.	30	8	22	
1.1	Знакомство со средой PyCharm. Понятия кода, интерпретатора, программы. Интегрированные среды, исполнение кода. Простейшие программы с выводом на экран.	4	2	2	Беседа
1.2	Условный оператор. Переменные и арифметика.	4	1	3	Опрос
1.3	Знакомство с циклом while.	4	1	3	Наблюдение
1.4	Отладчик.	2	1	1	Беседа
1.5	Знакомство с циклом for.	4	1	3	Тест
1.6	Погружение в условия. True, False, Break, Continue.	4	1	3	Беседа
1.7	Знакомство со списками.	4	1	3	Наблюдение
1.8	Решение задач по темам раздела 1.	4	-	4	Наблюдение
2	Базовые конструкции в Python.	46	12	34	
2.1	Дополнительные возможности цикла for. Кортежи. Сортировки.	4	2	2	Беседа
2.2	Знакомство со срезами и диапазонами. Равенство и совпадение объектов.	4	1	3	Наблюдение
2.3	Списочные выражения. Методы split и join.	4	1	3	Тест
2.4	Другие методы списков и строк.	4	2	2	Наблюдение
2.5	Функции.	8	2	6	Беседа
2.6	Области видимости переменных.	4	2	2	Опрос
2.7	Контрольная работа по темам разделов 1 и 2.	4	-	4	
2.8	Разбор контрольной работы.	2	-	2	Беседа
2.9	Введение в компьютерную графику. Модуль TKInter.	8	2	6	Беседа
2.10	Решение задач по темам раздела 2.	4	-	4	Наблюдение
3	Решение прикладных задач в Python.	68	15	53	
3.1	Функции. Функциональная парадигма программирования.	10	4	6	Беседа
3.2	Словари и множества.	8	4	4	Опрос

3.3	Обзор стандартной библиотеки Python.	4	1	3	Наблюдение
3.4	Дополнительные библиотеки Python.	10	4	6	Беседа
3.5	Промежуточный тест по разделу 3	2		2	Тест
3.6	Работа над итоговым проектом	26	2	24	Наблюдение
3.7	Подготовка к защите проектов	4	-	4	Беседа
3.8	Защита итоговых проектов	4	-	4	Защита проектов
	Итого	144	35	109	

3 Содержание программы

Раздел 1 Введение в программирование

Тема 1.1 Знакомство со средой PyCharm.

Теория. Понятия кода, интерпретатора, программы. Интегрированные среды, исполнение кода.

Практика. Простейшие программы с выводом на экран.

Тема 1.2 Условный оператор. Переменные и арифметика.

Теория. Алфавит и словарь языка Python. Объявление переменных. Типы данных. Арифметические операции. Операции с присваиванием. Линейные программы. Условный оператор.

Практика. Обзор типов данных. Вычисление математических выражений с помощью стандартных арифметических функций. Решение задач по теме.

Тема 1.3 Знакомство с циклом while.

Теория. Определение цикла. Оператор цикла с предусловием.

Практика. Решение задач по теме «Цикл while».

Тема 1.4 Отладчик.

Теория. Определение. Установка и запуск отладчика программного кода.

Практика. Решение задач.

Тема 1.5 Знакомство с циклом for.

Теория. Оператор цикла с постусловием. Оператор с известным числом повторов.

Практика. Решение задач по теме «Цикл for».

Форма подведения итогов: тест по теме «Циклы».

Тема 1.6 Погружение в условия. True, False, Break, Continue.

Теория. Принцип работы операторов True, False, Break, Continue.

Практика. Решение задач на тему «Погружение в условия».

Тема 1.7 Знакомство со списками.

Теория. Определение понятия «списки». Объявление списка. Действия над списками, над элементами списка.

Практика. Решение задач по теме «Списки»

Форма подведения итогов: самостоятельная работа по теме модуля.

Раздел 2 Базовые конструкции языка Python

Тема 2.1 Повторение. Дополнительные возможности цикла for. Кортежи. Сортировки.

