

Департамент образования и науки Курганской области
Государственное автономное нетиповое образовательное учреждение
Курганской области
«Центр развития современных компетенций»

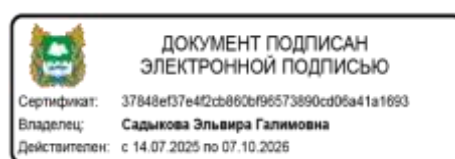
ПРИНЯТА

на заседании
педагогического совета
от «29» августа 2025 г.
Протокол № 1

УТВЕРЖДАЮ

И.о. директора ГАНУ КО «ЦРСК»

_____ Садыкова Э.Г.
Приказ от «29» августа 2025 г. №441



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА
технической направленности
«Программирование на Python»

Углубленный уровень
Возраст учащихся: 15–18 лет
Срок реализации: 9 месяцев

Автор-составитель:

Назимова Алена Юрьевна
педагог дополнительного
образования

г. Шадринск, 2025

Паспорт программы

Ф.И.О. автора/авторов	Назимова Алена Юрьевна
Учреждение	ГАНОУ КО «Центр развития современных компетенций» Детский технопарк «Кванториум» (г. Шадринск)
Квантум	IT-квантум
Тип программы	Дополнительная общеобразовательная (общеразвивающая) программа
Направленность программы	Техническая
Образовательная область	ИТ
Вид программы	
Продолжительность реализации программы	36 недель (9 месяцев)
Объем часов	144 часа
Линия освоения программы	проектная группа
Цель программы	формирование знаний и навыков алгоритмизации, изучение языка программирования Python, приобретение опыта полного цикла разработки программного обеспечения — от идеи до презентации готового продукта.
С какого года реализуется программа	2025

Содержание

1. Комплекс основных характеристик программы.....	4
1.1. Пояснительная записка	4
1.2. Цели и задачи программы. Планируемые результаты.....	5
1.3. Рабочая программа.....	7
2. Комплекс организационно-педагогических условий	14
2.1. Календарный учебный график	14
2.2. Формы текущего контроля / промежуточной аттестации	14
2.3. Материально-техническое обеспечение	14
2.4. Информационное обеспечение.....	14
2.5. Кадровое обеспечение	14
2.6. Методические материалы	15
2.7. Оценочные материалы	16
Список литературы	18

1. Комплекс основных характеристик программы

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа составлена с учетом следующих документов:

- Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказ Минпросвещения России от 27.07.2022 N 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Зарегистрировано в Минюсте России 26 сентября 2022 г. N 70226);
- Постановление Главного государственного санитарного врача РФ от 28 сентября 2020 г. N 28 "Об утверждении санитарных правил СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи"" (с изменениями и дополнениями);
- Письмо Минобрнауки России от 18.11.2015 N 09-3242 «О направлении информации» (вместе с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»);
- Профессиональный **стандарт** «Педагог дополнительного образования детей и взрослых», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 22 сентября 2021 г. N 652н (зарегистрирован Министерством юстиции Российской Федерации 17 декабря 2021 г., регистрационный N 66403)
- Методические рекомендации по разработке дополнительных общеобразовательных (общеразвивающих) программ в Курганской области (Департамент образования и науки Курганской области № 1661/9 от 21.07.2017 г.);
- устав, локальные акты и иные нормативные правовые документы ГАНОУ КО ЦРСК;
- положение о дополнительных общеобразовательных общеразвивающих программах ГАНОУ КО ЦРСК.

Направленность программы - техническая

Актуальность программы. Дополнительная общеобразовательная программа «Программирование на Python (проектная группа)» способствует приобщению учащихся к программированию, а также формированию целостного, системного представления об информационных технологиях в целом.

Обучение в IT-квантуме позволяет узнать о современных технологиях в сфере информационных технологий и программирования.

Образовательная программа «Программирование на Python (проектная группа)» создает благоприятные условия для развития инженерных способностей учащихся, расширяет и дополняет базовые знания по информационным технологиям, дает возможность удовлетворить интерес в программировании, проявить и реализовать свой творческий потенциал, что делает программу актуальной и востребованной.

