

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ НИЖЕГОРОДСКОЙ ОБЛАСТИ
Государственное бюджетное профессиональное образовательное учреждение
«НИЖЕГОРОДСКИЙ РАДИОТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

СОГЛАСОВАНО:
Директор МАОУ № 186
«Авторская академическая школа»

М.В.Буров



УТВЕРЖДАЮ:
Директор ГБПОУ «НРТК»
И.А.Кормщикова
Приказ от 30.08.2023г. № 96-ИТ/О



СОГЛАСОВАНО:
Педагогический совет
Протокол № 1 от 30.08.2023г.

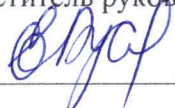
**ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ
ПРОГРАММА
«Основы программирования на Python»
(базовый уровень)**

Направленность – техническая
Возраст обучающихся: 11-17 лет
Срок реализации программы: 66 часов

г.Нижний Новгород
2023г.

Внутренняя экспертиза проведена. Программа рекомендована к рассмотрению на педагогическом совете учреждения.

Заместитель руководителя по ИМР

 /Русинова Е.Г.

«31» августа 2023 г.

Дополнительная общеразвивающая программа соответствует действующим федеральным, региональным нормативным документам Российской Федерации и локальным нормативным актам ГБПОУ «НРТК»

Информационная карта рабочей программы

Полное наименование программы	Дополнительная общеразвивающая программа «Основы программирования на Python» (базовый уровень)
Направленность	техническая
Общий объем программы	66 часов
Целевая категория обучающихся	обучающиеся 11–17 лет
Аннотация программы	Целью программы является формирование творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python. В ходе занятий обучающиеся приобретают знания и умения, которые могут быть использованы ими при дальнейшей сдаче экзаменов ОГЭ и частично 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.
Планируемые результаты реализации программы	Обучающиеся изучат основы, конструкции языка программирования Python, принципы и методы функционального программирования, объектно-ориентированного программирования, а также сформируют навыки алгоритмического и логического мышления, грамотной разработки программ в интегрированной среде разработки на языке Python.
Авторы-составители	- педагог дополнительного образования Варсеева Е.Н.; - методист Григорян Н.М.

1. Комплекс основных характеристик программы

1.1 Пояснительная записка

В современном мире все большую ценность приобретает навык не только грамотного использования готовых программных решений, но и умение создавать новые программы. Недостаточное количество часов, которое отводится на изучение программирования на уроках информатики в общеобразовательных школах, нередко представляет программирование лишь на элементарном уровне, как однообразный и зачастую непонятный процесс. Данная программа уделяет большое внимание практической работе, самостоятельному написанию программ, анализу программ, написанных другими разработчиками, что позволит обучающимся сформировать мотивацию к изучению программирования, сделает процесс обучения понятным и увлекательным.

Python – один из самых популярных скриптовых языков программирования, выбираемых для обучения программированию. Это объясняется простотой синтаксиса языка, интуитивной понятностью написанного кода, а также его практической востребованностью: обучающийся не просто изучает программирование, а знакомится с различными современными парадигмами программирования, которые он в будущем сможет применять в самых разных областях от разработки веб-приложений до машинного обучения.

Дополнительная общеразвивающая программа «Основы программирования на Python» (базовый уровень) реализуется посредством сетевого взаимодействия между ГБПОУ «Нижегородский радиотехнический колледж» и МАОУ № 186 «Авторская академическая школа».

Актуальность программы обосновывается тем, что синтаксис языка достаточно прост и интуитивно понятен, что позволяет сосредоточиться на логических и алгоритмических аспектах программирования. Кроме того, 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.

Прогностичность программы. Научившись программировать на языке Python, обучающиеся получат мощный и удобный инструмент для решения учебных и прикладных задач. Знания и умения, приобретенные в результате освоения курса, могут быть использованы учащимися 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 и изучения других языков.

Направленность программы. Программа обладает технической направленностью и предназначена для использования в системе дополнительного образования детей. Обучающиеся в ходе занятий приобщаются к инженерно-техническим знаниям в области информационных технологий, формируют логическое мышление, реализуют методы программирования и получают базисные знания в области информационных технологий.

Адресат программы. Дополнительная программа «Основы программирования на Python» (базовый уровень) предназначена для обучающихся в возрасте 11 – 17 лет, проявляющих интерес к программированию, без ограничений возможности здоровья. Содержание программы составлено с учётом возрастных и психологических особенностей детей данного возраста.

