

**Министерство образования и науки Смоленской области
муниципальное бюджетное общеобразовательное учреждение
«Средняя школа №10»**

Принято на заседании методического
(педагогического) совета

от «__» __ 08 2025 г. Протокол № 1

Утверждаю

Директор МБОУ «Средняя школа №10»

_____/Л.Г.Коган/

от «__» _____ 2025 г.



**Дополнительная общеобразовательная общеразвивающая программа
технической направленности
«Основы программирования.
Python»**

Возраст обучающихся: 11-15 лет

Срок реализации: 1 год

Составитель:

Антонова Наталья Сергеевна

педагог дополнительного образования

г. Рославль

2025 год

Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования. Python» разработана в соответствии с нормативно–правовыми документами:

- Федеральным законом «Об образовании в Российской Федерации» от 29 декабря 2012 г. № 273-ФЗ;
- «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (Приказ от 27 июля 2022 г. № 629);
- СанПиН 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодёжи» (Постановление Главного государственного санитарного врача РФ от 28.09. 2020г. № 28);
- Концепцией развития дополнительного образования детей до 2030 года (Распоряжение правительства РФ от 31 марта 2022 г. № 678-р);
- Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (Письмо Минобрнауки РФ «О направлении информации» от 18 ноября 2015 г. N 09- 3242);
- Уставом МБОУ «Средняя школа №10»;
- Программой воспитания МБОУ «Средняя школа №10»;
- Социальным заказом родителей (законных представителей)

Направленность – техническая.

Уровень освоения программы - общекультурный.

Программа представляет особый интерес в связи с широким распространением цифровой техники в обществе, в связи с возрастающей потребностью обучающихся в освоении цифровых технологий и повышением их интереса к современным языкам программирования.

Содержание программы предусматривает ознакомление воспитанников с основами работы в интегрированной среде программирования и основными возможностями языка программирования Python. Программа позволяет освоить практически все операторные конструкции и познакомиться с основными функционального и объектного программирования.

Актуальность программы заключается в том, что в настоящее время владение компьютерными технологиями рассматривается как важнейший компонент образования, играющий значимую роль в решении приоритетных задач образования – в формировании целостного мировоззрения, системно-информационной картины мира, учебных и коммуникативных навыков.

Воспитанники, прошедшие обучение по данной программе, получают знания, умения и навыки, необходимые для сознательного выбора в будущем профессии, связанной с программированием. Программа позволяет школьникам познакомиться и понять тонкости профессии программист и

профессий связанных с разработкой IT-приложений, оценить себя в этих профессиях, выработать профессионально важные качества.

Педагогическая целесообразность этой программы заключается в том, что она является целостной и непрерывной в течении всего процесса обучения, и позволяет школьнику шаг за шагом раскрывать в себе творческие возможности и само реализовать в современном мире. В процессе программирования дети получают дополнительное образование в области алгебры, геометрии и информатики.

Отличительные особенности программы.

Особенностью программы является её направленность на формировании у воспитанников потребности в инженерном образовании научном исследовании. Инновационной является авторская методика, позволяющая сочетать изучение достаточно сложного языка программирования с исследовательской деятельностью и разработкой проектов самых разнообразных по содержанию, позволяющая ребенку в конце курса обучения почувствовать себя настоящим профессионалом.

Адресат программы. Программа рассчитана для детей от 11 до 15 лет. Программа может корректироваться в процессе работы с учетом возможностей материально-технической базы, возрастных особенностей обучающихся, их способностей усваивать материал.

Обучающиеся, поступающие в объединение, проходят собеседование, направленное на выявление их индивидуальности и склонности к выбранной деятельности. Занятия проводятся в группах, подгруппах и индивидуально, сочетая принцип группового обучения с индивидуальным подходом. Условия набора детей в коллектив: принимаются все желающие.

Объем и сроки освоения программы. Программа рассчитана на 1 год обучения - 1 раз в неделю, итого 36 часов.

Занятия проводятся в компьютерном классе, оснащенном индивидуальными стационарными компьютерами и оборудованием для презентаций. В каждом занятии прослеживаются три части: теоретическая, практическая и исследовательская.

Учебные часы условно делятся на теорию и практику. Однако, теория включает в себя не только разбор конструкция языка программирования, алгоритмических приёмов, технологии решения задач, но и практические задания, выполнение которых непосредственно не связано с работой на персональном компьютере. На теории организуется обсуждение с учащимися

сложных вопросов, проектных решений, применение тех или иных методов в различных профессиях.

Практические занятия проводятся непосредственно на персональном компьютере, и предполагают работу в интегрированной среде программирования Python с 10 минутным перерывом, во время которого учащиеся выполняют упражнения для снятия зрительного напряжения и снятия напряжения с костно-мышечного аппарата.