Теория. Структура программы. Стандартные функции и арифметические выражения. Ветвления. Логические операции. Оператор цикла с известным количеством повторов, постусловием, предусловием. Оператор безусловного перехода.

Практика. Решение задач на тему «Цикл for».

Тема 2.2 Знакомство со срезами и диапазонами.

Теория. Понятие срезов и диапазонов. Равенство и совпадение объектов.

Практика. Решение задач по теме «Срезы и диапазоны»

Тема 2.3 Списочные выражения. Методы split и join.

Теория. Работа со списками. Применение методов split и join.

Практика. Решение задач на тему «Методы split и join».

Тема 2.4 Другие методы списков и строк.

Теория. Генераторы списков. Методы find и rfind. Метод replace и count.

Практика. Решение задач на методы списков и строк.

Тема 2.5. Знакомство с функциями.

Теория. Определение функции. Правила составления и вызова функций в программном коде. Локальные и глобальные переменные.

Практика. Решение задач на тему «Функции».

Тема 2.6 Области видимости переменных.

Теория. Локальная область видимости. Глобальная область видимости. Нелокальная область видимости.

Практика. Решение задач на тему «Области видимости переменных».

Тема 2.7 Введение в компьютерную графику. Модуль Tkinter.

Теория. Компьютерная графика и её виды. Кроссплатформенная библиотека для разработки графического интерфейса Tkinter.

Практика. Решение задач на тему «Модуль Tkinter».

Форма подведения итогов: контрольная работа по теме раздела.

Раздел 3 Решение прикладных задач в Python

Тема 3.1 Функции. Детальное изучение. Функциональная парадигма программирования.

Теория. Понятие функции. Именованные функции, инструкция `def`. Аргументы функции. Анонимные функции, инструкция `lambda`.

Практика. Решения задач на тему «Функции и функциональная парадигма программирования».

Тема 3.2 Словари и множества.

Теория. Словари (`dict`) и работа с ними. Методы словарей. Множества (`set` и `frozenset`).

Практика. Решение задач на тему «Словари и множества».

Тема 3.3 Обзор стандартной библиотеки Python.

Теория. Работа с модулями: создание, подключение инструкциями `import` и `from`. Модуль `os`, `sys`. Модуль `contextlib`. Модуль `abc`. Модуль `datetime`. Модуль `collections`.

Практика. Решение задач на тему «Стандартная библиотека Python»

Тема 3.4 Дополнительные библиотеки Python.

Теория. Модули `string`, `re`, `struct`, `CSV`, `smtplib`. Утилита `pip`.

Практика. Решение задач на тему «Дополнительные библиотеки Python».

Форма подведения итогов: тест по теме раздела.

Тема 3.5 Работа над итоговым проектом.

Защита итоговых проектов

4 Методическое обеспечение программы

Методы обучения:

- объяснительно-иллюстративный (беседы, объяснения, дискуссии);

- 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.

Педагогические технологии: проектная технология, здоровьесберегающие технологии, технология проблемного обучения. Проектная технология дает возможность самостоятельно конструировать свои знания, ориентироваться в информационном пространстве, развивается критическое и творческое мышление. Если проектная технология является спланированной и постоянной составляющей частью образовательного процесса, то будут созданы условия для формирования и развития внутренней мотивации учащихся к более качественному овладению знаниями, повышения мыслительной активности и приобретения навыков логического мышления. 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, воображения, познавательной мотивации, интеллектуальных эмоций.

5 Диагностика эффективности образовательного процесса

Входной контроль - имеет диагностические задачи и осуществляется в начале учебного года. Цель предварительной диагностики – зафиксировать начальный уровень подготовки обучающихся, имеющиеся знания, умения и навыки, связанные с предстоящей деятельностью. Входной контроль может

проводиться в следующих формах: творческие работы, самостоятельные работы, вопросники, тестирование и пр.

Промежуточная аттестация проводится на основании диагностики теоретических знаний и практических умений и навыков по итогам освоения модуля. Промежуточная аттестация проводится в следующих формах: защита кейсов, защита творческих или исследовательских работ и проектов, конференции, выставочный просмотр, смотр знаний и умений, викторины, олимпиада, конкурс, соревнование, турнир и пр.

