МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Комитет образования, науки и молодежной политики Волгоградской области
Комитет по образованию администрации Среднеахтубинского муниципального района Волгоградской области
МОУ СОШ №1 р.п.Средняя Ахтуба

РАССМОТРЕНО
Педагогическом Совете
Протокол № 1
от «28» августа 2026г.

УТВЕРЖДЕНО
Директор



Усков Д.В.
Приказ № 85/11
от 28.08.2026г.

РАБОЧАЯ ПРОГРАММА

по внеурочной деятельности курса

«ОСНОВЫ ПРОГРАММИРОВАНИЯ»

для 8-х классов основного общего образования

на 2026 – 2027 учебный год

Составитель: Демина Ирина Дмитриевна,
учитель информатики

р.п. Средняя Ахтуба 2026

Пояснительная записка

Данная программа естественнонаучной направленности «Язык программирования Python» модифицирована и рассчитана на 34 часа.

Изменение взглядов на предмет информатики как науки, её место в системе научного знания требует существенных изменений в содержании образования по информатике. В связи с этим особую актуальность приобретают раскрытие личностных резервов учащихся и создание соответствующей среды.

Никакая система задач, какой бы хорошей она ни была, никакие тренинги памяти, внимания и т. п. не дают того эффекта, который возникает в случае, если учащиеся осознают необходимость решения тех или иных задач, если у них появляется острая необходимость к преодолению интеллектуальных трудностей, связанных с познанием, если они видят смысл в сотрудничестве с ровесниками и учителем.

Содержание обучения, представленное в общеразвивающей программе «Язык программирования Python», позволяет вести обучение подростков в режиме актуального познания. Практическая направленность программы на создание внешних образовательных продуктов — блок-схем, алгоритмов, программ — способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у школьников знаний. Возникающие при этом познавательные переживания обусловливают сознательное отношение к изучению основных теоретических положений информатики.

Проявления трудолюбия, целеустремленности, возникающие при воплощении замыслов учащихся в рамках общеразвивающей программы «Язык программирования Python», стимулируют развитие индивидуально-личностных качеств школьников.

Активизация познавательного процесса позволяет учащимся более полно выражать свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт предпосылки по применению освоенных навыков программирования в других учебных курсах, а также способствует возникновению дальней мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

Общеразвивающая программа служит средством внутрипрофильной специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся.

Концепция общеразвивающей программы

Ключевой особенностью программы является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и реализации алгоритма с помощью средств программирования.

- В рамках общеразвивающей программы «Язык программирования Python» изучение основ программирования на языке Python — это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др. Исключительно велика роль программирования для формирования мышления школьников, приёмов умственных действий, умения строить модели, самостоятельного нахождения и составления алгоритмов решения задач, умения чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования общеинтеллектуальных и общеучебных умений школьников активизирует процесс индивидуально-личностного становления учащихся.

Общепедагогическая направленность занятий – гармонизация индивидуальных и социальных аспектов обучения по отношению к информационным технологиям. Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции — одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных школьников. Особая роль отводится широко представленной в общеразвивающей программе системе рефлексивных заданий. Освоение рефлексии направлено на осознание учащимися того важного обстоятельства, что наряду с разрабатываемыми ими продуктами в виде программ на компьютере рождается основополагающий образовательный продукт: освоенный инструментарий. Именно этот образовательный продукт станет базой для творческого самовыражения учащихся в форме различных программ.

Цели изучения общеразвивающей программы:

- понять значение алгоритмизации как метода познания окружающего мира, принципы структурной алгоритмизации;
- овладеть базовыми понятиями теории алгоритмов;
- научиться разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

Задачи общеразвивающей программы:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы;
- 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.

Содержание общеразвивающей программы:

Тема 1. Знакомство с языком Python

Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python.

Что такое программа. Первая программа. Структура программы на языке Python.

Комментарии.

Практическая работа 1.1. Установка программы Python Практическая работа 1.2. Режимы работы с Python

Учащиеся должны знать / понимать:

- понятие программы;
- структура программы на Python;
- режимы работы с Python.

Учащиеся должны уметь:

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе.

Тема 2. Переменные и выражения

Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.

Выражения. Операции. Порядок выполнения операций. Математические функции.

Композиция.

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.

Практическая работа 2.1. Работа со справочной системой Практическая работа 2.2. Переменные Практическая работа 2.3. Выражения

Практическая работа 2.5. Задачи на элементарные действия с числами

Учащиеся должны знать / понимать:

- общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- операторы ввода-вывода.

