

Республика Карелия
Администрация Прионежского муниципального района
МОУ «НОВОВИЛГОВСКАЯ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА №3»

«Рассмотрено на заседании»: педагогического совета школы протокол № № 2 от 31.08.2026 МОУ «СОШ №3 п. Новая Вилга»	«Утверждаю»: директор школы МОУ «СОШ №3, п. Новая Вилга» Мороз Татьяна Викторовна Подписано цифровой подписью: Мороз Татьяна Викторовна Дата: 2026.09.02 11:03:51 +03'00' / Т.В. Мороз/ Приказ № 147-о от 1.09.26г.
--	---

Рабочая программа
дополнительной общеобразовательной, общеразвивающей программы
технической направленности
«Программирование на языке Python»

Возраст детей – 11-16 лет
Срок реализации -1 год

Составитель программы:
Гаспорович О.Е.,
учитель информатики

п. Новая Вилга
2026 год

Пояснительная записка.

Нормативно-правовыми основаниями разработки дополнительных общеобразовательных общеразвивающих программ являются следующие документы:

- Федеральный закон № 273-ФЗ;
- СанПиН 2.4.4.3172–14 « Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей», утв. постановлением Главного государственного санитарного врача РФ от 04.07.2014 № 41;
- Концепция развития дополнительного образования детей;
- Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утв. приказом Минобрнауки России от 29.08.2013 № 1008 (далее — Порядок организации и осуществления образовательной деятельности по ДОП).

Комплекс основных характеристик дополнительной общеразвивающей программы:

1. Общая характеристика программы:

Направленность (профиль) программы — научно-техническая.

Актуальность программы обусловлена широким распространением информационно-коммуникационных технологий в обществе и необходимостью обеспечивать связанную с этим инфраструктуру специалистами. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём на уровне начинающего программиста.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке программирования. Для обучения в рамках программы выбран язык Python, который является достаточно эффективным и доступным инструментом достижения задач в области создания программных продуктов.

Синтаксис языка достаточно прост и интуитивно понятен, что понижает порог вхождения и позволяет сосредоточиться на алгоритмических аспектах программирования.

Новизна программы заключается в том, что Python дает более широкие возможности в области программирования, чем Pascal, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Этот язык быстрее и легче усваивается, чем Pascal. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

Адресат программы — ученики 5-11 классов;

Срок освоения программы — 1 год;

Объем программы - 34 часа;

Режим занятий — 1 час в неделю по 40 мин;

Формы обучения и виды - групповые занятия, индивидуальные занятия, демонстрация-объяснение, практические занятия.

В связи с возникающими непредвиденными (форс-мажорными) обстоятельствами в течение учебного года, обучение по данной программе возможно с применением дистанционных образовательных технологий. Для организации ДО можно использовать <http://moodle.nvschool3.ru/>, Zoom или другие онлайн- ресурсы.

2. Цель программы: способствовать формированию 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.

Задачи:

Образовательные (предметные):

- Познакомить с основными алгоритмическими конструкциями и правилами их записи, с основными способами организации данных.
- Научить составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций.
- Научить распознавать необходимость применения той или иной алгоритмической конструкции при решении задач.
- Научить организовывать данные для эффективной алгоритмической обработки.
- Научить разрабатывать алгоритмы и реализовывать их на языке программирования Python.
- Научить осуществлять отладку и тестирование программы.

Личностные:

- повышение общекультурного уровня;
- вооружение правильным методологическим подходом к познавательной и практической деятельности;
- выделение и раскрытие роли информационных технологий и компьютеров в развитии современного общества;
- привитие навыков сознательного и рационального использования компьютера в своей учебной, а затем и профессиональной деятельности;
- воспитание стремления к овладению техникой исследования;
- воспитание трудолюбия, инициативности и настойчивости в преодолении трудностей.

Метапредметные:

- владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить;
- планирование – определение последовательности промежуточных целей с учетом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели при помощи фиксированного набора средств;
- прогнозирование – предвосхищение результата.

Содержание учебного предмета

1. **Введение в программирование.** Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.
2. **Переменные и выражения**
 - а. Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.
 - б. Выражения. Операции. Порядок выполнения операций. Математические функции. Композиция.

- с. Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.
3. **Условный оператор.** Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.
 4. **Циклы.** Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.
Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.
 5. **Строки - последовательности символов.** Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки. Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.
 6. **Списки.** Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков. Срезы списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range.

Тематическое планирование

№	Тема	Количество часов
1.	Введение в программирование. Установка среды.	1
2.	Арифметические выражения, типы данных.	3
3.	Переменные.	3
4.	Типы данных.	3
5.	Ввод переменных.	3
6.	Целочисленная арифметика.	3
7.	Условный оператор if. Синтаксис условной инструкции.	3
8.	Цикл for.	3
9.	Цикл while.	3
10.	Строки.	3
11.	Списки.	3
12.	Операции со списками. Генераторы.	3
		34 ч.