Формы проведения занятий. По форме организации деятельности предполагаются групповые или индивидуальные формы занятий, количество обучающихся в группе – 7-17 человек.

Место проведения занятий: г. Нижний Новгород, ул. Генкиной, д. 84.

Режим занятий. Занятия проводятся 2 раза в неделю по 2 академических часа с перерывом 10 минут для проветривания помещения и отдыха обучающихся. Продолжительность одного занятия – 40 минут.

Условия реализации программы. Зачисление детей производится без предварительного отбора (свободный набор). Принимаются учащиеся, обладающие начальным уровнем компьютерной грамотности. Возраст учащихся внутри одной группы может не совпадать.

Срок освоения общеразвивающей программы определяется содержанием программы и составляет 4 месяца (66 часов).

Формы обучения: очная, с применением дистанционных образовательных технологий (Закон №273-ФЗ, гл.2, ст.17, п.2.).

Виды занятий: беседы, обсуждения, игровые формы работы, практические занятия, метод проектов. Также программа курса включает групповые и индивидуальные формы работы обучающихся (в зависимости от темы занятия). По типу организации взаимодействия педагога с обучающимися при реализации программы используются личностно-ориентированные технологии, технологии сотрудничества.

Уровень освоения программы. По уровню освоения программа общеразвивающая, базовая. Это обеспечивает возможность обучения с любым уровнем подготовки.

1.2 Цель и задачи программы

Цель программы – формирование базовых знаний и практических навыков программирования на языке Python, а также развитие мотивации к дальнейшему изучению языка и его использования в учебной и последующей профессиональной деятельности в различных предметных областях.

Для успешной реализации поставленной цели необходимо решить ряд поставленных задач:

Обучающие (предметные):

- сформировать и развить навыки алгоритмического и логического мышления, грамотной разработки программ;
- ознакомить с принципами и методами функционального программирования;
- ознакомить с принципами и методами объектно-ориентированного программирования;
- сформировать навыки работы в интегрированной среде разработки на языке Python;
- изучить конструкции языка программирования Python;
- ознакомить с основными структурами данных и типовыми методами обработки этих структур;
- сформировать навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- сформировать навыки поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач.

Развивающие (метапредметные):

- развить умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- развить умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая;
- развить умение критически оценивать правильность решения учебно- исследовательской задачи;
- развить умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- сформировать владение основами самоконтроля, способность к принятию решений;
- развить умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- сформировать компетентность в области использования информационно-коммуникационных технологий (ИКТ-компетенция);
- развить умение организовывать учебное сотрудничество и совместную деятельность с педагогом и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Воспитательные (личностные):

- сформировать ответственное отношение к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам;
- сформировать способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой мотивации к обучению и познанию;
- развить опыт участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;
- сформировать коммуникативную компетенцию в общении и сотрудничестве со сверстниками в процессе образовательной, учебно- исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- сформировать целостное мировоззрение, соответствующее современному уровню развития информационных технологий;
- сформировать осознанное позитивное отношение к другому человеку, его мнению, результату его деятельности;
- сформировать ценность здорового и безопасного образа жизни;
- обеспечить усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

2. Содержание дополнительной общеразвивающей программы
2.1. Учебный план

№ п/п	Наименование раздела/модуля	Количество часов			Формы промежуточного контроля
		Всего	Теория	Практика	
1	Введение в программу	4	3	1	-
2	Модуль 1. Основные элементы в программировании на языке Python .	20	6	14	Тестирование
3	Модуль 2. Изучение условий, циклов, функций в программировании на языке Python.	28	10	18	Тестирование
4	Модуль 3. Проектная деятельность.	10	0	10	
5	Итоговая аттестация	4	0	4	Защита проектов
Итого:		66	19	47	