Итоговая аттестация проводится по окончании обучения по программе.

Критерии оценки результатов аттестации обучающихся

Общими критериями оценки результативности обучения являются:

- оценка уровня теоретических знаний: широта кругозора,
- свобода восприятия теоретической информации, развитость практических навыков работы со специальной литературой, осмысленность и свобода использования специальной терминологии;
- оценка уровня практической подготовки учащихся: соответствие развития уровня практических умений и навыков программным требованиям, свобода владения специальным оборудованием и оснащением, качество выполнения практического задания, технологичность практической деятельности;
- оценка уровня развития и воспитанности обучающихся: культура организации самостоятельной деятельности, аккуратность и ответственность при работе, развитость специальных способностей, умение взаимодействовать с членами коллектива.

Возможные уровни теоретической подготовки обучающихся:

Высокий уровень – учащийся освоил практически весь объем знаний (80-100%), предусмотренных программой за конкретный период; специальные термины употребляет осознанно и в полном соответствии с их содержанием.

Средний уровень – у учащегося объем освоенных знаний составляет 50-79%; сочетает специальную терминологию с бытовой.

Низкий уровень – учащийся овладел менее чем 50% объема знаний, предусмотренных программой; учащийся, как правило, избегает употреблять специальные термины.

Возможные уровни практической подготовки обучающихся:

Высокий уровень – учащийся овладел 80-100% умениями и навыками, предусмотренными программой за конкретный период; работает с оборудованием самостоятельно, не испытывает особых трудностей; выполняет практические задания с элементами творчества.

Средний уровень – у учащегося объем усвоенных умений и навыков составляет 50-79%; работает с оборудованием с помощью педагога; в основном выполняет задания на основе образца.

Низкий уровень – учащийся овладел менее чем 50% умений и навыков, предусмотренных программой; испытывает затруднения при работе с оборудованием; обучающийся в состоянии выполнять лишь простейшие практические задания педагога.

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

В целях определения уровня усвоения программы учащимися осуществляются диагностические срезы:

- входная диагностика на основе анализа выбранной обучающимися роли в диагностической игре и степени их участия в реализации отдельных ее этапов, где выясняется начальный уровень знаний, умений и навыков учащихся, а так же выявляются их творческие способности.

- промежуточная диагностика позволяет выявить достигнутый на данном этапе уровень знаний, умений и навыков учащихся, в соответствии с реализованной проектной деятельностью. Предлагаются выполнение практических заданий, контрольные тесты.

- итоговая диагностика проводится в конце учебного курса (выставка и защита творческих проектов) и предполагает комплексную проверку

образовательных результатов по всем ключевым направлениям. Данный контроль позволяет проанализировать степень усвоения программы учащимися.

Результаты контроля фиксируются в диагностической карте.

Оценка уровней освоения модуля

Уровни	Параметры	Показатели
Высокий уровень (80-100%)	Теоретические знания.	Обучающийся освоил материал в полном объеме. Знает и понимает значение терминов, самостоятельно ориентируется в содержании материала по темам. учащийся заинтересован, проявляет устойчивое внимание к выполнению заданий.
	Практические умения и навыки.	Способен применять практические умения и навыки во время выполнения самостоятельных заданий. Правильно и по назначению применяет инструменты. Работу аккуратно доводит до конца. Может оценить результаты выполнения своего задания и дать оценку работы своего товарища.
	Конструкторские способности.(если нужны)	Учащийся способен узнать и выделить объект (конструкцию, устройство). Учащийся способен собрать объект из готовых частей или построить с помощью инструментов. Учащийся способен выделять составные части объекта. Учащийся способен видоизменить или преобразовать объект по заданным параметрам. Учащийся способен из преобразованного или видоизмененного объекта, или его отдельных частей собрать новый.
Средний уровень (50-79%)	Теоретические знания.	Учащийся освоил базовые знания, ориентируется в содержании материала по темам, иногда обращается за помощью к педагогу. Учащийся заинтересован, но не всегда проявляет устойчивое внимание к выполнению задания.
	Практические умения и навыки.	Владеет базовыми навыками и умениями, но не всегда может выполнить самостоятельное задание, затрудняется и просит помощи педагога. В работе допускает небрежность, делает ошибки, но может устранить их после наводящих вопросов или самостоятельно. Оценить результаты своей деятельности может с подсказкой педагога.
	Конструкторские	Учащийся может узнать и выделить объект (конструкцию,

