

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ ТВЕРСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
«ТВЕРСКОЙ ОБЛАСТНОЙ ЦЕНТР ЮНЫХ ТЕХНИКОВ»
СТРУКТУРНОЕ ПОДРАЗДЕЛЕНИЕ
«ДЕТСКИЙ ТЕХНОПАРК «КВАНТОРИУМ»**

СОГЛАСОВАНО
методическим советом
ГБУ ДО ТОЦЮТ
Протокол № 1

от «10» мая 2023 г.

УТВЕРЖДАЮ
Директор




А. А. Румянцев

«10» мая 2023 г.

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«Основы программирования на Python»**

Направленность: техническая

Общий объем программы: 288 часов

Возраст обучающихся: 12 - 17 лет

Срок реализации программы: 2 года

Уровень: 1 год – стартовый;

2 год – базовый

Автор: педагог дополнительного образования С. А. Рогонов

Рег. № 26-23

Тверь – 2023 г.

Информационная карта программы

Название	Дополнительная общеобразовательная общеразвивающая программа « Основы программирования на Python »
Направленность	техническая
Общий объем программы в часах	288 часов
Форма реализации	Очная
Целевая категория обучающихся	12-17 лет
Аннотация программы	Предлагаемая программа нацелена на развитие интереса обучающихся к программированию, в частности, обучающиеся знакомятся и получают навыки работы с языком программирования Python. Обучение по программе позволяет получить практические навыки и знания, выходящие за рамки школьных программ по информатике, математике.
Планируемые результаты реализации программы	<p>Обучающиеся получают:</p> <ul style="list-style-type: none"> – знания о языках программирования, их истории и современных тенденциях развития; – знания об основных принципах программирования; – навыки написания простейших программ; – навыки чтения программного кода и работы со средами программирования; – навыки работы с языком программирования Python, программирования в среде PyCharm, создания собственных программ и обеспечения их безотказной работы; – знания о программных средствах, основных принципах их функционирования; – навыки создания простейших приложений.

1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа **«Основы программирования на Python»** составлена в соответствии с требованиями:

- Федерального закона от 29.12.2012 г. №273-ФЗ «Об образовании в Российской Федерации»;
- приказа Министерства просвещения Российской Федерации РФ от 27.07.2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- письма Минобрнауки РФ от 11.12.2006 г. № 06-1844 «О Примерных требованиях к программам дополнительного образования детей»;
- письма Минобрнауки РФ от 18.11.2015 г. № 09-3242 «О направлении информации» вместе с методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы);
- приказа Министерства образования Тверской области от 23.09.2022 г. № 939/ПК «Об утверждении Регламента проведения независимой оценки качества дополнительных образовательных программ в Тверской области».

Направленность программы - техническая. Программа направлена на формирование алгоритмического мышления и овладение технологией обработки различных видов информации и основных приемов программирования. Обучение по данной программе создает благоприятные условия для интеллектуального и духовного воспитания личности обучающегося, его социально-культурного и профессионального самоопределения, развития познавательной активности и творческой самореализации.

Новизна программы обеспечивается тем, что дополнительная общеобразовательная общеразвивающая программа «Основы программирования на Python», реализуемая на базе «Детского технопарка «Кванториум», предоставляет возможность организовать образовательный процесс на основе установленных требований, сохраняя основные подходы и технологии в организации образовательного процесса. В то же время, преподаватель дополнительного образования может наполнять программу содержанием, в зависимости от имеющихся в Тверском регионе возможностей и тенденций развития экономики. Кроме того, новизна программы состоит в том, что она учитывает новые технологические уклады, которые требуют нового типа мышления и тесного взаимодействия с реальным сектором экономики.

Введение в дополнительное образование образовательной программы «Основы программирования на Python» с использованием таких методов, как командная работа, поиск проблем и их практическое решение, анализ и обобщение опыта, подготовка исследовательских проектов и их защита, элементы

соревнований, изменит картину восприятия обучающимися технических дисциплин, переводя их из разряда умозрительных в разряд прикладных.

Актуальность программы состоит в том, что она составлена с учётом современных потребностей рынка в специалистах в области информационных технологий. В настоящее время мы переживаем большие изменения в развитии общества. В современную жизнь человека всё больше внедряются компьютеры и информационные технологии. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём зачастую не на пользовательском уровне, а на уровне начинающего программиста.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке. Для обучения был выбран язык 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.

Научившись программировать на языке Python, обучающиеся получат мощный и удобный инструмент для решения как учебных, так и прикладных задач.

Вместе с тем чистота и ясность его конструкций позволит обучающимся с лёгкостью выучить любой другой язык программирования.

Знания и умения, приобретённые в результате освоения программы, 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.

В основу программы «Основы программирования на Python» заложен принцип практической направленности обучения.

Важным аспектом программы является самостоятельная работа над заданиями: обучающиеся учатся решать задачи без помощи преподавателя. Для этого в содержании программы фигурируют задания, в которых:

- для решения задачи необходимо найти какую-то информацию в сети Интернет;
- может потребоваться устранение ошибки, которую не так просто быстро обнаружить;
- условие сформулировано недостаточно прозрачно и ученику необходимо самостоятельно формализовать его (или задать правильные вопросы преподавателю).

Программа «Основы программирования на Python» предназначена для обучающихся разного уровня подготовки и с разной степенью мотивации.

Цель реализации программы: создание условий для изучения методов программирования на языке Python с целью использования их в учебной и

последующей профессиональной деятельности в различных предметных областях.

Задачи программы:

Обучающие:

- формировать и развивать навыки алгоритмического и логического мышления, грамотной разработки программ;
- познакомиться с принципами и методами функционального программирования;
- познакомиться с принципами и методами объектно-ориентированного программирования;
- приобрести навыки работы в интегрированной среде разработки на языке Python;
- изучить конструкции языка программирования Python;
- рассмотреть различные парадигмы программирования, предлагаемые языком Python (процедурная, функциональная, объектно-ориентированная);
- познакомиться с основными структурами данных и типовыми методами обработки этих структур;
- приобрести навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- приобрести навыки поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использование информации при решении задач;
- развить интерес к программированию.

Развивающие:

- способствовать формированию способностей решать проблемы и актуальные задачи в заданные сроки;
- развивать личностные компетенции, такие как память, внимание, способность логически мыслить и анализировать, концентрировать внимание на главном при работе над творческими и научными проектами в области информатики;
- расширять круг интересов, развивать самостоятельность, аккуратность, ответственность, активность, критическое и творческое мышление при работе в команде, при выполнении индивидуальных и групповых заданий;
- способствовать формированию основ технической культуры и грамотности;
- способствовать развитию творческих способностей обучающихся, познавательных интересов, развитию индивидуальности и самореализации;
- расширять технологические навыки при подготовке различных информационных материалов;
- развивать познавательные способности обучающихся, пространственное мышление, аккуратность и изобретательность при работе с техническими устройствами, при выполнении учебных проектов;

- способствовать формированию творческого подхода к поставленной задаче;
- сформировать самостоятельность и творческий подход к решению задач с использованием средств вычислительной техники;
- расширение кругозора обучающихся в области программирования для решения профессиональных задач (в том числе на основе системного подхода);
- способность грамотно строить коммуникацию, исходя из целей и ситуации;
- развивать навыки инженерного мышления, программирования, проектирования и эффективного использования электронного вычислительного оборудования.