Исследовательская часть предполагает всестороннее изучение возможностей отдельных конструкций языка программирования Python и возможность их применения при решении нестандартных задач прикладного характера, разработку оптимальных нестандартных алгоритмов, создание компактного программного кода. Обучающиеся могут предложить собственный алгоритм решения задачи и протестировать работу соответствующей программы.

Формы обучения:

- теоретическая форма, в которой преподаватель объясняет новый материал и консультирует обучающихся в процессе выполнения ими практических заданий на компьютере;
- практическая форма, в которой обучающиеся после занятий самостоятельно выполняют на компьютере практические задания.

Формы организации образовательного процесса: лекция, беседа, демонстрация, практические занятия, творческая работа; проектная деятельность.

Режим занятий

Занятия проходят 1 раз в неделю.

Цель и задачи программы

Цель: Создание условий для достижения обучающимися результатов развития в личностном, предметном, метапредметном направлениях, обеспечивающих их социальную адаптацию в области программирования, ИКТ на профессиональном уровне.

Развить личность обучающегося, способного к творческому самовыражению через овладение основами программирования на одном из самых востребованных языков программирования Python.

Понять значение алгоритмизации как метода познания окружающего мира, принципы структурной алгоритмизации; научиться разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

Задачи:

Обучающие:

- Познакомить с возможностями и особенностями современного языка программирования Python;
- Сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- Изучить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами); научить применять функции при написании программ на языке программирования Python;
- Научить отлаживать и тестировать программы, делать выводы о работе этих программ.
- Сформировать систему знаний, умений и навыков, необходимых в работе программиста;
- Обучить технологии работы со справочниками по языку программирования и поиску нужной информации на сайтах разработчиков программного обеспечения;
- Обучить приёмам предъявления результатов проделанной работы на конференциях и, конкурсах;
- Обучить коммуникативным навыкам при совместной работе над проектом.

Развивающие:

- Развить познавательные потребности и способности школьников
- Развивать творческие способности, алгоритмическое и логическое мышление;
- Развивать представления о возможностях применения современных компьютерных технологий в профессиональной деятельности;
- Развивать внимательность и наблюдательность, прививать навыки аккуратности и точности в работе;
- Повысить интеллектуальный уровень и расширить интеллектуальные навыки;
- Создать мотивацию к постоянному самообразованию.

Воспитательные:

- Воспитывать навыки взаимодействия при командной работе над проектом;
- Воспитывать чувство ответственности за результаты деятельности;
- Способствовать формированию культуры программирования;
- Способствовать формированию у воспитанников культуры создания и этики представления проектов на конференциях и в Интернете;

- Сформировать осознанное отношение к выбору будущей профессии.

Планируемые результаты

Предметные:

- изучить принципы структурного, функционального и объектного программирования на примере языка программирования Python;
- научиться формулировать и анализировать алгоритмы, составлять и отлаживать программы.

Личностные результаты:

- сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- гармоничное развитие творческих способностей и логического мышления учащихся;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов.

Метапредметные результаты:

- владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем;
- способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№ п/п	Название образовательных блоков, разделов	Коли честв о часов	Формы аттестации/ контроля
1	Основы охраны труда. Организация рабочего места.	2	Беседы, наблюдения, записи в тетради, проверка отчётов
2	Знакомство с языком Python.	2	Беседы, наблюдения, записи в тетради, проверка отчётов
3	Установка программы Python.	2	Беседы, наблюдения, записи в тетради, проверка отчётов

4	Переменные и выражения. Организация ввода и вывода данных. Операции.	6	Беседы, наблюдения, записи в тетради, проверка отчётов
5	Программирование ветвящихся алгоритмов.	6	Беседы, наблюдения, записи в тетради, проверка отчётов
6	Циклы	8	Беседы, наблюдения, записи в тетради, проверка отчётов
7	Стиль программирования и отладка программ	2	Беседы, наблюдения, записи в тетради, проверка отчётов
8	Работа над проектами	8	Лекции, беседы, проверка отчётов
итого		36	

Содержание учебного плана

1. Вводное занятие. Знакомство с языком Python.

Теория. История создания языка, его плюсы и минусы. Сферы использования и перспективы.

Понятия кода, интерпретатора, программы. Простейшие программы с выводом на экран.

Практика. Решение примеров и вывод на экран.

Промежуточный контроль беседа

2. Переменные и выражения. Организация ввода и вывода данных. Операции.

Теория. Переменные и арифметика.

Практика. Решение уравнений с применением математических операторов

Промежуточный контроль практическая работа

3. Программирование ветвящихся алгоритмов.

Теория. Понятие о языке высокого уровня Python. Основные управляющие конструкции

алгоритмов с ветвлением в Python.

Практика. Решение задач с условиями, построение алгоритмов решения условных задач.