Отличительные особенности программы. Данная программа ориентирует обучающихся на углубленное изучение алгоритмизации и программирования. Также учащиеся научатся разрабатывать собственные приложения на Python.

Адресат программы – обучающиеся 15-18 лет, интересующиеся программированием.

Срок реализации программы – 36 учебных недель.

Объем программы. Программа «Программирование на Python (проектная группа)» рассчитана на 9 месяцев обучения. Общий объем 144 академических часа.

Формы обучения, особенности организации образовательного процесса.

Образовательный процесс организуется в очной форме с применением электронного обучения. Численный состав группы – 15 человек, режим занятий – 2 учебных часа 2 раза в неделю. Продолжительность 1 учебного часа - 45 минут, перерыв между учебными занятиями – 10 минут (в соответствии с СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи» (Постановление Главного санитарного врача Российской Федерации от 28.09.2020 г. № 28)).

Возможность реализации индивидуального образовательного маршрута – организацией образовательного процесса предусматривается возможность реализации индивидуальных образовательных маршрутов как в части выбора тематики проектов и режима работы над проектами, так и в выборе сложности проекта и глубины освоения программных сред в зависимости от уровня подготовки обучающихся и их способностей.

Наличие детей-инвалидов и детей с ограниченными возможностями здоровья (ОВЗ) – программой предусмотрены возможности обучения детей инвалидов и лиц с ОВЗ, в том числе с использованием дистанционных образовательных технологий.

Наличие талантливых детей в объединении. Для талантливых и одаренных детей предполагается ИОМ по сопровождению проектной деятельности (обучающиеся выбирают тему проекта повышенной сложности, при выполнении проекта получают индивидуальные консультации педагога, параллельно осваивают содержание материала, не рассмотренное педагогом на занятиях). При демонстрации способностей освоения программы в сокращенные сроки и досрочной защиты готового проекта, обучающиеся прикрепляются наставниками в команды по выполнению проектов.

Уровни сложности содержания программы – повышенный, проектная группа.

1.2. Цели и задачи программы. Планируемые результаты

Целью программы является развитие компетенций в области алгоритмизации, программирования на языке Python.

Задачи программы:

- формирование профессиональных знаний и компетенций по программированию;
- формирование личностных и межличностных компетенций;

– развитие познавательной и творческой активности обучающихся посредством включения их в различные виды соревновательной и конкурсной деятельности.

Требования к результатам освоения программы модуля

Профессиональные и знаниевые компетенции (Hard Skills):

- знания основ алгоритмизации;
- навыки разработки программ на Python;
- понимание основ проектирования программных продуктов.

Личностные и межличностные компетенции (Soft Skills):

- работа в команде: работа в общем ритме, эффективное распределение задач и др.;
- развитие познавательных интересов учащихся,
- умение ориентироваться в информационном пространстве, продуктивно использовать техническую литературу для поиска сложных решений;
- навыки ведения проекта, проявление компетенции в вопросах, связанных с темой проекта, выбор наиболее эффективных решений задач в зависимости от конкретных условий;
- развитие критического мышления;
- проявление технического мышления, познавательной деятельности, творческой инициативы, самостоятельности;
- способность творчески решать технические задачи;
- готовность и способность применения теоретических знаний по физике, информатике для решения задач в реальном мире;
- способность правильно организовывать рабочее место и время для достижения поставленных целей.

Артефакты - не менее одного выполненного проекта.

1.3. Рабочая программа

Учебный план

№ п/п	Название раздела программы	Количество часов			Формы промежуточной аттестации
		всего	теория	практика	
1.	Введение в проектную деятельность	4	2	2	-
2.	Инструменты разработчика: Git и не только	6	4	2	-
3.	Работа с внешними сервисами (API)	10	4	6	-
4.	Хранение данных: Базы данных и ORM	12	6	6	-
5.	Проектирование и планирование проекта	8	4	4	-
6.	Разработка проекта	68	10	58	Код-ревью раз в 1-2 недели
7.	Тестирование, отладка и оптимизация	20	4	16	-
8.	Документирование и подготовка к защите	10	4	6	-
9.	Промежуточная аттестация	6	2	4	Презентация и защита проекта
	Итого	144	40	104	

Содержание

Тема 1. Введение в проектную деятельность. Количество часов: теория 2 ч, практика 2 ч.