2.2 Содержание учебного плана

№ п/п	Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, проект	Количество часов			Основные виды деятельности обучающихся
			Всего	Теория	Практика	
1.	Вводное занятие.	Инструктаж по технике безопасности. Ознакомление со средами разработки и языками программирования.	2	2	0	Беседа, наблюдение за работой педагога, ответы на контрольные вопросы. <i>Оборудование:</i> Моноблочное интерактивное устройство.
2.	Ознакомление со средой PyCharm.	Теория	2	1	1	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Ознакомление со средой PyCharm, ее настройка, технологии программирования и разработки.				
		Практика				
		Реализация и тестирование первой базовой программы по шаблону в среде PyCharm.				
Модуль 1. Основные элементы в программировании на языке Python .						
3.	Базовые определения, элементы и синтаксис в Python.	Теория	6	1	5	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Определений и создание переменных string, a, собственных переменных, типов переменных int, float, double, bool, long и других, а также синтаксиса написания программ.				
		Практика				
		Решение простейших математических задач с заданием переменных, выбора их типа(int, float, double, bool, long) в зависимости от условий с вводом значений через консоль при помощи input в среде разработки PyCharm.				
4.	Списки, словари и множества в Python.	Теория	6	2	4	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Типы списков list, range, словарей dict и множеств set, frozenset в Python. Оценка полученных знаний с помощью тестирования.				
		Практика				
		Создание списков list и range в среде разработки PyCharm, использование словарей dict и решение задач с использованием множеств set и frozenset в Python.				

5.	Основные операторы в Python.	Теория	4	1	3	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Определения и обозначения основных операторов >, <, ==, !=, >=, <= и прочих. Подключение модулей import math, изучение работы с основными операторами на языке программирования Python.				
		Практика				
6.	Форматирование строк на языке Python.	Теория	2	1	1	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Форматирования строк с помощью команды format с использованием :=, {}, флагов '<', '>', '=', '^', '+', '-' и прочих инструментов форматирования строк.				
		Практика				
7.	Функции и их аргументы.	Теория	2	1	1	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Функций func, add, break и других, работа с консолью с помощью input и прочих методов работы с данными и элементами.				
		Практика				
8.	Условия на языке Python.	Теория	8	2	6	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Условия в Python, условный оператор if, if-elif-else, логика построения программ, создание блок-схем с условием.				
		Практика				
Модуль 2. Изучение условий, циклов, функций в программировании на языке Python.						
8.	Условия на языке Python.	Теория	8	2	6	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Условия в Python, условный оператор if, if-elif-else, логика построения программ, создание блок-схем с условием.				
		Практика				
8.	Условия на языке Python.	Теория	8	2	6	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Условия в Python, условный оператор if, if-elif-else, логика построения программ, создание блок-схем с условием.				
		Практика				
8.	Условия на языке Python.	Теория	8	2	6	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Условия в Python, условный оператор if, if-elif-else, логика построения программ, создание блок-схем с условием.				
		Практика				
8.	Условия на языке Python.	Теория	8	2	6	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Условия в Python, условный оператор if, if-elif-else, логика построения программ, создание блок-схем с условием.				
		Практика				

9.	Циклы и функции на языке Python.	Теория	10	4	6	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Циклы, цикл while, цикл do while, цикл for в Python, логика построения программ с функциями и циклами, построение блок-схемы для разработки программ.				
10	Матрицы, одномерные массивы и способы сортировки.	Практика	10	4	6	Наблюдение за работой педагога, самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Теория				
		Создание матрицы и массивы с помощью NumPy, функции arange() и shape. Способы преобразования массивов и матриц: извлечение, перенос и математические операции элементов. Способы сортировки Bubble Sort, Selection Sort, Insertion Sort данных в Python.				
		Практика				
		NumPy с одномерными массивами randint и матрицы A[], решение задач на операции с матрицами и массивами, а также их переменными с помощью пройденных ранее тем. Задачи на сортировку с применением Bubble Sort, Selection Sort, Insertion Sort.				
Модуль 3. Проектная деятельность.						
11.	Практикум: разработка собственной программы	Практика	10	0	10	Самостоятельная работа со средой PyCharm, консультирование с преподавателем по возникающим вопросам, ответы на контрольные вопросы преподавателя. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
		Выбор тематики творческой работы, разработка блок-схемы программы, создание программы в формате решения сложной математической задачи/игры/мини-приложения по выбранной тематике, оформление проекта.				
12.	Итоговая аттестация.	Защита индивидуальных или групповых проектов, подведение итогов курса.	4	0	4	Публичная защита собственного/группового итогового проекта в группе, рефлексия по итогам курса. <i>Оборудование:</i> Ноутбуки, мыши, моноблочное интерактивное устройство.
Итого			66	19	47	

2.3. Календарный учебный график

Номер учебной недели/ группа	сентябрь				октябрь				ноябрь				декабрь				
	04.09.2023-10.09.2023	11.09.2023-17.09.2023	18.09.2023-24.09.2023	25.09.2023-01.10.2023	02.10.2023-08.10.2023	09.10.2023-15.10.2023	16.10.2023-22.10.2023	23.10.2023-29.10.2023	30.10.2023-05.11.2023	06.11.2023-12.11.2023	13.11.2023-19.11.2023	20.11.2023-26.11.2023	27.11.2023-03.12.2023	04.12.2023-10.12.2023	11.12.2023-17.12.2023	18.12.2023-24.12.2023	25.12.2023-31.12.2023
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
ПП-1	У	У	У	У	У	П	У	У	У	У	У	У	П	У	У	У	И

Считать нерабочими праздничными днями: 4 ноября.