Учащиеся должны уметь:

- пользоваться интерфейсом среды программирования Python;
- использовать команды редактора;
- организовывать ввод и вывод данных;
- записывать арифметические выражения.

Тема 3. Условные предложения

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.

Практическая работа 3.1. Логические выражения Практическая работа 3.2. "Условный оператор" Практическая работа 3.3. Множественное ветвление Практическая работа 3.4. "Условные операторы"

Самостоятельная работа № 1. Решение задач по теме "Условные операторы".

Зачетная работа № 1. "Составление программ с ветвлением".

Учащиеся должны знать / понимать:

- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not; Учащиеся должны уметь:
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов.

Тема 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.

Практическая работа 4.1. "Числа Фибоначчи" Практическая работа 4.2. Решение задачи с циклом for. Практическая работа 4.3. Реализация циклических алгоритмов Практическая работа 4.4. Случайные числа Практическая работа 4.5. Решение задач с циклом. Самостоятельная работа № 2 "Составление программ с циклом"

Творческая работа № 1. "Циклы"

Учащиеся должны знать / понимать:

- циклы с условием и их виды;
- правила записи циклов условием;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.

Учащиеся должны уметь:

- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;

определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи;

Планируемые результаты программы

- рамках курс «Языки программирования. Python» учащиеся овладевают следующими знаниями, умениями и способами деятельности:
  - умеют составлять алгоритмы для решения задач;
  - умеют реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
  - владеют основными навыками программирования на языке Python;
  - умеют отлаживать и тестировать программы, написанные на языке Python.

Учебный план

	Название раздела, темы	Количество часов			Формы аттестации, контроля
		всего	теория	практика	
	Знакомство с языком Python	4	2	2	Текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий
	Переменные и выражения	8	4	4	Текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий
	Условные предложения	10	4	6	Зачет
	Циклы	12	5	7	Творческая работа

Календарный учебный график

№	Дата	Форма занятия	Кол-во часов	Тема занятия	Место проведен	Форма контроля
1	03.09.21	Лекция	1	Общие сведения о языке	Кабинет 1-3	Текущий рефлексивный самоанализ
2	10.09.21	Практическая работа	1	Практическая работа 1.1.Установка программы Python	Кабинет 1-3	контроль и самооценка учащимися выполняемых
3	17.09.21	Лекция	1	Режимы работы программы Python	Кабинет 1-3	Текущий рефлексивный самоанализ
4	24.09.21	Практическая работа	1	Практическая работа 1.2. Режимы работы с Python	Кабинет 1-3	контроль и самооценка учащимися выполняемых
5	01.10.21	Лекция	1	Переменные	Кабинет 1-3	Текущий рефлексивный самоанализ
6	08.10.21	Практическая работа	1	Практическая работа 2.1. Работа со справочной системой	Кабинет 1-3	контроль и самооценка учащимися выполняемых
7	15.10.21	Практическая работа	1	Практическая работа 2.2. Переменные	Кабинет 1-3	контроль и самооценка учащимися выполняемых

8	22.10.21	Лекция	1	Выражения	Кабинет 1-3	Текущий рефлексивный самоанализ
9	29.10.21	Практическая работа	1	Практическая работа 2.3. Выражения	Кабинет 1-3	контроль и самооценка учащимися выполняемых
10	12.11.21	Лекция	1	Ввод и вывод	Кабинет 1-3	Текущий рефлексивный самоанализ
11	19.11.21	Практическая работа	1	Задачи на элементарные действия с числами	Кабинет 1-3	контроль и самооценка учащимися выполняемых
12	26.11.21	Практическая работа	1	Практическая работа 2.5. Задачи на элементарные действия с числами	Кабинет 1-3	контроль и самооценка учащимися выполняемых
13	03.12.21	Лекция	1	Логические выражения и операторы	Кабинет 1-3	Текущий рефлексивный самоанализ
14	10.12.21	Практическая работа	1	Практическая работа 3.1. Логические выражения	Кабинет 1-3	контроль и самооценка учащимися выполняемых
15	17.12.21	Лекция	1	Условный оператор	Кабинет 1-3	Текущий рефлексивный самоанализ
16	24.12.21	Практическая работа	1	Практическая работа 3.2. "Условный оператор"	Кабинет 1-3	контроль и самооценка учащимися выполняемых
17	14.01.22	Лекция	1	Множественное ветвление	Кабинет 1-3	Текущий рефлексивный самоанализ
18	21.01.22	Практическая работа	1	Практическая работа 3.3. Множественное ветвление	Кабинет 1-3	контроль и самооценка учащимися выполняемых
19	28.01.22	Лекция	1	Реализация ветвления в языке Python	Кабинет 1-3	Текущий рефлексивный самоанализ
20	04.02.22	Практическая работа	1	Практическая работа 3.4. "Условные операторы"	Кабинет 1-3	контроль и самооценка учащимися выполняемых
21	11.02.22	Практическая работа	1	Самостоятельная работа № 1 по теме "Условные операторы"	Кабинет 1-3	контроль и самооценка учащимися выполняемых