Теория: Цели и задачи курса. Правила работы в кабинете и ТБ. Обзор возможных направлений для проектов: веб-приложения (Flask/Django), desktop (Tkinter/PyQT), игры (Pygame/Arcade), научные вычисления (numpy/pandas), боты (Telegram/Discord), парсеры. Критерии оценки проекта (сложность, чистота кода, функциональность, презентация). Постановка цели по SMART. Жизненный цикл проекта.

Практика: Мозговой штурм по генерации идей для проектов. Формирование предварительных целей каждым учащимся. Установка необходимого ПО (Git, редактор кода, дополнительные библиотеки).

Тема 2. Инструменты разработчика: Git и не только. Количество часов: теория 4 ч, практика 2 ч.

Теория: Системы контроля версий. Git. Основные команды: git init, git add, git commit, git push/pull, git branch, git merge. Структура репозитория: README.md,

LICENSE, .gitignore, requirements.txt. Семантическое версионирование (SemVer). Ведение CHANGELOG. Понятие Roadmap. Хостинги (GitHub, GitLab).

Практика: Создание аккаунта на GitHub. Создание репозитория для своего проекта. Настройка .gitignore и requirements.txt. Написание первого коммита. Клонирование чужого репозитория, его анализ.

Тема 3. Работа с внешними сервисами (API). Количество часов: теория 4 ч, практика 6 ч.

Теория: Понятие API. REST API. HTTP-методы (GET, POST, PUT, DELETE). Коды ответов сервера. Библиотеки requests и http.client. Сериализация и десериализация данных (JSON). Аутентификация и авторизация (API-ключи, OAuth 2.0). Переменные окружения для хранения секретов.

Практика: Написание скриптов для получения данных с публичных API (например, погода, курсы валют, данные YouTube/VK). Создание простого Telegram-бота с использованием python-telegram-bot или aiogram.

Тема 4. Хранение данных: Базы данных и ORM. Количество часов: теория 6 ч, практика 6 ч.

Теория: Реляционные базы данных. Основные понятия: таблицы, строки, столбцы, первичный и внешний ключи. Обзор SQL: основные операторы (SELECT, INSERT, UPDATE, DELETE, WHERE, JOIN). Введение в SQLite. Понятие ORM (Object-Relational Mapping). Библиотека SQLAlchemy (Core или ORM) или peewee.

Практика: Создание простой БД и написание SQL-запросов. Разработка консольного приложения (например, "Библиотека" или "Трекер задач") с использованием БД для хранения состояния.

Тема 5. Проектирование и планирование проекта. Количество часов: теория 4 ч, практика 4 ч.

Теория: Паттерны проектирования (на примере Фасада, Синглтона). Принципы SOLID (обзорно). UML-диаграммы: Use Case и диаграммы последовательностей. Методологии разработки: Waterfall vs Agile (Scrum/Kanban). Инструменты: Trello, Notion, диаграммы Ганта. Составление Технического Задания (ТЗ).

Практика: Создание UML-диаграммы проекта. Написание ТЗ. Декомпозиция проекта на задачи и этапы. Создание доски задач в Trello или аналоге. Выбор стека технологий.

Тема 6. Разработка проекта. Количество часов: теория 10 ч, практика 58 ч.

Теория: Мини-лекции по запросу группы. Темы определяются нуждами учащихся (напр.: "Настройка виртуального окружения", "Работа с графикой в Pygame", "Создание веб-форм на Flask", "Асинхронность в Python", "Работа со звуком")..

Практика: Непосредственная работа над кодом проекта. Сквозная деятельность: ведение Git-репозитория, документации. Регулярные код-ревью: учащиеся делятся на пары/микрогруппы и проводят взаимное ревью кода по чек-листу (читаемость, соблюдение PEP8, архитектура). Педагог выполняет роль ментора, проводя консультации и помогая решать сложные проблемы.

