Министерство науки и высшего образования Российской Федерации

Муниципальное бюджетное учреждение дополнительного образования
«Центр детского научного и инженерно-технического творчества»
города Невинномысска

СОГЛАСОВАНО
Педагогическим советом
Протокол № 1
от «27» августа 2026 г.

УТВЕРЖДАЮ
Директор
_____Т.В. Чилхачоян
«27» августа 2026 г.

Дополнительная общеобразовательная программа технической направленности

Алгоритмы и цифровые системы

8-11 класс
Срок реализации программы – 4 года

Авторы-составители:
Бенескул А.В., педагог
Фоменко О.Н., педагог
Завялик О.П., педагог
Чилхачоян Т.В., педагог

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат: 008830DD740B303171238478D0E3864D90
Владелец: ЧИЛХАЧОЯН ТАТЬЯНА ВАЛЕРЬЕВНА
Действителен: с 04.05.2026 по 28.07.2027
Время подписи: 28.08.2026 в 16:04:12
Организация: МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ "ЦЕНТР ДЕТСКОГО НАУЧНОГО И ИНЖЕНЕРНО-ТЕХНИЧЕСКОГО ТВОРЧЕСТВА" ГОРОДА НЕВИННОМЫССКА
Должность: Директор

Подписано с помощью цифрового сервиса «На-Подписи»

Невинномысск, 2026

СОДЕРЖАНИЕ

1. Пояснительная записка
2. Учебно-тематический план и содержание
3. Организационно-педагогические условия реализации программы
4. Список литературы

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа технической направленности «Алгоритмы и цифровые системы» предназначена для обучающихся 8–11 классов и реализуется в рамках дополнительного научно-технического образования. Актуальность программы обусловлена необходимостью системной подготовки школьников к решению актуальных инженерных и IT-задач с использованием языка Python, формирования навыков проектного мышления и освоения современных инструментов разработки программного обеспечения, анализа данных и компьютерного зрения. Программа разработана с учётом возрастных когнитивных особенностей учащихся, обеспечивая плавную прогрессивную нагрузку от синтаксических основ до создания комплексных прикладных систем.

Актуальность программы обусловлена требованиями современного общества к формированию системы работы с заинтересованными инженерным и научно-техническим творчеством детьми в условиях дополнительного образования.

Программа разработана на основе следующих документов:

– закон Российской Федерации «Об образовании» (Федеральный закон от 29 декабря 2012 г. № 273-ФЗ);

– приказ Министерства просвещения РФ от 27 июля 2022 г. №629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

– концепция развития дополнительного образования детей до 2030 года (Распоряжение Правительства РФ от 31 марта 2022 г. №678-р);

– постановление Главного государственного санитарного врача от 28.09.2020 г. № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодёжи»;

– постановление Главного государственного санитарного врача от 28.01.2021 г. № 2 «Об утверждении санитарных правил СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания».

Педагогическая целесообразность программы обуславливается стимулированием интересов учащихся к дисциплинам технического направления, экспериментальным исследованиям, проектной деятельности, в обеспечении адаптации школьников к жизни в обществе, профессиональной ориентации, а также выявлении и поддержке учащихся, проявивших выдающиеся способности.

Программа может быть реализована с помощью дистанционных технологий, технологий смешанного и модульного обучения.

Цель программы – формирование у обучающихся компетенций в области прикладной разработки программного обеспечения, работы с данными и интеллектуальными системами, готовых к самостоятельному проектированию и реализации IT-решений.

Задачи программы:

– Освоить синтаксис, парадигмы и стандартную экосистему языка Python, включая процедурное и объектно-ориентированное программирование;

– Сформировать навыки проектирования и работы со структурами данных, алгоритмами сортировки и поиска, а также реляционными базами данных;

– Развить компетенции в области веб-разработки, проектирования REST API и интеграции внешних сервисов;

– Ознакомиться с методами анализа данных, машинного обучения и компьютерного зрения для решения прикладных задач;

– Обеспечить закрепление теоретических знаний через системную реализацию промежуточных и итоговых проектных работ.

Объем, содержание и планируемые результаты программы определены, исходя из особенностей интересов и потребностей учащихся в области технических наук. Вместе с тем при определении объема и содержания программы учитывалась сложность конкретной темы по отношению к другим темам раздела, возможность приобретения учащимися практического опыта и осуществления межпредметных связей. Порядок тем и блоков, представленный в учебно-тематическом плане, допускается корректировать в пределах программы класса.

