

Выписка из основной образовательной программы  
основного общего образования МБОУ СОШ № 186  
(утвержденной приказом №138-од от 27.08.2026г.)

Муниципальное бюджетное общеобразовательное учреждение города Новосибирска  
«Средняя общеобразовательная школа № 186»

**Рабочая программа курса  
«Основы программирования» по информатике  
для 9 класса (17 учебных часов)**

## Пояснительная записка

**Цель курса:** сформировать у учащихся базовые навыки алгоритмического мышления и практического программирования, подготовить к осознанному применению программных средств для решения прикладных задач и к дальнейшему изучению IT-дисциплин.

### Задачи курса:

- познакомить с основными понятиями программирования (алгоритм, переменная, тип данных, структура управления);
- освоить синтаксис и базовые конструкции языка программирования (на примере Python как наиболее доступного для начинающих);
- научиться разрабатывать, тестировать и отлаживать простые программы;
- развить умение разбивать задачу на подзадачи и выбирать подходящий способ реализации;
- сформировать навыки работы с данными (числа, строки, списки), обработки условий и циклов, использования функций.

### Планируемые результаты:

- **Предметные:** учащиеся смогут писать программы с линейной, разветвляющейся и циклической структурой, использовать базовые типы данных и структуры, применять функции для структурирования кода, решать типовые задачи (поиск максимума/минимума, фильтрация данных, перебор вариантов).
- **Метапредметные:** развитие логического и системного мышления, умения планировать этапы решения, проверять гипотезы, анализировать ошибки и корректировать код.
- **Личностные:** формирование уверенности в работе с технологиями, готовности к самостоятельному поиску информации и освоению новых инструментов, ответственного отношения к результатам своей работы.

## Тематическое планирование (17 часов)

| № | Тема занятия                                                        | Содержание                                                                                                                                                                                          | Виды деятельности учащихся                                                                                                     | Кол-во часов |
|---|---------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|--------------|
| 1 | Введение в программирование. Техника безопасности. Среда разработки | Понятие алгоритма и программы. Роль программирования в современном мире. Правила работы в компьютерном классе. Установка и настройка среды (IDLE/VS Code/онлайн-редактор). Первый запуск программы. | Просмотр демо-примеров. Настройка рабочего места. Написание и запуск первой программы (print). Обсуждение правил безопасности. | 1            |
| 2 | Переменные, типы данных, ввод/вывод                                 | Переменные и правила именования. Основные типы: int, float, str, bool. Операции над данными. Ввод данных через input(), вывод через print(). Преобразование типов.                                  | Практикум: написание программ для ввода и вывода данных, выполнение арифметических операций, обработка                         | 1            |

|    |                                      |                                                                                                                                                     |                                                                                                                               |   |
|----|--------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|---|
|    |                                      |                                                                                                                                                     | ошибок при преобразовании типов.                                                                                              |   |
| 3  | Арифметика и логика в программах     | Арифметические операции, приоритет. Деление, целочисленное деление, остаток от деления. Логические выражения, операторы сравнения, булевы значения. | Решение задач на расчёт (стоимость, сдача, конвертация единиц). Построение логических выражений.                              | 1 |
| 4  | Условные конструкции: if, elif, else | Простые и составные условия. Логические операторы and, or, not. Вложенные условия. Практические задачи на выбор действия.                           | Написание программ с ветвлением: определение чётности, проверка диапазона, классификация данных (например, оценка по баллам). | 1 |
| 5  | Циклы: while                         | Понятие цикла. Условие продолжения. Инкремент/декремент счётчика. Бесконечные циклы и способы выхода. Обработка ошибок ввода.                       | Практикум: суммирование чисел, поиск первого подходящего значения, валидация ввода пользователя.                              | 1 |
| 6  | Циклы: for и перебор коллекций       | Цикл по диапазону (range). Перебор элементов списка, строки. Использование индексов.                                                                | Решение задач: вывод последовательностей, подсчёт статистики по данным, обработка строк (подсчёт символов, поиск подстроки).  | 1 |
| 7  | Списки и работа с коллекциями        | Создание списков, доступ по индексу, срезы. Методы списков: append, remove, sort, len. Изменение и фильтрация элементов.                            | Практическая работа: формирование списка данных, сортировка, удаление дубликатов, выборка по условию.                         | 1 |
| 8  | Строки и базовые операции с текстом  | Индексация, срезы строк. Методы строк: lower, upper, replace, split, join. Проверка вхождения подстроки.                                            | Задачи: очистка текста от лишних символов, форматирование, разбиение строки на слова, подсчёт частоты слов.                   | 1 |
| 9  | Функции: определение и вызов         | Понятие функции. Параметры и аргументы. Возвращаемые значения (return). Область видимости переменных.                                               | Создание собственных функций для типовых операций (расчёт площади, конвертация валют, проверка корректности данных).          | 1 |
| 10 | Работа с файлами (чтение и запись)   | Открытие файла, режимы доступа. Чтение строк, запись данных. Обработка исключений при работе с файлами.                                             | Практикум: чтение списка имён из файла, подсчёт статистики, сохранение результатов в файл.                                    | 1 |
| 11 | Обработка ошибок и исключений        | Понятие исключения. Конструкция try-except. Типовые ошибки: ValueError, FileNotFoundError.                                                          | Написание надёжных программ: защита от некорректного ввода, обработка отсутствия                                              | 1 |