Тема 7. Тестирование, отладка и оптимизация. Количество часов: теория 4 ч, практика 16 ч.

Теория: Тестирование ПО. Юнит-тесты. Библиотека unittest или pytest. Модульные, интеграционные и системные тесты. Понятие "баг-репорта". Логирование с помощью библиотеки logging. Профилирование кода. Принципы оптимизации (алгоритмическая сложность, кэширование).

Практика: Написание набора юнит-тестов для ключевых функций своего проекта. Добавление логирования основных событий в приложении. Профилирование и попытка оптимизации одного из "узких" мест.

Тема 8. Документирование и подготовка к защите. Количество часов: теория 4 ч, практика 6 ч.

Теория: Важность документации. Виды документации (для пользователя, для разработчика). Стили написания комментариев (Docstring). Структура презентации проекта: проблема, решение, технологии, демо, выводы. Искусство публичных выступлений: работа с волнением, структура речи, взаимодействие с аудиторией.

Практика: Оформление README.md (описание, установка, использование). Создание презентации (5-7 слайдов). Подготовка скринкаста или видеозаписи с демонстрацией проекта. Репетиция выступления перед малой группой и получение обратной связи.

Тема 9. Промежуточная аттестация. Количество часов: теория 2 ч, практика 4ч.

Теория: Заключительный инструктаж по процедуре защиты. Критерии оценки.

Практика: Презентация и защита проекта, ответы на вопросы, демонстрация функционала. Коллективное обсуждение результатов курса.

Тематический план

№ п/п	Название раздела программы	Дата проведения занятия	Кол-во часов	Тема занятия	Форма занятия	Форма текущего контроля / промежуточной аттестации
1.	Вводное занятие		2	Цели и задачи курса. Правила работы в кабинете и ТБ.	Экскурсия	Беседа
2.	Инструменты разработчика: Git и не только		4	Системы контроля версий. Git. Основные команды: git init, git add, git commit, git push/pull, git branch, git merge. Структура репозитория: README.md, LICENSE, .gitignore, requirements.txt. Семантическое версионирование (SemVer). Ведение CHANGELOG. Понятие Roadmap. Хостинги (GitHub, GitLab).	Теоретическое занятие	Беседа
			2	Создание аккаунта на GitHub. Создание репозитория для своего проекта. Настройка .gitignore и requirements.txt. Написание первого коммита. Клонирование чужого репозитория, его анализ.	Практическое занятие	Наблюдение за работой
3.	Работа с внешними сервисами (API)		2	Понятие API. REST API. HTTP-методы (GET, POST, PUT, DELETE). Коды ответов сервера. Библиотеки requests и http.client. Сериализация и десериализация данных (JSON).	Теоретическое занятие	Беседа
			2	Написание скриптов для получения данных с публичных API (например, погода, курсы валют, данные YouTube/VK).	Практическое занятие	Наблюдение за практической работой
			2	Аутентификация и авторизация (API-ключи,	Теоретическое занятие	Беседа

				OAuth 2.0). Переменные окружения для хранения секретов.		
			4	Создание простого Telegram-бота с использованием python-telegram-bot или aiogram.	Практическое занятие	Наблюдение за практической работой
4	Хранение данных: Базы данных и ORM		4	Реляционные базы данных. Основные понятия: таблицы, строки, столбцы, первичный и внешний ключи. Обзор SQL: основные операторы (SELECT, INSERT, UPDATE, DELETE, WHERE, JOIN).	Теоретическое занятие	Беседа
			2	Создание простой БД и написание SQL-запросов.	Практическое занятие	Наблюдение за практической работой
			2	Введение в SQLite. Понятие ORM (Object-Relational Mapping). Библиотека SQLAlchemy (Core или ORM) или peewee.	Теоретическое занятие	Беседа
			4	Разработка консольного приложения (например, "Библиотека" или "Трекер задач") с использованием БД для хранения состояния.	Практическое занятие	Наблюдение за практической работой
5	Проектирование и планирование проекта		2	Паттерны проектирования (на примере Фасада, Синглтона). Принципы SOLID (обзорно). UML-диаграммы: Use Case и диаграммы последовательностей.	Теоретическое занятие	Беседа
			2	Создание UML-диаграммы проекта.	Практическое занятие	Наблюдение за практической работой
			2	Методологии разработки: Waterfall vs Agile (Scrum/Kanban). Инструменты: Trello, Notion,	Теоретическое занятие	Беседа