К отличительным особенностям программы относится модульно-концентрический подход к построению содержания и выраженная практико-ориентированная направленность. Программа реализует принцип «от простого к сложному»: от синтаксических основ Python и алгоритмического мышления, через объектно-ориентированное проектирование и работу с данными к разработке веб-сервисов, компьютерному зрению и элементам машинного обучения. Каждый модуль завершается защитой проектной работы мини-проект в середине года, итоговый проект – в конце, данный подход помогает развить системное закрепление компетенций через создание реальных программных продуктов. Программа интегрирует междисциплинарные связи с математикой, физикой и инженерными дисциплинами, формируя у обучающихся готовность к решению прикладных задач в сфере IT, участию в олимпиадах и продолжению образования в технических вузах.

Срок реализации программы – 4 года.

Продолжительность учебного года – 34 недели.

Общий объем – 544 часов.

Формы и режим занятий:

Занятия проводятся по 4 часа в неделю в постоянных группах учащихся 8-11 классов (оптимальное количество участников в группе: 8-15 человек), в форме теоретических, практических и индивидуальных занятий, а также консультаций по выполнению проектов.

Ожидаемые результаты освоения программы курса:

1. Самостоятельное проектирование архитектуры программных модулей и реализация их на Python с применением принципов ООП, паттернов проектирования и контроля версий (Git).

2. Умение работать с данными: загрузка, очистка, визуализация и построение базовых прогнозных моделей с использованием библиотек Data Science;

3. Навыки разработки веб-приложений (frontend/backend), настройки API, работы с СУБД и развёртывания сервисов;

4. Опыт применения алгоритмов компьютерного зрения для обработки изображений и видеопотоков в практических задачах;

5. Готовность планировать, разрабатывать, тестировать и публично защищать промежуточные и итоговые проекты, демонстрируя навыки инженерного и алгоритмического мышления.

Результаты освоения программы определяются с использованием пятибалльной шкалы.

Контроль освоения программы – текущий, промежуточный и итоговый.

Текущий контроль осуществляется на занятиях (ответы у доски, письменные работы, практические работы и устные ответы, домашние задания); после изучения блока или набора взаимосвязанных блоков (выделенных педагогом) защита практической работы или устный опрос.

Промежуточный контроль проводится в соответствии с учебно-тематическим планом после освоения темы или набора взаимосвязанных тем в форме контрольной работы, содержащей устную и практическую часть или защиты практической работы.

Итоговый контроль – осуществляется в форме итогового проекта. Ученик может выбрать 3 направления для разработки итогового проекта, общественно полезные проекты, проекты для внутреннего использования, и проекты, направленные на коммерческие компании.

Программой не предусмотрено использование тестов для итогового контроля.

2. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН И СОДЕРЖАНИЕ 8 КЛАСС

[136 часов, 4 часа в неделю]

№	Наименование тем и блоков	Общее количество учебных часов	Теор.	Практ.
Тема 1	**Основы Python**	**30**	**6**	**24**
Блок 1	Синтаксис, типы данных, операторы ввода/вывода	10	2	8
Блок 2	Условные конструкции, циклы, итераторы	10	2	8
Блок 3	Функции, асинхронность, области видимости, обработка исключений	10	2	8
Тема 2	**Структуры данных**	**36**	**10**	**26**
Блок 1	Списки, кортежи, множества, словари	12	4	8
Блок 2	Строки, регулярные выражения, работа с файлами	10	2	8
Блок 3	Введение в алгоритмы сортировки и поиска	14	4	10
	Мини проект (концепт итогового проекта)	**2**		
Тема 3	**Введение в проектную деятельность**	**18**	**4**	**14**
Блок 1	Жизненный цикл ПО, системы контроля версий Git	6	2	4
Блок 2	Тестирование, отладка, документирование кода	12	2	10
Тема 4	**Базы данных и основы веб-разработки**	**32**	**8**	**24**
Блок 1	Основы HTML/CSS, статические сайты	16	4	12
Блок 2	Реляционные СУБД, язык SQL	16	4	12
	Итоговый проект	**2**		
	Резерв	**16**		
	Всего	136		

2. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН И СОДЕРЖАНИЕ 9 КЛАСС

[136 часов, 4 часа в неделю]