Условные обозначения:

У – учебные занятия,

П – промежуточный контроль,

И – итоговая аттестация.

2.4. Планируемые результаты

Обучающие (предметные):

- сформирован навык алгоритмического и логического мышления, грамотной разработки программ;
- обучающиеся ознакомлены с принципами и методами функционального программирования;
- обучающиеся ознакомлены с принципами и методами объектно-ориентированного программирования;
- сформирован навык работы в интегрированной среде разработки на языке Python;
- изучены конструкции языка программирования Python;
- обучающиеся ознакомлены с основными структурами данных и типовыми методами обработки этих структур;
- сформирован навык разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- сформирован навык поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач.

Развивающие (метапредметные):

- развито умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- развито умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая;
- развито умение критически оценивать правильность решения учебно- исследовательской задачи;
- развито умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- сформировано владение основами самоконтроля, способность к принятию решений;
- развито умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- сформирована компетентность в области использования информационно-коммуникационных технологий (ИКТ-компетенция);
- развито умение организовывать учебное сотрудничество и совместную деятельность с педагогом и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Воспитательные (личностные):

- сформировано ответственное отношение к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам;

- сформирована способность к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой мотивации к обучению и познанию;
- развит опыт участия в социально-значимых проектах, повышен уровень самооценки благодаря реализованным проектам;
- сформированы коммуникативные навыки, навык сотрудничества в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- сформировано целостное мировоззрение, соответствующее современному уровню развития информационных технологий;
- сформировано осознанное позитивное отношение к другому человеку, его мнению, результату его деятельности.

2.5. Определение результативности реализации программы Формы аттестации

Система отслеживания результатов обучающихся выстроена следующим образом:

- текущий контроль;
- промежуточный контроль;
- итоговый контроль.

Текущий контроль осуществляется путём наблюдения, определения качества выполнения практических работ, отслеживания динамики развития обучающегося в рамках учебных занятий. Способы проверки уровня освоения тем: опрос, выполнение упражнений, наблюдение, оценка выполненных самостоятельных работ.

Промежуточный контроль производится путем устных опросов с использованием визуальных и раздаточных материалов, а также с помощью тестирований, реализованных на различных онлайн платформах. После тестирований и устных опросов проводится обсуждение, рефлексия по вопросам и темам, которые вызвали затруднения, а так же повторение данных тем.

Итоговый контроль проводится в виде общей рефлексии, анализа результатов прохождения обучающимися программы и итоговой аттестации. Итоговая аттестация осуществляется в форме защиты проекта, демонстрации работы созданного приложения и оценивается по 100-балльной шкале, которая переводится в один из уровней освоения образовательной программы (Таблица 2).

Итоговый проект оценивается специально формируемой комиссией. Состав комиссии (не менее 3-х человек): педагог (в обязательном порядке), представитель администрации учебной организации, приветствуется привлечение IT-профессионалов, представителей высших и других учебных заведений.

Критерии оценивания итоговых работ разрабатываются педагогом в зависимости от содержания программы, ее целей и задач. Критерии оценивания итоговых работ обучающихся по программе «Основы программирования на Python» (базовый уровень) представлены в таблице 1:

Таблица 1

Критерии оценивания итогового проекта

Критерии оценивания проекта		
	Критерии	Баллы
Оценка представленной работы: (тема)		
1.	Обоснование выбора темы. Соответствие содержания сформулированной теме, поставленным целям и задачам.	0 – некорректно сформулирована тема, отсутствует обоснование выбора темы, цель не сформулирована, тема раскрыта поверхностно и кратко. 5 – нет обоснования темы, цель сформулирована не чётко, тема раскрыта не полностью. 10 – обоснование выбора темы есть, цель сформулирована нечетко, тема раскрыта не полностью. 15 – обоснование выбора темы есть, цель сформулирована в соответствие с темой, тема раскрыта полностью.
Оценка представленной работы: наполнение проекта		
2.	Разнообразие источников информации, целесообразность их использования.	0 – использована не соответствующая теме проекта информация. 5 – большая часть представленной информации не относится к теме проект.

