

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Муниципальное автономное общеобразовательное учреждение
«Центр образования «Аксиома»

ВЫПИСКА ИЗ ОСНОВНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ
СРЕДНЕГО ОБЩЕГО ОБРАЗОВАНИЯ,
утвержденной приказом директора Центра образования «Аксиома» от 01.09.2025 г. № 221

РАБОЧАЯ ПРОГРАММА

учебного курса «Программирование на Python» (углубленный уровень)

среднего общего образования

10-11 классы

Информатика в настоящее время - одна из фундаментальных отраслей научного знания, формирующая системно-информационный подход к анализу окружающего мира, изучающая информационные процессы, методы и средства получения, преобразования, передачи, хранения и использования информации, стремительно развивающаяся и постоянно расширяющаяся область практической деятельности человека, связанная с использованием информационных технологий.

В современных условиях образовательная деятельность в области информационно-коммуникационных технологий является чрезвычайно востребованной. Поэтому разработано достаточное число образовательных программ в данном направлении. Информатика отличается от большинства технических дисциплин своей практической направленностью и чрезвычайной изменчивостью предмета изучения, связанной с динамичным развитием аппаратных и программных средств. Эта изменчивость предмета влечет за собой постоянное обновление образовательных программ.

В связи с развитием и внедрением в повседневную жизнь информационно-коммуникационных технологий возрос интерес к программированию.

В школьном курсе информатики вопросы программирования рассматриваются лишь в ознакомительном плане и на это выделяется недостаточное количество часов, как следствие – формальное восприятие учащимися основ современного программирования. Образовательная программа «Программируем вместе» направлена на устранение данного пробела. Это особенно актуально в новом информационном обществе, где пользователей очень часто не устраивают возможности программ и им хочется адаптировать приложения для своих конкретных потребностей.

Знания, полученные при изучении образовательной программы «Программирование на Python», учащиеся могут использовать для самостоятельного написания программ. Кроме этого в процессе обучения у учащихся формируются навыки программирования, представление о профессии программиста, механизм работы и устройство операционной системы Windows. Знания и умения, приобретенные в результате освоения программы «Программирование на Python», являются фундаментом для дальнейшего совершенствования мастерства в области объектно-ориентированного программирования, а также помогут учащимся в дальнейшем обучении в вузах и в профессиональной деятельности.

Рабочая программа учебного курса «Программирование на Python» составлена в соответствии с требованиями Федерального государственного образовательного стандарта среднего общего образования.

Основной целью учебного курса является обучение программированию через создание творческих проектов по информатике. Курс развивает творческие способности учащихся, а также закладывает пропедевтику наиболее значимых тем курса информатики и позволяет успешно готовиться к участию в олимпиадах по программированию.

Изучение курса позволит учащимся познакомиться с концепцией программирования в средах программирования Python.

На протяжении учебного курса рассматриваются базовые приемы программирования, такие как написание программного кода, создание форм, объявление переменных, вычисление выражений, использование ветвлений, выбора и циклических конструкций и многое, многое другое. При этом осваиваются приемы создания различных программ (приложений).

Описание места учебного курса в учебном плане:

Программа учебного курса «Программирование на Python» разработана для обучающихся 10-11-х общеобразовательных классов и рассчитана на 68 часа (1 раз в неделю по 45 мин.). Программа составлена с учетом санитарно-гигиенических требований, возрастных особенностей обучающихся (непрерывная работа за ПК не более 25 мин.) и рассчитана на работу в учебном компьютерном классе.

Срок освоения программы – 2 года.

Методы обучения:

- самостоятельная индивидуальная работа;
- групповая работа;
- проектная работа.

Формы обучения:

- мини-лекции с элементами дискуссии;
- компьютерные практикумы.

Занятия включают лекционную и практическую часть. В качестве основной формы организации учебных занятий используется выполнение обучающимися практических заданий за компьютером (компьютерный практикум). Роль учителя состоит в небольшом по времени объяснении нового материала и постановке задачи, а затем консультировании обучающихся в процессе выполнения ими практической работы.

Текущий контроль выполняется по результатам выполнения учениками практических заданий на компьютере.

Итоговый контроль осуществляется по результатам защиты итоговых проектов. В начале курса каждому учащемуся предлагается в течение всего времени изучения курса разработать проект для решения некоторой задачи. В процессе защиты учащийся должен представить работающую компьютерную программу, которая решает поставленную перед ним задачу, и обосновать способ ее решения.

На каждом уроке материал излагается следующим образом:

- повторение основных понятий и методов работы с ними;
- изучение нового материала;
- основные приёмы работы (самостоятельная практическая работа);
- проекты для самостоятельного выполнения.

Формы подведения итогов реализации программы:

- опрос,
- наблюдение,
- защита проектов.

Результаты освоения курса

В ходе изучения курса в основном формируются и получают развитие следующие *метапредметные результаты*:

- умение самостоятельно планировать пути достижения целей, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в рамках предложенных условий и требований, корректировать свои действия в соответствии с изменяющейся ситуацией;
- умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально и в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учета интересов; формулировать, аргументировать и отстаивать свое мнение; формирование и развитие компетентности в области использования информационно-коммуникационных технологий (далее ИКТ-компетенции).

Вместе с тем вносится существенный вклад в развитие *личностных результатов*:

- формирование ответственного отношения к учению, готовности и способности обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию, осознанному выбору и построению дальнейшей индивидуальной траектории образования на базе ориентировки в мире профессий и профессиональных предпочтений, с учетом устойчивых познавательных интересов, а также на основе формирования уважительного отношения к труду, развития опыта участия в социально значимом труде;

- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, общественно полезной, учебно-исследовательской, творческой и других видов деятельности.