№	Наименование тем и блоков	Общее количество учебных часов	Теор.	Практ.
Тема 1	**ООП в Python**	**30**	**6**	**24**
Блок 1	Классы, объекты, конструкторы, инкапсуляция	10	2	8
Блок 2	Наследование, полиморфизм, абстрактные классы	10	2	8
Блок 3	Магические методы, перегрузка операторов, dataclass	10	2	8
Тема 2	**Основы анализа данных**	**36**	**12**	**24**
Блок 1	NumPy: массивы, матричные операции	12	4	8
Блок 2	Pandas: датафреймы, очистка, агрегация данных	12	4	8
Блок 3	Matplotlib: Работа с графиками	12	4	8
	Мини проект (концепт итогового проекта)	**2**		
Тема 3	**Основы машинного обучения**	**22**	**4**	**18**
Блок 1	Scikit-learn: классификация, регрессия, метрики	12	2	10
Блок 2	Кластеризация, подготовка датасетов, валидация	10	2	8
Тема 4	**Разработка мини-приложений**	**28**	**8**	**20**
Блок 1	Архитектура MVC, GUI-фреймворки (Tkinter/CustomTkinter)	14	4	10
Блок 2	Упаковка приложений, работа с конфигурациями	14	4	10
	Итоговый проект	**2**		
	Резерв	**16**		
	Всего	136		

2. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН И СОДЕРЖАНИЕ 10 КЛАСС

[136 часов, 4 часа в неделю]

№	Наименование тем и блоков	Общее количество учебных часов	Теор.	Практ.
Тема 1	**Анализ и обработка изображений. Техника безопасности**	**30**	**6**	**24**
Блок 1	Трансформация изображения	10	2	8
Блок 2	Изменение значений компонентов цвета	10	2	8
Блок 3	Применение фильтров	10	2	8
Тема 2	**Компьютерное зрение: трекинг, распознавание лиц**	**36**	**12**	**24**
Блок 1	Алгоритмы отслеживания (MOSSE, KCF), оптический поток	12	4	8
Блок 2	Поиск объектов	12	4	8
Блок 3	Распознавание лиц, Haar-каскады, нейросетевые эмбеддинги	12	4	8
	Мини проект (концепт итогового проекта)	**2**		
Тема 3	**Веб-разработка**	**22**	**4**	**18**
Блок 1	Клиент-серверная архитектура, протокол HTTP/HTTPS	12	2	10
Блок 2	Flask/FastAPI: маршруты, шаблоны, обработка форм	10	2	8
Тема 4	**API и сервисы. Базы данных**	**28**	**8**	**20**
Блок 1	RESTful архитектура, методы запросов, статус-коды	14	4	10
Блок 2	Форматы JSON/XML, аутентификация, Swagger/OpenAPI, SQL	14	4	10
	Итоговый проект	**2**		
	Резерв	**16**		
	Всего	136		

2. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН И СОДЕРЖАНИЕ 11 КЛАСС

[136 часов, 4 часа в неделю]

№	Наименование тем и блоков	Общее количество учебных часов	Теор.	Практ.
Тема 1	**Введение в формальные модели и системы**	**30**	**6**	**24**
Блок 1	Формальная логика, таблицы истинности, булевы уравнения	10	2	8
Блок 2	Теория множеств, комбинаторные конфигурации, вероятности	10	2	8
Блок 3	Графы, матрицы смежности, обходы (DFS/BFS), связность	10	2	8
Тема 2	**Алгоритмизация и основы программирования**	**36**	**12**	**24**
Блок 1	Асимптотический анализ, оценка сложности О-нотация	12	4	8
Блок 2	Стек, очередь, дерево, хеш-таблицы, реализация на Python	12	4	8
Блок 3	Рекурсия, динамическое программирование, жадные алгоритмы	12	4	8
	Мини проект (концепт итогового проекта)	**2**		
Тема 3	**Архитектура вычислительных систем и основы управления**	**26**	**4**	**22**
Блок 1	Системы счисления, кодирование информации, логические вентили	12	2	10
Блок 2	Организация памяти, процессорные циклы, параллелизм	14	2	12
Тема 4	**Решение комплексных задач.**	**24**	**8**	**16**
Блок 1	Стратегии разбора олимпиадных задач, эвристики, отладка	12	4	8
Блок 2	Оптимизация решений, анализ памяти и времени выполнения	12	4	8
	Итоговый проект	**2**		
	Резерв	**16**		
	Всего	136		

СОДЕРЖАНИЕ ПРОГРАММЫ 8 класс

Тема 1. Основы Python

Блок 1. Синтаксис, типы данных, операторы ввода/вывода.