Воспитательные:

- воспитывать дисциплинированность, ответственность, самоорганизацию;
- воспитывать трудолюбие, уважение к труду;
- воспитывать чувство патриотизма, гражданственности, гордости за достижения отечественной науки и техники;
- воспитывать мотивацию обучающихся к изобретательству, созданию собственных программных реализаций;
- воспитывать социально-значимые качества личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность.

Отличительной особенностью данной программы является то, что она реализуется в логике проектно-исследовательской деятельности обучающихся, с соблюдением всех базовых циклов проекта: от планирования деятельности до презентации и обсуждения её результатов. Проекты засчитываются как итоговые работы по курсу обучения. Они могут быть как индивидуальными, так и групповыми. Итоговые работы обязательно презентуются – это дает возможность обучающемуся увидеть значимость своей деятельности и получить оценку работы как со стороны сверстников, так и со стороны взрослых (педагогов, родителей и др.).

Также, отличительной особенностью является то, что данная программа:

- обеспечивает знакомство с фундаментальными понятиями алгоритмизации и программирования на доступном уровне;
- имеет практическую направленность с ориентацией на реальные потребности, соответствующие возрасту обучающегося;
- охватывает как алгоритмическое направление, так и вопросы практического использования полученных знаний при решении задач из различных областей знаний;
- допускает возможность варьирования в зависимости от уровня подготовки и интеллектуального уровня обучающихся (как группового, так и индивидуального);

– предусматривает возможность индивидуальной работы с обучающимися.

Практическая значимость программы заключается в том, что она способствует более успешному овладению знаниями и умениями по направлению «Программирование» через развитие самостоятельности обучающихся и оптимизацию средств и методов обучения.

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на Python» реализуется в течение двух лет.

В первый год обучения, при изучении программы 1 уровня, обучающиеся обязательно должны научиться работать с программным обеспечением (hard skills) и приобрести навыки, которые очень важны как для участия в коллективных проектах, так и жизни в социуме: работать совместно, брать на себя ответственность, выполнять определенную роль в командной работе, помогать и сочувствовать друг другу и т. д. (soft skills).

Во второй год обучения, на 2 уровне, продолжается углубленное изучение программного обеспечения, активное использование консультаций, увеличение индивидуальной работы, формирование проектных команд и работа над индивидуальными и групповыми проектами.

Адресат программы. Программа предназначена для обучающихся в возрасте с 12 до 17 лет, без ограничений возможностей здоровья, проявляющих интерес к программированию. Количество обучающихся в группе – 10-14 человек.

Форма обучения: очная

Уровень программы: 1 год – стартовый;
2 год – базовый.

Форма реализации образовательной программы: традиционная, с элементами дистанционных технологий

Организационная форма обучения: групповая, всем составом группы. Группа разновозрастная, постоянного состава.

Режим занятий: занятия с обучающимися каждого года обучения проводятся 2 раза в неделю по 2 академических часа. Продолжительность 1 академического часа – 45 минут.

При организации учебных занятий используются следующие **методы обучения:**

По внешним признакам деятельности педагога и обучающихся:

- *словесный* – беседа, лекция, обсуждение, рассказ, анализ;
- *наглядный* – показ, просмотр видеофильмов и презентаций;
- *практический* – самостоятельное выполнение заданий.

По степени активности познавательной деятельности обучающихся:

- *объяснительно-иллюстративные* – обучающиеся воспринимают и усваивают готовую информацию;
- *репродуктивный* – обучающиеся воспроизводят полученные знания и освоенные способы деятельности;
- *исследовательский* – овладение обучающимися методами научного познания, самостоятельной творческой работы.

По логичности подхода:

- *аналитический* – анализ этапов выполнения заданий.

По критерию степени самостоятельности и творчества в деятельности обучающихся:

- *частично-поисковый* – обучающиеся участвуют в коллективном поиске в процессе решения поставленных задач, выполнении заданий досуговой части программы;
- метод проблемного обучения;
- метод дизайн-мышления;
- метод проектной деятельности.

Возможные формы проведения занятий

Основной тип занятий – комбинированный, сочетающий в себе элементы теории и практики. Большинство заданий программы выполняется самостоятельно с помощью персонального компьютера и необходимых программных средств.

Программа состоит из 3-х разделов. Каждый раздел охватывает отдельную информационную технологию или её часть. Каждая тема программы начинается с постановки задачи – характеристики предметной области или конкретной программы на языке Python, которую предстоит изучить. С этой целью преподаватель проводит демонстрацию презентации или показывает саму программу, а также готовые работы, выполненные в ней. Закрепление знаний проводится с помощью практики, отработки умений самостоятельно решать поставленные задачи, соответствующих минимальному уровню планируемых результатов обучения.

Основные задания являются обязательными для выполнения всеми обучающимися. Задания выполняются на компьютере с использованием интегрированной среды разработки. При этом обучающиеся не только формируют новые теоретические и практические знания, но и приобретают новые технологические навыки.

Методика обучения ориентирована на индивидуальный подход. Для того, чтобы каждый обучающийся получил наилучший результат обучения, программой предусмотрены индивидуальные задания для самостоятельной работы на домашнем компьютере. Такая форма организации обучения стимулирует интерес обучающегося к программированию, активность и самостоятельность обучающихся, способствует объективному контролю глубины и широты знаний,

повышению качества усвоения материала обучающимися, позволяет педагогу получить объективную оценку выбранной им тактики и стратегии работы, методики индивидуального обучения и обучения в группе, выбора предметного содержания.

Для самостоятельной работы используются разные по уровню сложности задания, которые носят репродуктивный и творческий характер. Количество таких заданий в работе может варьироваться.

В ходе обучения проводится промежуточное тестирование по темам для определения уровня знаний обучающихся. Выполнение контрольных заданий способствует активизации учебно-познавательной деятельности и ведёт к закреплению знаний, а также служит индикатором успешности образовательного процесса.

Методы организации учебного процесса

При организации занятий по программе «Основы программирования на Python», для достижения поставленных целей и решения поставленных задач используются формы проведения занятий с активными методами обучения:

- занятие в форме проблемно-поисковой деятельности;
- занятие с использованием межпредметных связей;
- занятие в форме мозгового штурма;
- занятие в форме частично-поисковой деятельности.

Формы и методы контроля:

- тестирование;
- устный опрос;
- самостоятельные и контрольные работы;
- участие в проектной деятельности.

Общая характеристика учебного процесса:

- при изучении программы используются практические и самостоятельные работы;
- программа заканчивается написанием кода для решения одной из задач.