				диаграммы Ганта. Составление Технического Задания (ТЗ).		
			2	Написание ТЗ. Декомпозиция проекта на задачи и этапы. Создание доски задач в Trello или аналоге. Выбор стека технологий.	Практическое занятие	Наблюдение за практической работой
6.	Разработка проекта		68	Мини-лекции по запросу группы. Темы определяются нуждами учащихся. Непосредственная работа над кодом проекта. Сквозная деятельность: ведение Git-репозитория, документации. Регулярные код-ревью: учащиеся делятся на пары/микрогруппы и проводят взаимное ревью кода по чек-листу (читаемость, соблюдение PEP8, архитектура). Педагог выполняет роль ментора, проводя консультации и помогая решать сложные проблемы.	Практическое занятие	Наблюдение за работой, код-ревью раз в 1-2 недели.
7.	Тестирование, отладка и оптимизация		4	Тестирование ПО. Юнит-тесты. Библиотека unittest или pytest. Модульные, интеграционные и системные тесты. Понятие "баг-репорта". Логирование с помощью библиотеки logging. Профилирование кода. Принципы оптимизации (алгоритмическая сложность, кэширование).	Теоретическое занятие	Беседа
			16	Написание набора юнит-тестов для ключевых функций своего проекта. Добавление логирования основных событий в приложении. Профилирование и попытка оптимизации одного из "узких" мест.	Практическое занятие	Наблюдение за практической работой
8.	Документирование и подготовка к защите		2	Важность документации. Виды документации (для пользователя, для разработчика). Стили написания комментариев (Docstring).	Теоретическое занятие	Беседа

			2	Оформление README.md (описание, установка, использование).	Практическое занятие	Наблюдение за практической работой
			2	Структура презентации проекта: проблема, решение, технологии, демо, выводы. Искусство публичных выступлений: работа с волнением, структура речи, взаимодействие с аудиторией.	Теоретическое занятие	Беседа
			4	Создание презентации (5-7 слайдов). Подготовка скринкаста или видеозаписи с демонстрацией проекта. Репетиция выступления перед малой группой и получение обратной связи.	Практическое занятие	Наблюдение за практической работой
9.	Промежуточная аттестация		2	Заключительный инструктаж по процедуре защиты. Критерии оценки.	Теоретическое занятие	Беседа
			4	презентация и защита проекта, ответы на вопросы, демонстрация функционала. Коллективное обсуждение результатов курса.	Защита проекта	Выступление с презентацией

2. Комплекс организационно-педагогических условий

2.1. Календарный учебный график

Начало учебных занятий	1 сентября 2025 года
Продолжительность обучения (1 учебный год)	1 сентября 2025 года – 31 мая 2026 года (36 учебных недель)
Промежуточная аттестации	25 – 31 мая 2026 года
Каникулы	31 декабря 2025 года – 11 января 2026 года

2.2. Формы текущего контроля / промежуточной аттестации

Формы текущего контроля:

1. Беседа
2. Наблюдение за практической работой
3. Творческие проекты

При организации практических занятий и работе над проектами работа организуется малыми группами по 2-3 человека или индивидуально. Преобладающей формой текущего контроля выступает проверка хода решения кейса (выполнения проекта).

Форма промежуточной аттестации – выполнение проекта по заданной тематике.

2.3. Материально-техническое обеспечение

Компьютерное оборудование

Компьютеры с предустановленной операционной системой - 15 шт.

Мониторы - 15 шт.

Клавиатура USB - 15 шт.

Мышь USB - 15 шт.

Презентационное оборудование:

Моноблочное интерактивное устройство – 1 шт.