Изучение базового синтаксиса, скалярных типов данных, арифметических и логических операций. Организация консольного ввода/вывода, форматирование строк.

Блок 2. Условные конструкции, циклы, итераторы.

Освоение ветвления (if/elif/else), циклических конструкций (for, while), операторов break/continue. Генераторы последовательностей для автоматизации повторяющихся операций.

Блок 3. Функции, области видимости, обработка исключений.

Декларация пользовательских функций, передача аргументов, возврат значений. Управление глобальными/локальными переменными. Обработка исключений (try/except/finally) для повышения надёжности кода.

Практическая работа №1-2: Настройка рабочего окружения (интерпретатор, IDE). Отработка операций с базовыми типами данных, арифметикой и форматированным выводом. Создание интерактивного консольного диалога с пользователем.

Практическая работа №3-4: Реализация алгоритмов ветвления с использованием сложных логических выражений. Написание и оптимизация итерационных алгоритмов с помощью циклов for и while, отладка вложенных конструкций.

Практическая работа №5-6: Декларация и вызов пользовательских функций, отработка работы с областями видимости и возвратом значений. Реализация безопасного ввода данных с обработкой потенциальных исключений. Финальная отладка модуля кода.

Тема 2. Структуры данных

Блок 1. Списки, кортежи, множества, словари.

Применение изменяемых и неизменяемых коллекций. Методы поиска, фильтрации, добавления и удаления элементов. Списковые включения (list comprehensions) для компактной генерации данных.

Блок 2. Строки, регулярные выражения, работа с файлами.

Операции со строками: конкатенация, срезы, методы форматирования. Применение регулярных выражений для парсинга текстов. Чтение и запись файлов различных форматов (txt, csv) с использованием контекстных менеджеров.

Блок 3. Введение в алгоритмы сортировки и поиска.

Изучение классических алгоритмов сортировки (пузырьковая, выбором, быстрая) и поиска (линейный, бинарный). Оценка их временной сложности и применимости на реальных массивах данных.

Практическая работа №1–2: Работа со списками, кортежами и множествами. Реализация задач на фильтрацию, объединение и пересечение коллекций, отработка синтаксиса списковых включений.

Практическая работа №3–4: Парсинг и сериализация данных в форматах txt/csv. Чтение, модификация и сохранение файлов с использованием контекстных менеджеров (with). Обработка исключений при работе с внешними источниками.

Практическая работа №5–6: Реализация алгоритмов сортировки (пузырьковая, выбором) и поиска (линейный, бинарный) на примере пользовательских массивов. Сравнение производительности на наборах данных разного размера, разработка мини проекта.

Защита мини проекта с презентацией

Тема 3. Введение в проектную деятельность

Блок 1. Жизненный цикл ПО, системы контроля версий Git.

Принципы версионирования, работа с репозиториями: init, add, commit, push. Ветвление (branch), слияние (merge) и разрешение конфликтов в Git.

Блок 2. Тестирование, отладка, документирование кода.

Написание модульных тестов (unittest/pytest), отладка в IDE с использованием точек останова. Оформление технической документации (docstrings, README.md).

Практическая работа №1-2: Инициализация локального Git-репозитория, создание структуры папок проекта, выполнение базовых операций версионирования (commit, log, status). Работа с удалённым репозиторием (GitHub): clone, push, pull.

Практическая работа №3-4: Написание модульных тестов для ранее созданных функций. Отладка кода с использованием точек останова и пошагового выполнения. Оформление документации к модулю и финальная публикация проекта в удалённый репозиторий.

Тема 4. Базы данных и основы веб-разработки

Блок 1. Основы HTML/CSS, статические сайты.

Вёрстка статических страниц: структура HTML-документа, базовые теги, стилизация через CSS. Основы клиентской интерактивности.

Блок 2. Реляционные СУБД, язык SQL.

Проектирование реляционных схем: таблицы, первичные/внешние ключи, связи. Написание SQL-запросов (SELECT, INSERT, UPDATE, DELETE, JOIN) для манипуляции данными

Практическая работа №1–2: Вёрстка многостраничного статического сайта (главная, о проекте, контакты) с использованием семантических тегов HTML5 и базовой стилизации CSS. Практическая работа.

Практическая работа №3–4: Проектирование простой реляционной схемы (например, «Ученики–Оценки»). Создание таблиц в SQLite, выполнение CRUD-операций через SQL-запросы.