	способности. (если нужны)	устройство). Учащийся не всегда способен самостоятельно разобрать, выделить составные части конструкции. Учащийся не способен видоизменить или преобразовать объект по заданным параметрам без подсказки педагога.
Низкий уровень (меньше 50%)	Теоретические знания.	Владеет минимальными знаниями, ориентируется в содержании материала по темам только с помощью педагога.
	Практические умения и навыки.	Владеет минимальными начальными навыками и умениями. Учащийся способен выполнять каждую операцию только с подсказкой педагога или товарищей. Не всегда правильно применяет необходимый инструмент или на использует вовсе. В работе допускает грубые ошибки, не может их найти их даже после указания. Не способен самостоятельно оценить результаты своей работы.
	Конструкторские способности. (если нужны)	Учащийся с подсказкой педагога может узнать и выделить объект (конструкцию, устройство). Учащийся с подсказкой педагога способен выделять составные части объекта. Разобрать, выделить составные части конструкции, видоизменить или преобразовать объект по заданным параметрам может только в совместной работе с педагогом.

Сводная таблица результатов обучения по модулю

№ п/п	ФИ обучающегося	Оценка теоретических знаний	Оценка практических умений и навыков	Конструкторские способности (или другие, по желанию)	Итоговая оценка
1.					
2.					
3.					
4.					
5.					
6.					
7.					

Учебно-методические средства обучения:

- специализированная литература по направлению, подборка журналов,
- наборы технической документации к применяемому оборудованию,

- образцы программ и систем, выполненные обучающимися и педагогом,
- плакаты, фото и видеоматериалы,
- учебно-методические пособия для педагога и обучающихся, включающие дидактический, информационный, справочный материалы на различных носителях, компьютерное и видео оборудование.

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

6 Материально-техническое обеспечение

Рабочее место учащегося:

- ноутбук с выходом в сеть Интернет.

Рабочее место наставника

- ноутбук с выходом в сеть Интернет;

- технические средства обучения (ТСО) (мультимедийное устройство).

Список рекомендуемой литературы

1. Бэрри П. Изучаем программирование на Python. — М., 2017. — 624 с.
2. Буйначев С. К. Основы программирования на языке Python: учебное пособие. — Екатеринбург: Изд-во Урал. ун-та, 2014. — 91 с.
3. Бхаргава А. Грокаем алгоритмы: иллюстрированное пособие для программистов и любопытствующих. — СПб.: Питер, 2017. — 288 с.
4. Гэддис Т. Начинаем программировать на Python / пер. с англ. 4-е изд. — СПб.: БХВ-Петербург, 2019. — 768 с.
5. Мюллер Дж. Python для чайников. — СПб. : Диалектика, 2019. — 416 с.
6. Луридас П. Алгоритмы для начинающих: теория и практика для разработчика. — М. : Эксмо, 2018. — 608 с.
7. Лутц М. Изучаем Python, пер. с англ. 3-е изд. — СПб.: Символ Плюс, 2009. — 848 с.
8. Рафгарден Т. Совершенный алгоритм. Жадные алгоритмы и динамическое программирование. — СПб.: Питер, 2020. — 256 с.
9. Рейтц К., Шлюссер Т. Автостопом по Python. — СПб. : Питер, 2017. — 336 с.
10. Фёдоров Д. Ю. Программирование на языке высокого уровня Python: учебное пособие для прикладного бакалавриата. — М. : Издательство Юрайт, 2019. — 161 с.