		10 – проект содержит незначительный объем подходящей информации из ограниченного числа однотипных источников. 15 - проект содержит достаточно полную информацию из разнообразных источников.
3.	Анализ хода работы, определение выводов проекта.	0 – не предприняты попытки проанализировать ход и результат проекта. 5 – анализ заменен кратким описанием хода и порядка проекта. 10– представлен развернутый обзор проекта по достижению целей, заявленных в проекте. 15 - представлен исчерпывающий анализ ситуаций, складывавшихся в ходе работы над проектом, сделаны необходимые выводы, намечены перспективы проекта.
4.	Демонстрация программной части проекта, работоспособность программы проекта, типизация, общедоступность, оптимизированность кода программы.	0 – программа не представлена, не продемонстрирована работоспособность. 10 – программа представлена, не продемонстрирована работоспособность, код программы плохо типизирован, не ясен для стороннего специалиста, оптимизированность отсутствует. 15 – программа представлена, продемонстрирована частичная/некорректная работа программы, программа типизирована, понятна для стороннего специалиста, оптимизированность не полная. 25 – программа представлена, продемонстрирована полная, корректная работоспособность программы, программа типизирована, понятна для стороннего специалиста, оптимизированность полная.
Оценка выступления		
5.	Качество публичного выступления, владение материалом.	0 – участник с трудом читает текст. 5 – участник читает текст. 10 – участник допускает речевые и грамматические ошибки. 15 – речь участника грамотная и безошибочная, хорошо владеет материалом.
6.	Качество представления продукта	0 – участник представляет проект, не собственного исполнения. 5 – участник представляет типовой проект. 10 – участник улучшил/расширил типовой проект. 15 - участник создал собственный уникальный проект на основе полученных знаний и типовых шаблонов.

Таблица 2

Критерии оценивания уровня усвоения программы.

Уровни усвоения программы	Результат
Высокий уровень усвоения программы	Обучающийся демонстрирует высокую заинтересованность в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом тестировании показывают отличное знание теоритического материала, практическое применение знаний воплощается в качественный итоговый продукт.
Средний уровень усвоения программы	Обучающийся демонстрирует достаточную заинтересованность в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом тестировании показывают хорошее знание теоритического материала, практическое применение знаний воплощается в продукт, требующий незначительные доработки.

Низкий уровень усвоения программы	Обучающийся демонстрирует низкий уровень заинтересованности в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом тестировании показывают недостаточное знание теоритического материала, практическая работа не соответствует требованиям.
Баллы, набранные обучающимся	Уровень освоения
0-30 баллов	низкий
31-70 баллов	средний
71-100 баллов	высокий

**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 – пользователь	Обучающийся полностью освоил данный язык. 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 или ситуациях повышенной сложности.
5 уровень – мастерство	Уровень развития навыка, при котором обучающийся становится авторитетом и экспертом в среде сверстников. Обучающийся способен передавать остальным необходимые знания и навыки для освоения и развития данного навыка.

3. Комплекс организационно-педагогических условий реализации дополнительной общеразвивающей программы

3.1 Материально-техническое обеспечение

Требования к помещению:

- помещение для занятий, отвечающее требованиям СанПин для учреждений дополнительного образования;
- качественное освещение;
- столы, стулья по количеству обучающихся и 1 рабочее место для педагога.

Общее оборудование:

- ноутбук тип 1 – 1 шт.;
- ноутбук тип 2 – 13 шт.

Презентационное оборудование:

- моноблочное интерактивное устройство SMART MX265-V2 – 1 шт.;
- доска элементная магнитно-маркерная настенная с полимерным покрытием, 150 см x 120 см – 18 шт.;
- Web-камера A4Tech PK-940HA – 1 шт.;
- доска-флипчарт магнитно-маркерная 70 x 100 см BRAUBERG Стандарт – 1 шт.;
- напольная мобильная стойка для интерактивных досок DIGIS DSM-P1060CL – 1 шт.

Периферийное оборудование:

- МФУ струйный EPSON L14150, цветной (C11CH96404) – 1 шт.