Практическая работа №5: Добавление интерактивных элементов на сайт , динамическое изменение стилей.

Практическая работа №6: Разработка комплексного приложения, интегрирующего консольную логику, работу с БД и веб-интерфейс. Сборка, тестирование и публичная защита итогового проекта.

Защита итогового проекта с презентацией

СОДЕРЖАНИЕ ПРОГРАММЫ 9 класс

Тема 1. ООП в Python

Блок 1. Классы, объекты, конструкторы, инкапсуляция.

Создание классов, инициализация экземпляров, сокрытие внутренних состояний. Разграничение атрибутов класса и экземпляра.

Блок 2. Наследование, полиморфизм, абстрактные классы.

Построение иерархий наследования, переопределение методов, реализация полиморфных интерфейсов. Использование абстрактных базовых классов.

Блок 3. Магические методы, перегрузка операторов, dataclass.

Применение магических методов (__str__, __repr__, __add__), перегрузка операторов для пользовательских типов, оптимизация хранения данных через @dataclass.

Практическая работа №1-2: Проектирование классов, инициализация объектов, реализация конструкторов __init__. Моделирование предметных областей через создание экземпляров классов.

Практическая работа №3-4: Построение иерархий наследования, переопределение методов, отработка полиморфизма на примере обработки разнородных объектов в едином цикле.

Практическая работа №5-6: Перегрузка операторов через магические методы, использование @dataclass для декларативного описания структур данных.

Тема 2. Основы анализа данных

Блок 1. NumPy: массивы, матричные операции, оптимизация.

Векторизованные вычисления, многомерные массивы, матричная арифметика для ускорения обработки числовых данных.

Блок 2. Pandas: датафреймы, очистка, агрегация данных.

Загрузка неструктурированных данных (CSV/Excel), очистка от аномалий, группировка, сводные таблицы и базовая визуализация.

Блок 3. Matplotlib: Работа с графиками.

Изучение графических способов представления данных, создание различных типов графиков от простых диаграмм до трехмерных графиков.

Практическая работа №1–2: Создание и манипуляция массивами NumPy, выполнение векторизованных операций, сравнение производительности с циклами Python.

Практическая работа №3–4: Загрузка и очистка данных в Pandas, фильтрация, группировка, агрегация и экспорт результатов в файлы.

Практическая работа №5–6: Создание различных типов графиков и диаграмм. Разработка консольного аналитического минипроекта на основе реального датасета.

Защита мини проекта с презентацией

Тема 3. Основы машинного обучения

Блок 1. Scikit-learn: классификация, регрессия, метрики.

Обучение моделей классификации и регрессии, подбор гиперпараметров, оценка метрик качества (accuracy, precision, recall, MSE).

Блок 2. Кластеризация, подготовка датасетов, валидация.

Методы кластеризации (KMeans, DBSCAN), разбиение выборок на обучающую/тестовую, кросс-валидация и визуализация результатов.

Практическая работа №1-2: Подготовка датасета: нормализация, кодирование категориальных признаков, разбиение на выборки.

Практическая работа №3-4: Обучение и оценка моделей классификации/регрессии в Scikit-learn, подбор гиперпараметров через GridSearch.

Практическая работа №5–6: Применение методов кластеризации, визуализация кластеров, интерпретация результатов.

Тема 4. Разработка мини-приложений

Блок 1. Архитектура MVC, GUI-фреймворки (Tkinter/CustomTkinter).

Проектирование графического интерфейса, привязка событий, реализация логики взаимодействия с пользователем.

Блок 2. Упаковка приложений, работа с конфигурациями.

Сборка исполняемых файлов (pyinstaller), настройка внешних зависимостей и конфигурационных файлов (JSON/YAML).

Практическая работа №1–2: Создание графического интерфейса в Tkinter/CustomTkinter, привязка обработчиков событий, валидация ввода.

Практическая работа №3–4: Интеграция бизнес-логики с интерфейсом, сохранение/загрузка настроек, обработка ошибок пользователя.

Практическая работа №5: Проектирование и реализация полноценного мини-приложения с использованием ООП, анализа данных и графического интерфейса. Сборка, тестирование.

Защита итогового проекта с презентацией

СОДЕРЖАНИЕ ПРОГРАММЫ 10 класс

Тема 1. Анализ и обработка изображений. Техника безопасности

Блок 1. Трансформация изображения.