22	18.02.22	Практическая работа	1	Зачетная работа № 1. "Составление программ с	Кабинет 1-3	Зачет
23	25.02.22	Лекция	1	Оператор цикла с условием	Кабинет 1-3	Текущий рефлексивный самоанализ
24	04.03.22	Практическая работа	1	Практическая работа 4.1. "Числа Фибоначчи"	Кабинет 1-3	контроль и самооценка учащимися выполняемых
25	11.03.22	Лекция	1	Оператор цикла for	Кабинет 1-3	Текущий рефлексивный самоанализ
26	18.03.22	Практическая работа	1	Практическая работа 4.2. Решение задачи с циклом for.	Кабинет 1-3	контроль и самооценка учащимися
27	25.03.22	Лекция	1	Вложенные циклы	Кабинет 1-3	Текущий рефлексивный самоанализ
28	01.04.22	Практическая работа	1	Практическая работа 4.3. Реализация циклических алгоритмов	Кабинет 1-3	контроль и самооценка учащимися выполняемых
	08.04.22	Лекция	1	Случайные числа	Кабинет 1-3	Текущий рефлексивный самоанализ
	15.04.22	Практическая работа	1	Практическая работа 4.4. Случайные числа	Кабинет 1-3	контроль и самооценка учащимися выполняемых
	22.04.22	Практическая работа	1	Примеры решения задач с циклом	Кабинет 1-3	контроль и самооценка учащимися выполняемых
	29.04.22	Практическая работа	1	Практическая работа 4.5. Решение задач с циклом	Кабинет 1-3	контроль и самооценка учащимися выполняемых
	06.05.22	Практическая работа	1	Самостоятельная работа № 2 "Составление программ с	Кабинет 1-3	контроль и самооценка учащимися выполняемых
	13.05.22	Практическая работа	1	Творческая работа № 1. "Циклы"	Кабинет 1-3	Презентация творческих работ

Условия реализации программы

Материально-техническое обеспечение программы

Компьютерный класс + Программное обеспечение:

1. Операционная система: Windows XP (или выше).
2. Среда разработки: Python 3.3 (или выше)

Методы обучения

Отбор методов обучения обусловлен необходимостью формировать информационную и коммуникативную компетентности учащихся, реализовывать личностно-ориентированное обучение, направлять их на самостоятельное решение разнообразных проблем, развивать исследовательские и творческие способности. Решение данных задач кроется в организации деятельностного подхода к обучению, в проблемном изложении материала учителем, в переходе от репродуктивного вида работ к самостоятельным, поисково-исследовательским видам деятельности. Поэтому основная методическая установка в данном курсе — обучение учащихся навыкам самостоятельной творческой деятельности.

Способы оценивания уровня достижений учащихся

Предметом диагностики и контроля в курсе «Основы программирования на языке Python» являются внешние образовательные продукты учащихся (созданные блок-схемы, программы), а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам курса.

Качество внешней образовательной продукции желательно оценивать по следующим параметрам:

- алгоритм должен быть оптимальным по скорости выполнения и максимально простым в реализации на языке программирования;
- программа должна выполнять поставленные задачи;
- по степени «читаемости кода» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.).

Проверка достигаемых учащимися результатов производится в следующих формах:

- текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
- текущая диагностика и оценка учителем деятельности школьников;
- итоговая оценка деятельности и образовательной продукции ученика в соответствии с его индивидуальной образовательной программой освоения программы;

Итоговый контроль проводится в конце всего курса тестированием.

Литература и источники

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.

3. Сайт проекта Open Book Project openbookproject.net содержит серию практических примеров на Python Криса Мейерса.

3. *Python. Подробный справочник* Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.

3. *Python. Справочник* Марка Лутца. Справочник по наиболее часто использующимся функциям и модулям.