Промежуточный контроль практическая работа

4. Циклы

Теория. Устройство циклов for и while. Основные управляющие конструкции циклического

алгоритма в Python. Алгоритм Евклида.

Практика. Решение циклических задач. Совмещение конструкции цикла с условной

конструкцией для решения задач с циклами и условиями.

Промежуточный контроль практическая работа

5. Стил ь программирования и отладка программ

Теория. Ввод в программу. Типы данных, примеры работы с типами данных.

Метод тар.

Практика. Укрепление навыка работы с переменными на примерах.

Выполнение практических заданий

Промежуточный контроль практическая работа

6. Работа над проектами

Теория. Понятие «проект». Цели, задачи. Визуализация проектной идеи (схемы, блок-схемы). Планирование проекта. Реализация.

Практика. Разработка проекта.

Промежуточный контроль Защита проекта.

Календарно-учебный график

№ п/п	месяц	Тема занятия	Кол- во часо в	Форма занятия	Форма контроля
1		Инструктаж по технике безопасности. Организация рабочего места.	2	Комбинированное занятие	Беседа, наблюдение
2		Знакомство с языком Python.	2	Комбинированное занятие	Беседа, наблюдение
3		Установка программы Python.	2	Практическое занятие	Беседа, наблюдение
1. Переменные и выражения. Организация ввода и вывода данных. Операции.					
4		Переменные	1	Комбинированное занятие	Беседа, наблюдение
5		Выражения	1	Комбинированное занятие	Беседа, наблюдение
6		Операции. Элементарные действия с числами	2	Практическое занятие	Беседа, наблюдение
7		Ввод и вывод данных	2	Практическое занятие	Беседа, наблюдение
2. Программирование ветвящихся алгоритмов.					
8		Логические выражения и операторы. Условный оператор. Множественное	2	Комбинированное занятие	Беседа, наблюдение

		ветвление.			
9		Реализация ветвления на языке Python	2	Комбинированное занятие	Беседа, наблюдение
10		Решение задач на программирование ветвящихся алгоритмов	2	Практическое занятие	Беседа, наблюдение
3. Циклы					
11		Оператор цикла с условием Оператор While	2	Комбинированное занятие	Беседа, наблюдение
12		Оператор цикла с параметром FOR	2	Комбинированное занятие	Беседа, наблюдение
13		Вложенные циклы	2	Комбинированное занятие	Беседа, наблюдение
14		Реализация циклических алгоритмов	2	Практическое занятие	Беседа, наблюдение
4. Стил ь программирования и отладка программ					
15		Стил ь программирования	1	Комбинированное занятие	Беседа, наблюдение
16		Отладка программ	1	Комбинированное занятие	Беседа, наблюдение
5. Работа над проектами					
17		Требования к проектам. Выбор темы	2	Комбинированное занятие	Беседа, наблюдение
18		Разработка алгоритма и написание программного	2	Практическое занятие	Беседа, наблюдение

		кода			
19		Работа над тестированием и отладкой программ	2	Практическое занятие	Беседа, наблюдение
20		Защита проектов	2	Комбинированное занятие	Беседа, наблюдение

Условия реализации программы: материально-техническое, информационное и кадровое обеспечение

Материально-техническое обеспечение программы:

- кабинет Информатики
- комплект столов и стульев на 16 посадочных мест;
- стол для педагога;
- раздаточный материал
- ноутбуки с комплектом программ по изучению языка программирования;
- Телевизор;
- Интернет.

Информационное обеспечение:

- методические и дидактические материалы
- презентации, подготовленные к каждому занятию.

Кадровое обеспечение программы.

Реализацию программы обеспечивает педагог дополнительного образования, обладающий не только профессиональными знаниями, но и компетенциями в организации и ведении образовательной деятельности творческого объединения.

Формы аттестации/контроля

Система оценивания -безотметочная. Используется только словесная оценка достижений учащихся.

Формы отслеживания и фиксации образовательных результатов: аналитическая справка, аналитический материал, видеозапись, журнал посещаемости.

Формы предъявления и демонстрации образовательных результатов: аналитическая справка, демонстрация программ, контрольная работа, презентация итогового проекта перед родителями и педагогами.

Оценочные материалы

Для оценки результативности программы используются следующие методики и диагностики:

- наблюдение в ходе обучения с фиксацией результата;
- проведение проверочных работ;
- анализ, обобщение и обсуждение результатов обучения;
- проведение открытых занятий с их последующим обсуждением;
- участие в проектной деятельности школы, города;
- участие в соревнованиях муниципального, зонального и регионального уровней;
- оценка выполненных практических работ.

Методические материалы

- Раздаточный материал контролирующего и обучающего характера по каждой теме.
- Задания на практические работы.
- Карточки с индивидуальными заданиями.