Ожидаемые результаты освоения программы «Основы программирования на Python»

Личностные результаты:

- формирование ответственного отношения к обучению, способности довести до конца начатое дело путём завершения творческого учебного проекта;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию;
- развитие опыта участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;

- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;

- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;

- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;

- формирование ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Метапредметные результаты:

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

- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая;

- умение критически оценивать правильность решения учебно-исследовательской задачи;

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

- владение основами самоконтроля, способность к принятию решений;

- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;

- формирование и развитие компетентности в области использования информационно-коммуникационных технологий (ИКТ-компетенция);

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

Предметные результаты:

- умение определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей, создавать на их основе несложные программы анализа данных, читать и понимать несложные программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня;

- формирование представлений об основных предметных понятиях («информация», «алгоритм», «исполнитель», «модель») и их свойствах;

- развитие логических способностей и алгоритмического мышления, умения составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами — линейной, условной и циклической;

- умение выполнять пошагово (с использованием компьютера или вручную) несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;

- навыки и опыт разработки программ в выбранной среде программирования, включая тестирование и отладку программ;
- умение использовать основные управляющие конструкции объектно-ориентированного программирования и библиотеки прикладных программ, выполнять созданные программы;
- умение 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;
- формирование умения соблюдать нормы информационной этики и права.

Компетентностный подход реализации программы позволяет осуществить формирование у обучающегося как личностных, так и профессионально-ориентированных компетенций через используемые формы и методы обучения, нацеленность на практические результаты.

В процессе обучения по программе у обучающегося формируются:

универсальные компетенции (SoftSkills):

- умение работать в команде: работа в общем ритме, эффективное распределение задач;
- наличие высокого познавательного интереса;
- умение ориентироваться в информационном пространстве, продуктивно использовать техническую литературу для поиска сложных решений;
- умение ставить вопросы, связанные с темой проекта, выбор наиболее эффективных решений задач в зависимости от конкретных условий;
- наличие критического мышления;
- проявление технического мышления, познавательной деятельности, творческой инициативы, самостоятельности;
- способность творчески решать технические задачи;
- готовность и способность применения теоретических знаний по физике, информатике для решения задач в реальном мире;
- способность правильно организовывать рабочее место и время для достижения поставленных целей;

предметные компетенции (HardSkills):

обучающиеся должны *знать*:

- основы современных языков программирования;
- порядок использования на практике как простых, так и сложных структур данных и конструкции для работы с ними;
- способы поиска и обработки ошибки в коде;
- способы разбиения решения задачи на подзадачи;
- как написать грамотный, красивый код;
- как анализировать свой или чужой код;
- как работать с информацией: находить, оценивать и использовать

- взаимосвязь между потребностями пользователей и свойствами проектируемых предметов и процессов;
 - способы планирования деятельности, разбиения задач на подзадачи, распределения ролей в рабочей группе;
- обучающиеся должны *уметь*:
- следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;
 - составлять план проекта, включая выбор темы, анализ предметной области, разбиение задачи на подзадачи;
 - использовать основные алгоритмические конструкции для решения задач;
 - применять полученные знания в практической деятельности;
 - подготовить отчет о проделанной работе; публично выступить с докладом;
- обучающиеся должны *владеть*:
- навыками пользования персональным компьютером и его периферийным оборудованием.

Мониторинг образовательных результатов

Система отслеживания, контроля и оценки результатов обучения по данной программе имеет три основных критерия:

- надежность знаний и умений предполагает усвоение терминологии, способов и типовых решений в сфере программирования и информационных технологий;
- сформированность личностных качеств определяется как совокупность ценностных ориентаций в сфере программирования и информационных технологий, отношения к выбранной деятельности, понимания ее значимости в обществе;
- готовность к продолжению обучения в сфере программирования и информационных технологий определяется как осознанный выбор более высокого уровня освоения выбранного вида деятельности, готовность к соревновательной и публичной деятельности.

Способы определения результативности реализации программы и формы подведения итогов реализации программы

В процессе обучения проводятся разные виды контроля результативности усвоения программного материала.

Текущий контроль проводится на занятиях в виде наблюдения за успехами каждого обучающегося, процессом формирования компетенций. Текущий контроль успеваемости носит безотметочный характер и служит для определения педагогических приемов и методов для индивидуального подхода к каждому обучающемуся, корректировки плана работы с группой.

Периодический контроль проводится по окончании изучения каждой темы в виде конкурсов или представления практических результатов выполнения

заданий. Конкретные проверочные задания промежуточной аттестации разрабатывает педагог с учетом возможности проведения промежуточного анализа процесса формирования компетенций. Периодический контроль проводится в виде педагогического анализа результатов анкетирования, тестирования, зачётов, опросов, выполнения учащимися диагностических заданий, участия обучающихся в мероприятиях (викторинах, соревнованиях). активности обучающихся на занятиях и т.п.

Итоговый контроль проводится в виде педагогического анализа результатов выполнения учащимися диагностических заданий, участия обучающихся в мероприятиях (викторинах, соревнованиях), защиты проектов, решения задач поискового характера. Итоги реализации программы подводятся в виде итоговой аттестации в форме защиты индивидуального или группового проекта в ходе публичного выступления с демонстрацией проектной работы. В процессе проведения итоговой аттестации оценивается результативность освоения программы.

Критерии оценивания приведены в таблицах 1,2,3.

Таблица 1

**Критерии оценивания сформированности компетенций
SoftSkills и HardSkills**

Уровень	Описание поведенческих проявлений
1 уровень - недостаточный	Обучающийся не владеет навыком, не понимает его важности, не пытается его применять и развивать.
2 уровень – развивающийся	Обучающийся находится в процессе освоения данного навыка. Обучающийся понимает важность освоения навыков, однако не всегда эффективно применяет его в практике.
3 уровень – опытный пользователь	Обучающийся полностью освоил данный навык. Обучающийся эффективно применяет навык во всех стандартных, типовых ситуациях.
4 уровень – продвинутый пользователь	Особо высокая степень развития навыка. Обучающийся способен применять навык в нестандартных ситуациях.
5 уровень – мастерство	Уровень развития навыка при котором обучающийся становится экспертом в среде сверстников, способен передавать остальным необходимые знания и навыки для освоения и развития данного навыка.

Критерии оценивания проекта

	Критерий	Баллы (от 0 до 3)
Оценка представленной работы: (тема)		
1.	Обоснование выбора темы. Соответствие содержания сформулированной теме, поставленным целям и задачам.	1 – не было обоснования темы, цель сформулирована нечетко, тема раскрыта не полностью 2 – был обоснован выбор темы, цель сформулирована нечетко, тема раскрыта не полностью 3 – было обоснование выбора темы, цель сформулирована в соответствии с темой, тема раскрыта полностью
2.	Рефлексия Владение рефлексией; социальное и прикладное значение полученных результатов (для чего? чему научились?), выводы	0 – нет выводов 1 – выводы по работе представлены неполно 2 – выводы полностью соответствуют теме и цели работы
Оценка выступления участников:		
3.	Качество публичного выступления, владение материалом	1 – участник читает текст 2 – участник допускает речевые и грамматические ошибки 3 – речь участника грамотная и безошибочная, хорошо владеет материалом
4.	Качество представления продукта проекта.	1 – участники представляют продукт 2 – оригинальность представления продукта 3 – оригинальность представления и качество выполнения продукта
5.	Умение вести дискуссию, корректно защищать свои идеи, эрудиция докладчика	1 – не умеет вести дискуссию, слабо владеет материалом 2 – участник испытывает затруднения в умении отвечать на вопросы комиссии и слушателей 3 – участник умеет вести дискуссию. Доказательно и корректно защищает свои идеи
6.	Дополнительные баллы	0-3