|    |                                                      |                                                                                                                                                |                                                                                                       |   |
|----|------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|---|
|    |                                                      |                                                                                                                                                | файла, логирование ошибок.                                                                            |   |
| 12 | Словари и структурированные данные                   | Создание словарей, доступ к значениям, добавление и удаление элементов. Перебор ключей и значений.                                             | Практическое применение: хранение данных пользователя, подсчёт частоты элементов, группировка данных. | 1 |
| 13 | Модули и библиотеки (обзор)                          | Понятие модуля. Импорт модулей. Использование готовых функций (например, <code>math</code> , <code>random</code> ). Генерация случайных чисел. | Примеры: расчёт тригонометрических значений, генерация паролей, моделирование бросков кубика.         | 1 |
| 14 | Простые алгоритмы и анализ задач                     | Этапы решения задачи: постановка, моделирование, выбор структуры, реализация, тестирование. Разбор типовых шаблонов.                           | Самостоятельное решение задач с предварительным планированием (блок-схема или псевдокод).             | 1 |
| 15 | Мини-проект: консольное приложение                   | Выбор темы (калькулятор, викторина, список дел). Проектирование структуры, написание кода, тестирование.                                       | Работа над проектом: определение требований, разбиение на модули, реализация функционала, отладка.    | 1 |
| 16 | Презентация и разбор проектов                        | Демонстрация результатов. Анализ решений, обсуждение альтернатив, выявление ошибок.                                                            | Представление проектов, взаимное рецензирование, разбор типичных ошибок и способов их устранения.     | 1 |
| 17 | Итоговое занятие. Практикум «Найди и исправь ошибку» | Систематизация знаний. Отработка навыков отладки. Работа с чужим кодом.                                                                        | Выполнение заданий на поиск и исправление ошибок, рефлексия по курсу, выдача сертификатов.            | 1 |

## Методические рекомендации

- **Язык и среда:** в качестве основного языка рекомендуется Python (версии 3.x) — он лаконичен, имеет богатую стандартную библиотеку и широко используется в образовании. Для работы подойдут IDLE, VS Code или онлайн-платформы (Replit, Google Colab) — выбор зависит от оснащения кабинета.
- **Дифференциация:** на каждом занятии предлагайте 3 уровня задач: базовый (отработка синтаксиса), повышенный (комбинация конструкций), олимпиадный/проектный (нестандартные условия, оптимизация). Это позволит вовлечь всех учащихся.
- **Формы работы:** чередуйте демонстрацию кода учителем с самостоятельным написанием, парной отладкой и групповыми мини-проектами. Используйте приём «код-ревью» — учащиеся проверяют решения друг друга по чек-листу (корректность, читаемость, обработка ошибок).

- **Оценка:** применяйте критериальное оценивание: правильность результата, чистота кода (имена переменных, комментарии), обработка ошибок, самостоятельность. Итогом курса может стать портфолио с проектами и листингами лучших решений.
- **Материально-техническое обеспечение:** компьютерный класс с доступом в интернет, проектор для демонстрации кода, возможность сохранения файлов на сетевом диске или в облаке.

### **Пример задачи для занятия №7 (работа со списками)**

Дан список оценок учеников класса: [4, 5, 3, 5, 2, 4, 5]. Напишите программу, которая:

1. выведет все оценки;
2. посчитает средний балл;
3. найдёт количество пятёрок;
4. удалит все двойки и выведет обновлённый список.

#### **Возможное решение:**

Программа рассчитана на последовательное освоение материала и может быть адаптирована под уровень класса и технические возможности школы.