Изучение методов масштабирования, поворота, отражения и обрезки изображений. Практика изменения геометрии кадров с помощью библиотек Python.

Блок 2. Изменение значений компонентов цвета.

Работа с цветовыми моделями (RGB, HSV, LAB). Корректировка яркости, насыщенности, контрастности и перевод между цветовыми пространствами.

Блок 3. Применение фильтров.

Применение базовых фильтров для сглаживания, повышения резкости, выделения границ и морфологических операций. Знакомство с ядрами свёртки.

Практическая работа №1–2: Загрузка и предобработка изображений. Реализация геометрических трансформаций (масштабирование, поворот, кадрирование) в OpenCV.

Практическая работа №3–4: Работа с цветовыми каналами. Разделение, смешивание и конвертация цветовых пространств, применение масок по цвету.

Практическая работа №5–6: Наложение фильтров размытия, резкости и детекции границ (Canny, Sobel). Сравнение результатов обработки на тестовых наборах.

Тема 2. Компьютерное зрение: трекинг, распознавание лиц

Блок 1. Алгоритмы отслеживания (MOSSE, KCF), оптический поток.

Отслеживание перемещения объектов в видеопотоке, компенсация дрожания камеры, прогноз траекторий и работа с оптическим потоком (Lucas-Kanade).

Блок 2. Поиск объектов.

Изучение алгоритмов обнаружения объектов на изображении. Использование методов выделения контуров, шаблонов и каскадов Хаара.

Блок 3. Распознавание лиц, Haar-каскады, нейросетевые эмбеддинги.

Детекция лиц, извлечение признаков и сравнение эмбеддингов для идентификации. Этические аспекты применения биометрии.

Практическая работа №1–2: Захват видеопотока с камеры, реализация базового трекинга объектов, отрисовка траекторий в реальном времени.

Практическая работа №3–4: Детекция объектов по шаблонам и контурам, настройка пороговых значений для фильтрации ложных срабатываний.

Практическая работа №5–6: Распознавание лиц с использованием предобученных моделей, сравнение векторов признаков, сохранение результатов в лог.

Практическая работа №7: Разработка приложения компьютерного зрения, интегрирующего детекцию и трекинг.

Защита мини проекта с презентацией

Тема 3. Веб-разработка

Блок 1. Клиент-серверная архитектура, протокол HTTP/HTTPS.

Принципы сетевого взаимодействия, структура запросов и ответов, коды состояния, безопасность передачи данных и жизненный цикл HTTP-сессий.

Блок 2. Flask / FastAPI: маршруты, шаблоны, обработка форм.

Создание динамических серверных приложений, рендеринг шаблонов (Jinja2), валидация входящих данных и работа с пользовательскими формами.

Практическая работа №1–2: Настройка веб-сервера на Flask/FastAPI, создание базовых маршрутов, обработка GET/POST-запросов и возврат ответов.

Практическая работа №3–4: Рендеринг HTML-шаблонов, передача переменных из сервера в клиент, реализация форм с серверной валидацией.

Практическая работа №5: Интеграция статики (CSS/JS), настройка сессий и куки, развёртывание приложения на локальном сервере.

Тема 4. API и сервисы. Базы данных

Блок 1. RESTful архитектура, методы запросов, статус-коды.

Проектирование RESTful-эндпоинтов, стандартизация ответов, маршрутизация ресурсов и обработка ошибок клиента/сервера (4xx, 5xx).

Блок 2. Форматы JSON/XML, аутентификация, Swagger / OpenAPI, SQL.

Обмен данными в машиночитаемых форматах, реализация JWT-токенов, генерация интерактивной документации API и интеграция реляционных СУБД.

Практическая работа №1–2: Разработка REST API: CRUD-операции, возврат JSON-ответов, обработка исключений и логирование запросов.

Практическая работа №3–4: Реализация аутентификации через JWT, защита маршрутов, автогенерация документации через Swagger/OpenAPI.

Практическая работа №5: Подключение PostgreSQL/SQLite к API, миграции схем, выполнение сложных SQL-запросов через ORM.

Практическая работа №6: Проектирование и реализация веб-сервиса с полным циклом API, БД и клиентской частью.

Защита итогового проекта с презентацией

СОДЕРЖАНИЕ ПРОГРАММЫ 11 класс

Тема 1. Введение в формальные модели и системы

Блок 1. Формальная логика, таблицы истинности, булевы уравнения.