Критерии оценивания уровня освоения программы

Уровни освоения программы	Результат
Высокий уровень освоения программы	Обучающиеся демонстрируют высокую заинтересованность в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом занятии показывают отличное знание теоретического материала, практическое применение знаний воплощается в качественный продукт
Средний уровень освоения программы	Обучающиеся демонстрируют достаточную заинтересованность в учебной, познавательной и творческой деятельности. На итоговом занятии показывают хорошее знание теоретического материала, практическое применение знаний воплощается в продукт, требующий незначительной доработки
Низкий уровень освоения программы	Обучающиеся демонстрируют низкий уровень заинтересованности в учебной, познавательной и творческой деятельности. На итоговом занятии показывают недостаточное знание теоретического материала, практическая работа не соответствует требованиям

2. Содержание программы

2.1 Первый год обучения

2.1.1 УЧЕБНЫЙ ПЛАН

дополнительной общеобразовательной общеразвивающей программы «Основы программирования на Python. 1 уровень»

№ п/п	Наименование раздела, модуля, темы	Количество часов		
		Всего	Теория	Практика
1.	Введение в программирование Техника безопасности. Знакомство с компонентной базой	4	2	2
2.	Знакомство с языком Python	24	2	22
3.	Базовые конструкции в Python	32	10	22
4.	Решение прикладных задач в Python, в том числе, с использованием дистанционных технологий	68	18	50
5.	Практическая реализация проектной (кейсовой) задачи, в том числе, с использованием дистанционных технологий	10	0	10
6.	Публичная защита проектов и (или) участие в финальных этапах межрегиональных и всероссийских конкурсов	6	0	6
	ИТОГО	144	32	112

2.1.2 УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН
дополнительной общеобразовательной общеразвивающей программы
«Основы программирования на Python. 1 уровень»

№ п/п	Наименование модулей	Кол-во часов, всего	в том числе		Форма аттестации/ контроля
			теория	практика	
1.	Введение в программирование Техника безопасности. Знакомство с компонентной базой	4	2	2	
1.1	Техника безопасности. Требования, предъявляемые к обучающимся. Знакомство с компонентной базой и используемым оборудованием	1	1	0	Ответы обучающихся в процессе диалога
1.2	Обзор мирового инженерного опыта в сфере программирования	1	1	0	Ответы обучающихся в процессе диалога
1.3	Определение информационного поля для общения	1	0	1	Результаты регистрации
1.4	Знакомство с системой дистанционного обучения и регистрация в системе	1	0	1	
2.	Знакомство с языком Python	24	2	22	
2.1	Знакомство с языком Python. Понятия кода, интерпретатора, программы. Интегрированные среды, исполнение кода. Простейшие программы с выводом на экран	2	0	2	Выполнение заданий
2.2	Условный оператор. Переменные и арифметика	2	0	2	Выполнение заданий

2.3	Простые встроенные функции	2	0	2	Выполнение заданий
2.4	Знакомство с циклом while	2	0	2	Выполнение заданий
2.5	Отладчик	2	0	2	Выполнение заданий
2.6	Знакомство с циклом for	2	0	2	Выполнение заданий
2.7	Погружение в условия. True, False, Break, Continue	2	0	2	Выполнение заданий
2.8	Акселератор инженерных проектов. Идея.	4	2	2	Выполнение заданий
2.9	Вложенные циклы (в задачах)	2	0	2	Выполнение заданий
2.10	Резервное время на решение задач	2	0	2	Выполнение заданий
2.11	Цифровой ликбез, технологический диктант	2	0	2	Ответы обучающихся
3.	Базовые конструкции в Python	32	10	22	
3.1.	Резервное время на решение задач	2	2	0	Устный опрос
3.2	Множества	2	2	0	Устный опрос
3.3	Строки. Индексация	2	2	0	Устный опрос
3.4	Строки. Срезы	2	2	0	Устный опрос
3.5	Резервное время на решение задач.	2	2	0	Устный опрос
3.6	Знакомство со списками	2	0	2	Выполнение заданий
3.7	Кортежи. Преобразование коллекций	2	0	2	Выполнение заданий
3.8	Списочные выражения. Методы split и join	2	0	2	Выполнение заданий
3.9	Другие методы списков и строк. Сортировка	2	0	2	Выполнение заданий
3.10	Самостоятельная работа №1 на простые коллекции. Разбор самостоятельной	2	0	2	Выполнение заданий

	работы				
3.11	Вложенные списки	2	0	2	Выполнение заданий
3.12	Знакомство со словарями	2	0	2	Выполнение заданий
3.13.	Резервное время на решение задач. Подготовка к контрольной работе	2	0	2	Выполнение заданий
3.14	Контрольная работа по темам 1-го и 2-го модулей	2	0	2	Выполнение заданий
3.15	Анализ результатов контрольной работы. Резервное время на решение задач.	2	0	2	Выполнение заданий
3.16	Введение в компьютерную графику. Модуль TKInter	2	0	2	Выполнение заданий
4.	Решение прикладных задач в Python, в том числе, с использованием дистанционных технологий	68	18	50	
4.1.	Акселератор инженерных проектов. Модель.	2	1	1	Представление модели
4.2.	Функции. Функции и функциональная парадигма программирования	2	1	1	Решение задач и контрольных работ
4.3.	Функции. Возвращение значений из функций	2	1	1	Решение задач и контрольных работ
4.4.	Области видимости переменных	2	1	1	Решение задач и контрольных работ
4.5.	Функции: передача параметров	2	1	1	Решение задач и контрольных работ
4.6.	Решение задач.	2	0	2	Решение задач и контрольных работ
4.7.	Функции с переменным числом аргументов	2	0	2	Решение задач и контрольных работ
4.8.	Решение задач.	2	0	2	Решение задач и контрольных работ
4.9.	Функции как объект. Лямбда-функции	2	0	2	Решение задач и контрольных работ

4.10.	Решение задач. Рекурсия	2	0	2	Решение задач и контрольных работ
4.11.	Обработка коллекций. Поточковый ввод sys.stdin	2	0	2	Решение задач и контрольных работ
4.12.	Решение задач	2	0	2	Решение задач
4.13.	Самостоятельная работа №2 на функции. Разбор самостоятельной работы	2	0	2	Решение задач и контрольных работ
4.14.	Обзор стандартной библиотеки Python (random)	2	0	2	Решение задач и контрольных работ
4.15.	Дополнительные библиотеки Python	8	4	4	Решение задач и контрольных работ
4.16.	Решение задач на библиотеки	2	0	2	Решение задач и контрольных работ
4.17.	Самостоятельная работа №3 на библиотеки. Разбор самостоятельной работы	2	0	2	Решение задач и контрольных работ
4.18.	Библиотеки Python. NumPy. Резервное время на решение задач.	2	0	2	Решение задач и контрольных работ
4.19.	Введение в ООП	18	9	9	Решение задач и контрольных работ
4.20.	Решение задач. Подготовка к контрольной работе	4	0	4	Решение задач и контрольных работ
4.21.	Итоговая контрольная работа по темам модулей 1, 2 и 3	2	0	2	Решение задач и контрольных работ
4.22.	Цифровой ликбез, технологический диктант	2	0	2	Результаты работы с тренажерами
5.	Практическая реализация проектной (кейсовой) задачи, в том числе с использованием дистанционных технологий	10	0	10	Индивидуальные задания
6.	Публичная защита проектов и (или) участие в финальных этапах	6	0	6	