- Раздаточный материал справочного характера.
- Раздаточный материал теоретического характера.
- Демонстрационные материалы в электронном виде.

Диагностический инструментарий и формы контроля

Для определения уровня знаний, умений, навыков обучающихся и проведения диагностики используется трехуровневая система:

Высокий уровень:

- сфера знаний и умений: отличное владение понятийным аппаратом, безошибочно и точное, грамотное выполнение заданий, правильная работа, соблюдение правил ТБ при работе с техникой, точное планирование своей работы;
- сфера творческой активности: обучающийся проявляет выраженный интерес к занятиям, творческой деятельности, обстановке и педагогу; активно принимает участие в конкурсах различного уровня;
- сфера личностных результатов: прилагает усилия к преодолению трудностей; слаженно работает в коллективе, умеет выполнять задания самостоятельно.

Средний уровень:

- сфера знаний и умений: знание базовых понятий, соблюдение правил ТБ при работе с компьютерами, выполнение заданий с допущением неточности; не достаточно рациональное использование рабочего времени;
- сфера творческой активности: включение обучающихся в работу достаточно активно (с желанием), или с проявлением интереса к работе, но присутствует быстрая утомляемость; участие в конкурсах (внутриучрежденческого и городского уровней);
- сфера личностных результатов: планирование работы по наводящим вопросам педагога или самостоятельно, но с небольшими погрешностями; возникновение трудностей при работе в коллективе (присутствует желание добиться положительного результата в работе).

Низкий уровень:

- сфера знаний и умений: слабое развитие понятийного аппарата, отсутствие достаточного уровня работы с языком программирования;
- сфера творческой активности: начало выполнения задания только после дополнительных побуждений, а во время работы частое переключение внимания, выполнение заданий недостаточно грамотно;
- сфера личностных результатов: нерациональное использование времени; планирование собственной работы только по наводящим вопросам педагога, не

умение выполнять задания.

Программой «Робототехника» предусматриваются следующие виды контроля: предварительный, текущий, итоговый, а также промежуточный, результаты которых фиксируются в листах оценивания.

Предварительный контроль проводится в первые дни обучения для выявления исходного уровня подготовки обучающихся, чтобы скорректировать учебно-тематический план, определить направление и формы индивидуальной работы (метод: анкетирование, собеседование).

Промежуточный контроль. В конце каждой четверти проводится итоговое занятие в форме зачета, состоящего из практической и теоретической частей. Проверка теоретического материала осуществляется в письменной форме (составляется из вопросов по каждому разделу программы). Практическая часть состоит из проверки умений и навыков по работе в системе программирования.

Текущий контроль проводится с целью определения степени усвоения обучающимися учебного материала и уровня их подготовленности к занятиям. Этот контроль должен повысить заинтересованность обучающихся в усвоении материала. Он позволяет своевременно выявлять отстающих, а также опережающих обучение с целью наиболее эффективного подбора методов и средств обучения.

Итоговый контроль проводится с целью определения степени достижения результатов обучения, закрепления знаний, ориентации обучающихся на дальнейшее самостоятельное обучение, участие в мероприятиях, конкурсах. На каждом занятии педагог использует взаимоконтроль и самоконтроль. Формы контроля: зачет, тестирование, письменный опрос, анкетирование, самостоятельная работа, педагогическое наблюдение.

Формы подведения итогов:

- участие в конкурсах, соревнованиях, сетевых проектах;
- выставки технического творчества;
- результаты работ обучающихся фиксируются на фото и видео в момент демонстрации созданных ими роботов из имеющихся в наличии учебных конструкторов по робототехнике;
- фото и видео материалы по результатам работ размещаются на сайте учреждения; предлагаются для участия на фестивалях и олимпиадах различных уровней

Список литературы Литература для учителя:

1. Домашняя страница Python www.python.org. Справочные материалы, официальная документация.
2. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>.
3. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>.
4. Сайт проекта OpenBookProject openbookproject.net содержит серию практических примеров на Python Криса Мейерса.
5. *Python. Подробный справочник* Дэвида М.Бизли—книга со справочной информацией о языке Python и модулях стандартной библиотеки.
6. *Python. Справочник* Марка Лутца. Справочник по наиболее часто используемым функциям и модулям.

Литература для учащегося:

1. Домашняя страница Python www.python.org. Справочные материалы, официальная документация.
2. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>.
3. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>.
4. Сайт проекта OpenBookProject openbookproject.net содержит серию практических примеров на Python Криса Мейерса.
5. *Python. Подробный справочник* Дэвида М.Бизли—книга со справочной информацией о языке Python и модулях стандартной библиотеки.
6. *Python. Справочник* Марка Лутца. Справочник по наиболее часто используемым функциям и модулям.