Логические операции, построение ДНФ/КНФ, решение систем булевых уравнений, упрощение выражений с помощью карт Карно.

Блок 2. Теория множеств, комбинаторные конфигурации, вероятности.

Операции над множествами, подсчёт перестановок/размещений/сочетаний, основы вероятностного моделирования и комбинаторного перебора.

Блок 3. Графы, матрицы смежности, обходы (DFS/BFS), связность.

Представление графов в памяти, обход компонент связности, поиск кратчайших путей, проверка на ацикличность и топологическая сортировка.

Практическая работа №1–2: Построение таблиц истинности, решение логических уравнений, программная реализация булевых функций на Python.

Практическая работа №3–4: Решение комбинаторных задач, генерация перестановок и сочетаний с помощью itertools, расчёт базовых вероятностей.

Практическая работа №5–6: Реализация обходов графов (DFS/BFS), поиск кратчайших путей (алгоритм Дейкстры), анализ связности на тестовых графах.

Тема 2. Алгоритмизация и основы программирования

Блок 1. Асимптотический анализ, оценка сложности О-нотация.

Анализ временной и пространственной сложности, выбор оптимального метода под ограничения задачи, оценка амортизированной сложности.

Блок 2. Стек, очередь, дерево, хеш-таблицы, реализация на Python.

Реализация абстрактных типов данных, операции вставки/удаления/поиска, сравнение эффективности структур и сценарии их применения.

Блок 3. Рекурсия, динамическое программирование, жадные алгоритмы.

Рекуррентные соотношения, мемоизация, разбиение сложных задач на подзадачи, применение жадных стратегий и доказательство их оптимальности.

Практическая работа №1–2: Оценка сложности алгоритмов, бенчмаркинг и сравнение производительности на тестовых данных разного размера.

Практическая работа №3–4: Реализация стека, очереди, бинарного дерева поиска и хеш-таблицы, отработка операций поиска и модификации.

Практическая работа №5–6: Решение задач на рекурсию и ДП (рюкзак, НОП, числа Фибоначчи), реализация жадных алгоритмов.

Практическая работа №7: Разработка алгоритмического решения олимпиадной задачи с анализом сложности и оптимизацией, создание мини проекта.

Защита мини проекта с презентацией

Тема 3. Архитектура вычислительных систем и основы управления

Блок 1. Системы счисления, кодирование информации, логические вентили.

Кодирование чисел, текста и графики, работа логических вентилей (И, ИЛИ, НЕ, XOR), построение полусумматоров и мультиплексоров.

Блок 2. Организация памяти, процессорные циклы, параллелизм.

Иерархия памяти (кэш/ОЗУ/ПЗУ), конвейеризация команд, принципы многозадачности, основы параллельных и распределённых вычислений.

Практическая работа №1–2: Перевод чисел между системами счисления, выполнение арифметических операций в двоичной и шестнадцатеричной системах.

Практическая работа №3–4: Моделирование логических схем (сумматоры, триггеры), анализ временных диаграмм и построение простых автоматов.

Практическая работа №5–6: Анализ влияния архитектуры на производительность, сравнение последовательных и многопоточных алгоритмов на Python.

Тема 4. Решение комплексных задач

Блок 1. Стратегии разбора олимпиадных задач, эвристики, отладка.

Декомпозиция нестандартных условий, применение эвристик, перебор с отсечениями, системная отладка и работа с edge-cases.

Блок 2. Оптимизация решений, анализ памяти и времени выполнения.

Профилирование кода, оптимизация констант и алгоритмов, сравнение альтернативных подходов по ресурсам и подготовка к профильным экзаменам.

Практическая работа №1–2: Разбор и решение задач повышенной сложности, применение эвристик, отсечений и бинарного поиска по ответу.

Практическая работа №3–4: Профилирование и оптимизация решений, сравнение подходов по времени и памяти, рефакторинг кода под лимиты.

Практическая работа №5: Разработка комплексного решения, интегрирующего алгоритмические, архитектурные и проектные компетенции.

Защита итогового проекта с презентацией

3. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Занятия проводятся в постоянных группах учащихся (оптимальное количество участников в группе: 8-15 человек), в форме теоретических, практических и индивидуальных занятий и консультаций (проектная деятельность, подготовка к олимпиадам, конференциям).

3.1 ТРЕБОВАНИЯ К ОБОРУДОВАНИЮ, ИНТЕРНЕТУ И ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ

Аппаратное обеспечение.