	межрегиональных и всероссийских конкурсов				
6.1	Разработка презентационных материалов. Подготовка слайдов и текста презентации для публичной защиты проекта	2	0	2	Результаты выполнения индивидуальных и групповых заданий
6.2	Участие в соревновательной деятельности, в т.ч., на Всероссийском уровне	2	0	2	Результаты соревнований
6.3	Акселератор инженерных проектов. Действующий прототип. Участие в публичной защите или презентации проекта	2	0	2	Результаты защиты
	Итого:	144	32	112	

2.1.3. СОДЕРЖАНИЕ ЗАНЯТИЙ

по дополнительной общеобразовательной общеразвивающей программе
«Основы программирования на Python. 1 уровень»

№ п/п	Наименование раздела, модуля, темы	Кол-во часов, всего	Содержание занятия
1	Введение в программирование Техника безопасности. Знакомство с компонентной базой	4	Обучающимся объясняется техника безопасности. Требования, предъявляемые к обучающимся при работе в аудитории на ноутбуках. Происходит знакомство с виртуальной средой взаимодействия: регистрация, организация личного кабинета, поиск и выкладывание материалов. Знакомство с системой автоматизированной проверки задач. Знакомство в системой дистанционного обучение и регистрация в системе
2.	Знакомство с языком Python	24	Основные понятия программирования: исполнитель, система команд, алгоритм, программа, среда разработки, интерпретатор, код программы и редактор кода. Понятие о языке высокого уровня Python. Структура программы, переменные и константы,

			<p>работа с числовыми переменными, арифметические операторы в Python. Основные управляющие конструкции алгоритмов с ветвлением в Python.</p> <p>Ввод-вывод в программе, условный оператор, оператор цикла с предусловием. Простейшие программы с использованием условного оператора if, оператора циклов while и операторов ввода-вывода. Технология разработки программы.</p> <p>Устройство циклов for. Основные управляющие конструкции циклического алгоритма в Python. Простейшие циклы и циклы с переменными.</p> <p>На этом этапе обучающиеся разрабатывают первые алгоритмы и программы, а также анализируют, на какие функциональные блоки может быть разбита программа, и определяют работоспособность разработанной программы. В течение модуля обучающиеся решают большое количество задач: от самых простых до сложных.</p>
3.	Базовые конструкции в Python	32	<p>Работа со списками, строками, множествами и кортежами в Python. Понятие итератора. Словари в Python. Вложенные списки. Сортировки. Взаимодействие между коллекциями. Решение задач. Списочные выражения.</p> <p>На этом этапе обучающиеся разрабатывают алгоритмы и программы на основе коллекция языка Python. В течение модуля обучающиеся решают большое количество задач: от самых простых до сложных.</p>
4.	Решение прикладных задач в Python, в том числе, с использованием дистанционных технологий	68	<p>Понятие ассоциативного массива. Модули в Python. Понятие подпрограммы, процедуры, функции. Функции в Python. Решение задач.</p> <p>Подключение и использование модулей стандартной библиотеки. Модульный принцип компоновки программы. Работа с документацией в стандартной библиотеке. Понятие репозитория различных пакетов Python. Работа с внешними библиотеками Python и утилитой pip. Решение задач.</p> <p>Самостоятельная работа: работа с конспектом, который описывает способы решения задач, разработка проекта по индивидуальному заданию, составление отчёта о выполнении</p>

			<p>индивидуальной или совместной работы.</p> <p>Тематика самостоятельных работ:</p> <ul style="list-style-type: none"> – задачи, в которых по заданному алгоритму необходимо написать программу; – задачи, в которых необходимо составить алгоритм решения и написать по нему программу; – задачи, для решения которых необходимо найти некоторую информацию в Интернете. <p>Зачётное занятие: выполнение итогового индивидуального задания по предложенной теме.</p>
5.	Практическая реализация проектной (кейсовой) задачи, в том числе с использованием дистанционных технологий	10	Самостоятельная работа обучающихся по доработке проекта
4.	Публичная защита проектов и (или) участие в финальных этапах межрегиональных и всероссийских конкурсов	4	Выполняется подготовка слайдов и текста презентации для публичной защиты проекта. Акселератор инженерных проектов. Представляется действующий проект.
	ИТОГО	144	

2.2 Второй год обучения

2.2.1 УЧЕБНЫЙ ПЛАН

дополнительной общеобразовательной общеразвивающей программы
«Основы программирования на Python. 2 уровень»

№ п/п	Наименование раздела, модуля, темы	Количество часов		
		Всего	Теория	Практика
1.	Техника безопасности. Знакомство с компонентной базой	2	0	2
2.	Решение прикладных задач в Python	10	0	10
3.	Базовые конструкции языка. Командная строка	10	4	6
4.	Библиотека PyQT. Пользовательский интерфейс	24	6	18
5.	Работа с репозиторием	6	1	5
6.	Библиотека PyGame. Игровые технологии	28	7	21
7.	Работа с API	34	13	21
8.	Практическая реализация проектной (кейсовой) задачи, в том числе, с использованием дистанционных технологий	24	0	24
9.	Публичная защита проектов и (или) участие в финальных этапах межрегиональных и всероссийских конкурсов	6	0	6
	ИТОГО	144	31	113

2.2.2 УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН
дополнительной общеобразовательной общеразвивающей программы
«Основы программирования на Python. 2 уровень»

№ п/п	Наименование модулей	Кол-во часов, всего	в том числе		Форма аттестации/ контроля
			теория	практика	
1.	Техника безопасности. Знакомство с компонентной базой	2	0	2	Ответы на вопросы
2.	Решение прикладных задач в Python	10	0	10	
2.1	Повторение. Решение задач на основные конструкции данных	2	0	2	Результаты решения задач
2.2	Повторение. Решение задач на классы	6	0	6	Результаты решения задач
2.3	Цифровой ликбез, технологический диктант	2	0	2	Ответы обучающихся
3.	Базовые конструкции языка. Командная строка	10	4	6	
3.1	Понятие исключения. Обработка исключений. Собственные исключения	2	1	1	Результаты решения задач
3.2	Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство файлов	2	1	1	Результаты решения задач
3.3	Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль schedule)	2	1	1	Результаты решения задач
3.4	Библиотека argparse. Задачи на создание скриптов с ее помощью	2	1	1	Результаты решения задач
3.5	Решение задач	2	0	2	Результаты решения задач
4.	Библиотека PyQT. Пользовательский интерфейс	24	6	18	
4.1	Что такое QT и PyQT. Знакомство	2	2	0	Устный опрос
4.2	QtDesigner, pyuis, два способа подключения uic-файла	2	1	1	Результаты решения задач
4.3	Диалоги, работа и с изображениями	2	1	1	Результаты решения задач