В части развития **предметных результатов** наибольшее влияние изучение курса оказывает на:

- формирование информационной и алгоритмической культуры; формирование представления о компьютере как универсальном устройстве обработки информации; развитие основных навыков и умений использования компьютерных устройств;

- формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умения соблюдать нормы информационной этики и права.

Регулятивные универсальные учебные действия. Обучающийся научится:

- целеполаганию, включая постановку новых целей, преобразование практической задачи в познавательную;

- самостоятельно анализировать условия достижения цели на основе учета выделенных учителем ориентиров действия в новом учебном материале;

- планировать пути достижения целей;

- уметь самостоятельно контролировать свое время и управлять им.

Коммуникативные универсальные учебные действия.

Обучающийся научится:

- учитывать разные мнения и стремиться к координации различных позиций в сотрудничестве;

- аргументировать свою точку зрения, спорить и отстаивать свою позицию не враждебным для оппонентов образом;

- задавать вопросы, необходимые для организации собственной деятельности и сотрудничества с партнером;

- осуществлять взаимный контроль и оказывать в сотрудничестве необходимую взаимопомощь.

- устанавливать и сравнивать разные точки зрения, прежде чем принимать решения и делать выбор.

Познавательные универсальные учебные действия.

Обучающийся научится:

- осуществлять выбор наиболее эффективных способов решения задач, в зависимости от конкретных условий;

- давать определение понятиям;

- устанавливать причинно-следственные связи;

- осуществлять логическую операцию установления родовидовых отношений, ограничение понятия;

- обобщать понятия — осуществлять логическую операцию перехода от видовых признаков к родовому понятию, от понятия с меньшим объемом к понятию с большим объемом;

- строить логическое рассуждение, включающее установление причинно-следственных связей.
- создавать и преобразовывать модели и схемы для решения задач.

В результате изучения программирования учащиеся должны

знать:

- структуру программы на языке Python
- понятие величины
- математические функции и операции языка
- типы данных
- операторы и их синтаксис
- правила написания процедур и функций на языке паскаль; их различия
- понятие одномерного и двумерного массива
- алгоритмы поиска и сортировки

уметь:

- «читать» готовую программу
- записывать математические выражения на языке паскаль
- описывать словесно работу операторов
- разрабатывать программы, требующие вложения одного и более операторов
- выделять вспомогательные алгоритмы в сложных задачах.
- формировать процедуры и функции. правильно строить обращения к процедурам и функциям.
- описывать, заполнять и выводить массивы

СОДЕРЖАНИЕ УЧЕБНОГО КУРСА

Язык программирования Python.

Языки программирования как класс специализированного программного обеспечения. Язык программирования Паскаль. Алфавит языка. Структура программы. Идентификаторы и служебные слова. Выражения и операции. Описание переменных и констант. Описание типов. Логические выражения.

Запуск и настройка программы Python. Создание, запуск, сохранение программы. Структура программы. Служебные слова. Отладка программного кода.

Основные алгоритмические конструкции. Способы записи алгоритмов. Основные алгоритмические конструкции. Основные блоки программы. Присваивание. Переменные. Арифметические, строковые, логические операции. Ввод и вывод данных, оператор присваивания.

Условный оператор. Полная и неполная развилка. Оператор выбора. Циклические конструкции. Цикл с предварительным условием. Цикл с последующим условием. Цикл с параметром. Вложенные циклы.

Массивы. Одномерные массивы. Формирование массива и вывод его элементов. Анализ элементов массива. Работа с несколькими массивами. Преобразование массива. Изменение элементов массива. Удаление и вставка элементов. Сортировка массива.

Символьные строки. Символы и строки. Посимвольный анализ и преобразование строк. Строки и числа. Обработка строк с помощью стандартных функций. Поиск и замена. Анализ и преобразование слов в строке.

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. Списки: примеры решения задач.

Тематическое планирование

№ п/п	Название темы	Количество часов
1.	Языки программирования как класс специализированного программного обеспечения. Техника безопасности.	2
2.	Синтаксис языков программирования.	2
3.	Запуск и настройка Python.	2
4.	Создание, компиляция, исполнение и отладка программ.	2
5.	Синтаксис и семантика языка Python. Типичные ошибки. Сообщения об ошибках.	2
6.	Данные. Типы данных. Константы.	2
7.	Числовой тип данных.	2
8.	Построение арифметических выражений.	2
9.	Оператор присваивания. Выполнение оператора присваивания.	2
10.	Операторы ввода и вывода данных.	2
11.	Встроенные функции. Операции над числами.	2
12.	Создание и отладка элементарной программы.	2
13.	Данные логического типа и логические выражения.	2
14.	Организация программ разветвляющейся структуры.	2
15.	Условный оператор. Полная и неполная формы оператора.	2
16.	Оператор выбора.	2
17.	Оператор цикла с предусловием.	2
18.	Оператор цикла с постусловием.	2
19.	Оператор цикла с известным числом повторений (параметром).	2
20.	Вложенность циклов.	2
21.	Одномерные массивы. Описание массива.	2
22.	Задачи на получение нового массива.	2
23.	Поиск в массиве.	2
24.	Сортировка массива.	2
25.	Двумерный массив. Обработка двумерного массива.	2
26.	Строковый, символьный тип данных. Основные операции.	2
27.	Стандартные функции, процедуры для работы с символьными величинами.	2
28.	Операции поиска и замены в символьных строках и массивах.	2
29.	Программирование алгоритмов обработки строк.	2

30.	Списки. Срезы списков	2
31.	Списки: примеры решения задач.	2
32.	Работа над программным проектом.	2
33.	Защита проектов.	2
34.	Перспективы развития технологий программирования. Итоговое занятие.	2