Дополнительное оборудование:

- роутер TP-Link Archer C6U – 1 шт.;
- ультрафиолетовый рециркулятор Milerd DZR-1 – 1 шт.;
- тележка для хранения и зарядки 20 ноутбуков – 1 шт.;
- огнетушитель ОУ-3 (ВСЕ) – 1 шт.

Материалы:

- подставка под огнетушитель П-15 – 1 шт.;
- комплект переходников (Набор Rexant 18-1203 USB 6 переходников+удлинитель – 1 шт.;
- компьютерная мышь Logitech Mouse M100USB – 14 шт.;
- наушники SVEN AP-860V – 13 шт.

Программное обеспечение:

- среда разработки Python IDLE;
- среда разработки PyCharm;
- среда разработки VS Code;
- среда разработки Spider.

3.2 Формы и методы организации учебно-воспитательного процесса

Основной тип занятий — комбинированный, сочетающий в себе элементы теории и практики. Большинство заданий курса выполняется самостоятельно с помощью персонального компьютера и необходимых программных средств. Единицей учебного процесса является модуль, каждый из них охватывает отдельную информационную технологию или её часть. С учётом регулярного повторения ранее изученных тем темп изучения отдельных разделов блока определяется субъективными и объективными факторами. Каждая тема курса начинается с постановки задачи — характеристики предметной области или конкретной программы на языке Python, которую предстоит изучить. С этой целью педагог 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 в группе. Задания выполняются на компьютере с использованием интегрированной среды разработки. При этом обучающиеся не только формируют 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 предмету, активность и самостоятельность учащихся, 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 промежуточная аттестация по темам для определения уровня знаний учеников. Выполнение контрольных заданий способствует активизации учебно- познавательной деятельности и ведёт к закреплению знаний, а также служит индикатором успешности образовательного процесса. При организации занятий по программе «Основы программирования на языке Python» (базовый уровень), 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.

На занятиях используются следующие **средства обучения:**

Демонстрационный материал:

- тематическая подборка презентационного материала по темам;
- примеры работающих приложений, разработанных на языке Python.

Наглядные пособия:

- видеоматериалы по тематике разработки программного обеспечения;
- блок-схемы алгоритмов.

Реализация программы предполагает использование *здоровьесберегающих* технологий. Специальных медицинских противопоказаний к занятиям не существует, но при выборе данного объединения родителям и педагогу необходимо с особым вниманием отнестись к обучающимся, относящимся к группе риска по зрению, так как на занятиях значительное время они пользуются компьютером. Педагогами проводится предварительная беседа с родителями, в которой акцентируется внимание на обозначенную проблему.

Здоровьесберегающая деятельность реализуется:

- созданием безопасных материально-технических условий;
- включением в занятие динамических пауз, периодической смены деятельности обучающихся;
- контролем педагога за соблюдением обучающимися правил работы за персональным компьютером;
- созданием благоприятного психологического климата в учебной группе в целом.

Список использованной литературы и интернет-ресурсы

Основная литература

1. Бейдер Д., Чистый Python. Тонкости программирования для профи, 2-издание, 2022.
2. Задачи по программированию. Под ред. С.М. Окулова. М.: БИНОМ, Лаборатория знаний, 2006.
3. Информатика и ИКТ. Задачник – практикум в двух частях. Под ред. И.Г. Семакина и Е.К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.
4. Лутц М., Изучаем Python Том 1, 5-е издание, 2019.
5. Лутц М., Программирование на Python, Том I и II, 4-е издание, 2011.
6. Мэтиз Э., Изучаем Python. Программирование игр, визуализация данных, веб-приложения, 3-издание, 2022.
7. Лутц, М. Изучаем PYTHON. СПб.: Символ-Плюс, 2011.
8. Окулов С.М., Пестов А.А. 100 задач по информатике. Киров: Изд-во ВГПУ, 2000 — 272с.
9. Окулов, С Основы программирования. М.: БИНОМ. Лаборатория знаний, 2012.
10. Поляков К.Ю., Е.А. Еремин. Информатика. Углубленный уровень. Учебник для 10 класса в двух частях. М.: БИНОМ. Лаборатория знаний, 2014.

Интернет-ресурсы

1. «Python 3 для начинающих. [Электронный ресурс]: <https://pythonworld.ru>
2. «Питонтьютор». [Электронный ресурс]: <https://pythontutor.ru>
3. Материалы и презентации к урокам в LMS Яндекс. Лицея.
4. Питон, Workbook. Задачник 21. [Электронный ресурс]: <https://smartiqa.ru/python-workbook>