4.4	Решение задач	2	0	2	Результаты решения задач
4.5	Работа с простыми таблицами (csv). Фильтрация, сортировка, вывод. Работа с табличными данными в PyQT	2	1	1	Результаты решения задач
4.6	Введение в БД, работа с SQL-таблицами и отображение данных в PyQT	4	1	3	Результаты решения задач
4.7	Сборка независимого приложения. Работа над проектом	10	0	10	Результаты работы над проектом
5.	Работа с репозиторием	6	1	5	
5.1	Введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием	2	1	1	Результаты решения задач
5.2	Совместная работа над проектом, основные понятия и команды	2	0	2	Результаты работы над проектом
5.3	Решение задач	2	0	2	Результаты решения задач
6.	Библиотека PyGame. Игровые технологии	30	7	23	
6.1	Введение в библиотеку PyGame	2	1	1	Результаты решения задач
6.2	Игровой цикл. События	2	1	1	Результаты решения задач
6.3	Клетчатое поле. Классические игры на клетчатом поле	4	1	3	Результаты решения задач
6.4	Решение задач	2	0	2	Результаты решения задач
6.5	Изображения. Спрайты	2	1	1	Результаты решения задач
6.6	Столкновения и другие взаимодействия	2	1	1	Результаты решения задач
6.7	Игра в целом	2	1	1	Результаты решения задач
6.8	Украшения игры	2	1	1	Результаты решения задач
6.9	Работа над проектом	8	0	8	Результаты работы над проектом

6.10	Цифровой ликбез, технологический диктант	2	0	2	Результаты работы с тренажерами
6.11	Акселератор инженерных проектов. Модель	2	0	2	Представление модели проекта
7.	Работа с API	34	13	21	
7.1	Работа с популярными форматами файлов (json, xml)	2	1	1	Результаты решения задач
7.2	Знакомство с API	2	1	1	Результаты решения задач
7.3	Работа с протоколом HTTP	2	1	1	Результаты решения задач
7.4	Решение задач на API Яндекс.Карт.	2	0	2	Результаты решения задач
7.5	Введение во flask. Обработка HTML-форм.	2	1	1	Результаты решения задач
7.6	Шаблоны. flask-wtf	2	1	1	Результаты решения задач
7.7	Знакомство с flask-sqlalchemy	2	1	1	Устный опрос
7.8	Flask-sqlalchemy	2	1	1	Результаты решения задач
7.9	REST-API. Понятие. Делаем простое Rest-api	2	1	1	Результаты решения задач
7.10	REST-API. Flask-restfull	2	1	1	Результаты решения задач
7.11	Разворачиваем проект в облаке	2	1	1	Результаты работы над проектом
7.12	Решение задач	2	0	2	Результаты решения задач
7.13	Чат-боты	6	2	4	Результаты решения задач
7.14	Алиса	4	1	3	Результаты решения задач
8.	Практическая реализация проектной (кейсовой) задачи, в том числе с использованием дистанционных технологий	24	0	24	Индивидуальные и групповые задания
9.	Публичная защита проектов и (или) участие в финальных этапах межрегиональных и всероссийских конкурсов	6	0	6	

8.1	Разработка презентационных материалов. Подготовка слайдов и текста презентации для публичной защиты проекта	2	0	2	Результаты выполнения индивидуальных и групповых заданий
8.2	Участие в соревновательной деятельности, в т.ч., на Всероссийском уровне	2	0	2	Результаты соревнований
8.3	Акселератор инженерных проектов. Действующий прототип. Участие в публичной защите или презентации проекта	2	0	2	Результаты защиты
Итого		144	31	113	

2.2.3 СОДЕРЖАНИЕ ЗАНЯТИЙ

по дополнительной общеобразовательной общеразвивающей программе
«Основы программирования на Python. 2 уровень»

№ п/п	Наименование раздела, модуля, темы	Кол-во часов, всего	Содержание занятия
1.	Техника безопасности. Знакомство с компонентной базой	2	Обучающиеся проходят инструктаж по технике безопасности, вспоминают работу в системе дистанционного обучения
2.	Решение прикладных задач в Python	10	Повторение пройденного. Обучающиеся решают задачи на основные конструкции данных и задачи на классы. Проходит урок цифры, урок атома, технологический диктант.
3.	Базовые конструкции языка. Командная строка	10	Обучающиеся изучают понятие исключения, обработку исключений, собственные исключения, файлы в Python, типы файлов и работа с ними, внутреннее устройство файлов, работу с командной строкой (скрипты, аргументы), периодические задачи (модуль schedule), библиотеку argparse и выполняют задачи на создание скриптов с ее помощью. Решают задачи.
4.	Библиотека PyQt. Пользовательский интерфейс	24	Обучающиеся узнают что такое QT и PyQt. Происходит знакомство с QtDesigner, pyuis, два способа подключения uic-файла. Диалоги, работа и с изображениями.

			Решение задач. Работа с простыми таблицами (csv). Фильтрация, сортировка, вывод. Работа с табличными данными в PyQT. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Сборка независимого приложения. Работа над проектом.
5.	Работа с репозиторием	6	Выполняется введение в репозитории. Подключение в PyCharm. Работа с удаленным репозиторием. Совместная работа над проектом, основные понятия и команды. Решение задач.
6.	Библиотека PyGame. Игровые технологии	28	Обучающиеся изучают: 1) Введение в библиотеку PyGame. 2) Игровой цикл. События 3) Клетчатое поле. 4) Классические игры на клетчатом поле. 5) Выполняют решение задач. 6) Изображения. Спрайты 7) Столкновения и другие взаимодействия. 8) Создание игры в целом. 9) Украшения игры. 10) Выполняют работу над проектом.
7.	Работа с API	34	Выходе изучения работы с API обучающиеся изучают: Работу с популярными форматами файлов (json, xml). Знакомство с API. Работа с протоколом HTTP. Решение задач на API Яндекс.Карт. Введение во flask. Обработка HTML-форм. Шаблоны. flask-wtf. Знакомство с flask-sqlalchemy. Flask-sqlalchemy. REST-API. Понятие. Делаем простое Rest-api. REST-API. Flask-restfull. Разворачиваем проект в облаке. Решение задач. Чат-боты. Алиса.
8.	Практическая реализация проектной (кейсовой) задачи, в том числе, с использованием дистанционных технологий	24	Обучающимися выполняется практическая реализация проектной (кейсовой) задачи, в том числе, с использованием различных дистанционных технологий.