Минимальные требования (8–9 классы): Процессор Intel Core i5 / AMD Ryzen 5 (или эквивалент), 8 ГБ ОЗУ, SSD 256 ГБ, монитор Full HD, стабильное интернет-соединение от 10 Мбит/с.

Рекомендуемые требования (10–11 классы, ML/CV): Процессор Intel Core i7 / AMD Ryzen 7, 16–32 ГБ ОЗУ, SSD 512 ГБ+, видеокарта NVIDIA GTX 1660 / RTX 3060 и выше (для ускорения вычислений OpenCV/нейросетей), монитор Full HD/4K, интернет от 30 Мбит/с.

Программное обеспечение:

ОС: Windows 10/11 Pro, macOS 12+, или Linux (Ubuntu 22.04+).

Среда разработки: VS Code / PyCharm Community, Jupyter Notebook / Google Colab.

Язык и библиотеки: Python 3.10+, стандартный набор (venv, pip), библиотеки: numpy, pandas, matplotlib, scikit-learn, opencv-python, flask/fastapi, sqlalchemy, requests, pytest, git.

Сервисы и СУБД: Git/GitHub (или GitLab), DB Browser for SQLite, PostgreSQL (для старших классов), Postman/Insomnia для тестирования API.

Дополнительно: Браузер Chrome/Firefox с поддержкой WebRTC (для видеопотоков в CV), пакет Microsoft Office/LibreOffice для оформления документации.

4. СПИСОК ЛИТЕРАТУРЫ

1. Ахмедханлы, Д. М. Основы алгоритмизации и программирования [Текст] : учеб.метод. пособие / Д.М. Ахмедханлы, Н.В. Ушмаева. – Тольятти: Издв. ТГУ, 2016. –123 с.

2. Лейн, Д. Машинное обучение для детей. Практическое введение в искусственный интеллект [Текст] : учебное пособие / Д. Лейн ; пер. с англ. М. А. Федотенко. – Москва : Лаборатория знаний, 2023. – 288 с. – (Школа юного программиста). – ISBN 978-5-93208-646-9.

3. Окулов, С. М. Программирование в алгоритмах [Текст] : учебное пособие / С. М. Окулов. – 7-е изд., перераб. – Москва : Лаборатория знаний, 2021. – 386 с. – (Программирование). – ISBN 978-5-93208-527-4.

4. Златопольский, Д. М. Основы программирования на языке Python [Текст] : учебное пособие для школьников / Д. М. Златопольский. – Москва : Лаборатория

знаний, 2021. – 224 с. – (Программирование для школьников). – ISBN 978-5-93208-527-4.

5. Документация по библиотеке OpenCV-Python на русском языке [Электронный ресурс] : руководство разработчика. – Режим доступа: https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html (дата обращения: 13.05.2026).

6. Инфоурок. Программа дополнительного образования «Программирование»: методические материалы [Электронный ресурс]. – 2022. – Режим доступа: https://infourok.ru/programma-dopolnitelnogo-obrazovaniya-programmirovanie-6075510.html (дата обращения: 13.05.2026).

7. Воронцов, К. В. Введение в машинное обучение [Электронный ресурс] : учебный курс / К. В. Воронцов. – Москва : МФТИ, 2023. – Режим доступа: https://www.machinelearning.ru/wiki/index.php?title=Введение_в_машинное_обучение (дата обращения: 13.05.2026).

8. Методические рекомендации по разработке программ дополнительного образования технической направленности [Электронный ресурс] / ФГАОУ ДПО «Академия Минпросвещения России». – 2023. – Режим доступа: https://academy.minpros.ru/methodics (дата обращения: 13.05.2026).

9. Гусева, А. И. Учимся программировать на Python : практическое руководство для начинающих [Текст] / А. И. Гусева. – Санкт-Петербург : БХВ-Петербург, 2022. – 304 с. – (Программирование для школьников). – ISBN 978-5-9775-6821-3.

10. Тюкачев, Н. А. Алгоритмы и структуры данных : учебное пособие для СПО [Текст] / Н. А. Тюкачев, В. Г. Хлебостроев. – Санкт-Петербург : Лань, 2021. – 232 с. – ISBN 978-5-8114-6817-1.

11. Документация по языку программирования Python на русском языке [Электронный ресурс] : официальный перевод. – Режим доступа: https://pythondoc.ru/ (дата обращения: 13.05.2026).