9.	Публичная защита проектов и (или) участие в финальных этапах межрегиональных и всероссийских конкурсов	6	Выполняется подготовка слайдов и текста презентации для публичной защиты проекта. Акселератор инженерных проектов. Представляется действующий прототип.
ИТОГО		144	

2.3. Календарный учебный график реализации программы

Год обучения	Название программы	Количество часов			Количество учебных		Даты начала и окончания	Продолжительность каникул
		все го	тео рия	прак тика	неде ль	дней		
1	Основы программирования на Python. 1 уровень	144	32	112	36	72	01.09.23 31.05.24	10 дней, январь
2	Основы программирования на Python. 2 уровень	144	31	113	36	72	01.09.23 31.05.24	10 дней, январь
Итого		288	63	225	72	144		20

3. Организационно-педагогические условия реализации дополнительной общеобразовательной общеразвивающей программы «Основы программирования на Python»

3.1. Материально-техническое обеспечение

Программа реализуется на базе детского технопарка «Кванториум».

Помещение - учебный кабинет, оформленный в соответствии с профилем проводимых занятий и оборудованный в соответствии с санитарными нормами.

Материальные ресурсы

№ п/п	Наименование	Количество (шт.)
1.	Презентационное оборудование	
1.1	Моноблочное интерактивное устройство TeachTouch 65". Интерактивный моноблочный дисплей, не менее 65 дюймов, со сверхвысоким разрешением	1
1.2	Напольная мобильная стойка для интерактивных досок с площадкой для крепления проекторов к стойке	1
2.	Компьютерное оборудование	
2.1	Ноутбук ASUS	14
2.2	Мышь A4Tech N-708X-1 Grey USB	14
2.3	МФУ (Копир, принтер, сканер), А4, ч/б, лазерный Kyocera M2640idw	1
3.	Программное обеспечение	
3.1	Офисное ПО Office Standart 2019 Open License	14
3.2	Антивирус KL4863RARDE: Kaspersky Endpoint Security Russian Edition. 100-149 Node 2 year Educational License	1

3.2 Информационное обеспечение Список рекомендуемой литературы Для педагога

1. Банкрашков, А.В. Программирование для детей на языке Python / А.В. Банкрашков. – М.: АСТ, 2018. – 288 с.
2. Бриггс, Джейсон Python для детей. Самоучитель по программированию / Джейсон Бриггс. – М.: Огни, 2013. – 177 с.
3. Бэрри, Пол Изучаем программирование на Python / Пол Бэрри. – М.: Эксмо, 2016. – 332 с.
4. Васильев, Александр Николаевич Python на примерах. Практический курс по программированию. Руководство / Васильев Александр Николаевич. – М.: Наука и техника, 2017. – 752 с.
5. Вордерман, К. Программирование на Python. Иллюстрированное руководство для детей / К. Вордерман, К. Стили, К. Квигли. – М.: Манн, Иванов и Фербер, 2017. — 346 с.
6. Гуриков, С.Р. Основы алгоритмизации и программирования на Python. Учебное пособие. Гриф МО РФ / С.Р. Гуриков. – М.: Инфра-М, Форум, 2018. – 707 с.
7. Златопольский, Д. М. Основы программирования на языке Python / Д.М. Златопольский. – М.: ДМК Пресс, 2017. – 277 с.
8. Лутц, М. Программирование на Python. Т. 1 / М. Лутц. – М.: Символ, 2016. – 992 с.
9. Лутц, М. Программирование на Python. Т. 2 / М. Лутц. - М.: Символ, 2016. - 992 с.
10. МакГрат, Майк Python. Программирование для начинающих / Майк МакГрат. – М.: Эксмо, 2013. – 727 с.
11. Мэтиз, Э. Изучаем PYTHON. Программирование игр, визуализация данных, веб-приложения / Э. Мэтиз. – СПб.: Питер, 2017. – 496 с.
12. Саммерфилд, М. Программирование на Python 3. Подробное руководство / М. Саммерфилд. – М.: Символ, 2016. – 608 с.
13. Федоров, Д. Ю. Программирование на языке высокого уровня Python: учебное пособие для прикладного бакалавриата / Д. Ю. Федоров. – 2-е изд., перераб. и доп. – М.: Издательство Юрайт, 2019. – 161 с.
14. Эрик, Мэтиз Изучаем Python. Программирование игр, визуализация данных, веб-приложения / Мэтиз Эрик. — М.: Питер, 2018. – 760 с.

Для обучающихся

1. Банкрашков, А.В. Программирование для детей на языке Python / А.В. Банкрашков. – М.: АСТ, 2018. – 288 с.
2. Бриггс, Джейсон Python для детей. Самоучитель по программированию / Джейсон Бриггс. – М.: Огни, 2013. – 177 с.
3. Бэрри, Пол Изучаем программирование на Python / Пол Бэрри. – М.: Эксмо, 2016. – 332 с.

4. Васильев, Александр Николаевич Python на примерах. Практический курс по программированию. Руководство / Васильев Александр Николаевич. – М.: Наука и техника, 2017. – 752 с.
5. Вордерман, К. Программирование на Python. Иллюстрированное руководство для детей / К. Вордерман, К. Стили, К. Квигли. – М.: Манн, Иванов и Фербер, 2017. – 346 с.
6. МакГрат, Майк Python. Программирование для начинающих / Майк МакГрат. – М.: Эксмо, 2013. – 727 с.
7. Мэтиз, Э. Изучаем PYTHON. Программирование игр, визуализация данных, веб-приложения / Э. Мэтиз. – СПб.: Питер, 2017. – 496 с.
8. Саммерфилд, М. Программирование на Python 3. Подробное руководство / М. Саммерфилд. – М.: Символ, 2016. – 608 с.

Для родителей

1. Банкрашков, А.В. Программирование для детей на языке Python / А.В. Банкрашков. – М.: АСТ, 2018. – 288 с.
2. Бриггс, Джейсон Python для детей. Самоучитель по программированию / Джейсон Бриггс. – М.: Огни, 2013. – 177 с.
3. Бэрри, Пол Изучаем программирование на Python / Пол Бэрри. – М.: Эксмо, 2016. – 332 с.
4. Васильев, Александр Николаевич Python на примерах. Практический курс по программированию. Руководство / Васильев Александр Николаевич. – М.: Наука и техника, 2017. – 752 с.
5. Вордерман, К. Программирование на Python. Иллюстрированное руководство для детей / К. Вордерман, К. Стили, К. Квигли. – М.: Манн, Иванов и Фербер, 2017. – 346 с.
6. МакГрат, Майк Python. Программирование для начинающих / Майк МакГрат. – М.: Эксмо, 2013. – 727 с.
7. Мэтиз, Э. Изучаем PYTHON. Программирование игр, визуализация данных, веб-приложения / Э. Мэтиз. – СПб.: Питер, 2017. – 496 с.
8. Саммерфилд, М. Программирование на Python 3. Подробное руководство / М. Саммерфилд. – М.: Символ, 2016. – 608 с.

Электронные образовательные ресурсы и интернет-ресурсы

1. Федеральный закон Российской Федерации от 29 декабря 2012 г. N 273-ФЗ «Об образовании в Российской Федерации» [Электронный ресурс]. – Режим доступа: <http://www.rg.ru/2012/12/30/obrazovanie-dok.html>
2. Лабораторные практикумы по программированию [Электронный ресурс] http://www.edu.holit.ua/index.php?option=com_content&view=category&layout=blog&id=72&Itemid=159(=ru
3. Основы программирования на языках C и C++ для начинающих. — Режим доступа: <http://cppstudio.com/>

4. Основы программирования на языке Python для начинающих. — Режим доступа: — Режим доступа: <https://pythonworld.ru/samouchitel-python>
5. Основы программирования на языке Python для начинающих. — Режим доступа: <https://itproger.com/>
6. Программирование на Python. — Режим доступа: <https://stepik.org>
7. Основы изучения HTML и CSS. — Режим доступа: <http://htmlbook.ru/>
8. Книги по изучению Python, Swift, JavaScript для начинающих. — Режим доступа: <https://bookflow.ru/knigi-poprogrammirovaniyu-dlya-detej/>
9. Свободно распространяемая программная система для изучения азов программирования дошкольниками и младшими школьниками. — Режим доступа: <https://piktomir.ru/>
10. CodeCombat — это платформа для учеников, чтобы изучать информатику во время игры. — Режим доступа: <https://codecombat.com/>

3.3 Использование дистанционных образовательных технологий при реализации программы

При реализации дополнительной общеобразовательной общеразвивающей программы «**Основы программирования на Python**» частично используются дистанционные технологии. Педагог вносит все методические материалы, используемые на каждом занятии, практические задания, задачи, учебный материал для самостоятельного изучения, ссылки на видео и иные Интернет-ресурсы на специальную платформу, созданную для каждой группы обучающихся по данной программе в «Системе дистанционного обучения Детский технопарк «Кванториум» Тверская область». Каждый обучающийся зарегистрирован в системе и имеет доступ к этим образовательным ресурсам. Загрузка материала осуществляется педагогом после проведения каждого занятия.

3.4 Кадровое обеспечение

Программу реализует педагог детского технопарка «Кванториум», имеющий среднее профессиональное или высшее образование по профилю педагогической деятельности, педагогическое образование и опыт работы с преподаваемой технологией и отвечающий квалификационным требованиям, указанным в профессиональном стандарте «Педагог дополнительного образования».

3.5 Методическое обеспечение

Особенности организации образовательной деятельности

Работа с обучающимися построена следующим образом: изложение теоретического материала, деление на команды, выполнение практических заданий, распределение ролей в команде и работа в команде, периодическая смена ролей и защита проделанной работы.

Практика показывает, что именно такая модель взаимодействия с обучающимися максимально эффективна, они учатся не только инженерно-

технической науке, но и работе в команде, умению слушать друг друга, советоваться и принимать решение сообща.

После основного теоретического курса организуется обучение в рамках мини-проектов и исследований, которое проводится как в индивидуальном формате, так и в группах с разной численностью участников. В целях специализации и погружения в данную программу обучающиеся разбиваются на проектные группы по 3-5 человек для выполнения, впоследствии, более узконаправленных проектов.

Методы образовательной деятельности

В период обучения применяются такие методы обучения и воспитания, которые позволят установить взаимосвязь деятельности педагога-наставника и обучающегося, направленную на решение образовательно-воспитательных задач.

По уровню активности используются методы:

- объяснительно-иллюстративный;
- эвристический метод;
- метод устного изложения, позволяющий в доступной форме донести до обучающихся сложный материал;
- метод проверки, оценки знаний и навыков, позволяющий оценить переданные педагогом материалы и, по необходимости, вовремя внести необходимые корректировки по усвоению знаний на практических занятиях;
- исследовательский метод обучения, дающий обучающимся возможность проявить себя, показать свои возможности, добиться определенных результатов.
- проблемного изложения материала, когда перед обучающимся ставится некая задача, позволяющая решить определенный этап процесса обучения и перейти на новую ступень обучения;
- закрепления и самостоятельной работы по усвоению знаний и навыков;
- диалоговый и дискуссионный.

Занятие состоит из теоретической (лекция, беседа) и практической части, создаются все необходимые условия для развития обучающихся.

Каждое занятие строится в зависимости от темы и конкретных задач, которые предусмотрены программой, с учетом возрастных особенностей обучающихся, их индивидуальной подготовленности.

Основные образовательные процессы: решение технических задач на базе современного оборудования, формирующих способы продуктивного взаимодействия с действительностью и разрешения проблемных ситуаций; познавательные квест-игры; технические соревнования и конкурсы.

Основные формы деятельности:

- познание и учение: освоение принципов функционирования сложного современного оборудования; освоение способов управления вниманием и возможностями организма;
- общение: принятие правил, ответственность как за собственные учебные достижения, так и за результаты в рамках «общего дела»;

- творчество: освоение подходов к разработке моделей управления как реальными, так и воображаемыми объектами, конструирование и программирование реалистических копий реальных и воображаемых объектов;
- игра: игра в команде, индивидуальные соревнования;
- труд: усвоение позитивных установок к труду и различным современным технологиям из области программирования.

Форма организации учебных занятий:

- беседа;
- лекция;
- решение задач;
- выполнение контрольной работы
- индивидуальная защита проектов.

Типы учебных занятий:

- первичного ознакомления с материалом;
- усвоение новых знаний;
- комбинированный;
- практические занятия;
- закрепление, повторение;
- итоговое.

Диагностика эффективности образовательного процесса осуществляется в течение всего срока реализации программы. Это помогает своевременно выявлять пробелы в знаниях, умениях обучающихся, планировать коррекционную работу, отслеживать динамику развития обучающихся. Для оценки эффективности образовательной программы выбраны следующие критерии, определяющие развитие интеллектуальных и технических способностей обучающихся: развитие памяти, воображения, образного, логического и технического мышления.

Результатом усвоения обучающимися программы являются: устойчивый интерес к занятиям программированием, результаты достижений в массовых мероприятиях различного уровня.

Учебно-методические средства обучения:

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

Педагогические технологии

В процессе обучения по программе используются разнообразные педагогические технологии:

- технологии развивающего обучения, направленные на общее целостное развитие личности, на основе активно-деятельного способа обучения, учитывающие закономерности развития и особенности индивидуума;
- технологии личностно-ориентированного обучения, направленные на развитие индивидуальных познавательных способностей каждого ребенка, максимальное выявление, раскрытие и использование его опыта;
- технологии дифференцированного обучения, обеспечивающие обучение каждого обучающегося на уровне его возможностей и способностей;

– технологии сотрудничества, реализующие демократизм, равенство, партнерство в отношениях педагога и обучающегося, совместно вырабатывают цели, содержание, дают оценки, находясь в состоянии сотрудничества, сотворчества.

– проектные технологии – достижение цели через детальную разработку проблемы, которая должна завершиться реальным, осязаемым практическим результатом, оформленным тем или иным образом;

– кейс-технологии, это интерактивные технологии, основанные на реальных или вымышленных ситуациях, направленные на формирование у обучающихся новых качеств и умений по решению проблемных ситуаций;

– компьютерные технологии, формирующие умение работать с информацией, исследовательские умения, коммуникативные способности.

В практике выступают различные комбинации этих технологий, их элементов.