2.4. Информационное обеспечение

1. Профильные сайты
2. Техническая литература

2.5. Кадровое обеспечение

Требования к педагогу установлены Профессиональным стандартом «Педагог дополнительного образования детей и взрослых», утвержденного приказом Министерства труда и социальной защиты РФ от 22.09.2021 № 652н к образованию и обучению (направление подготовки, освоение программ профессиональной переподготовки и пр.):

Требования к образованию и обучению	Высшее образование или среднее профессиональное образование в рамках укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования «Образование и педагогические науки» или Высшее образование либо среднее профессиональное образование в рамках иных укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования при условии его соответствия дополнительным общеразвивающим программам, дополнительным предпрофессиональным программам, реализуемым организацией, осуществляющей образовательную деятельность, и получение при необходимости после трудоустройства дополнительного профессионального образования педагогической направленности или Успешное прохождение обучающимися промежуточной аттестации не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ
Требования к опыту практической работы	Не менее двух лет в должности педагога дополнительного образования, иной должности педагогического работника - для старшего педагога дополнительного образования.
Особые условия допуска к работе	Отсутствие ограничений на занятие педагогической деятельностью, установленных законодательством Российской Федерации Прохождение обязательных предварительных и периодических медицинских осмотров
Другие характеристики	При привлечении к работе с несовершеннолетними в качестве руководителей экскурсий с обучающимися - прохождение инструктажа по обеспечению безопасности жизнедеятельности

2.6. Методические материалы

В качестве методов обучения по программе используются наглядно-практический, исследовательский проблемный, проектные методы, кейс-методы.

На занятиях используются различные формы организации образовательного процесса:

- индивидуальная
- индивидуально-групповая
- групповая.

На занятиях используются различные педагогические технологии:

- технология проблемного обучения;
- технология проектной деятельности;
- технология портфолио.

Формы учебных занятий:

- экскурсия;
- теоретическое (формирования новых знаний)
- практическое занятие;
- Workshop (рабочая мастерская);
- занятие – соревнование.

2.7. Оценочные материалы

Форма проведения: защита проекта

Цель: комплексная оценка и подведение итогов работы обучающихся объединения ИТ-квантум по итогам обучения по образовательной программе «Программирование на Python (проектная группа)».

Задачи:

- определить уровень освоения обучающимися предметных знаний, уровень развития умений и навыков при решении практических задач в области программирования на Python;
- определить уровень сформированности компетенций проектной деятельности в сфере ИТ.

Задание:

Обучающимся предлагается выполнить проект на произвольную тематику по составленному ими ТЗ.

Примерные темы проектов:

- веб-приложение для проведения квизов с рейтингом участников;
- бот-ассистент для образовательного учреждения (расписание, ДЗ, напоминания);
- сервис рекомендаций книг/фильмов/игр на основе API;
- бот-агрегатор новостей по темам (использование API, RSS);
- анализатор соцсетей (частота постов, тональность текстов, пользовательские реакции);
- утилита-органайзер файлов (сортировка по расширениям, датам и пр);
- конвертер медиаформатов.

Требования к проектам:

Базовые (обязательные для всех):

1. Минимальный функционал MVP:
 - 3+ ключевые возможности, заявленные в ТЗ
 - Работоспособность без критических ошибок
2. Использование внешних данных: интеграция минимум с 1 внешним источником (API/БД/файлы)
3. Кодовая база:
 - Разделение на модули (не менее 3 файлов .py)
 - Соответствие PEP-8 (проверка flake8)
4. Документация:
 - README.md с инструкцией по установке и запуску

- Примеры использования (скриншоты/gif)

Дополнительные для групп (2–3 человека):

1. Распределение ролей в команде (back-end, data, UI)
2. Использование Git с историей коммитов (минимум 15 коммитов)
3. Юнит-тесты для ключевых функций (покрытие $\geq 50\%$)

Оснащение:

- Интерактивная доска.
- ПК.
- Доступ к сети Интернет.

Продолжительность презентации работ: 10 минут (5 минут презентация, 2 минуты демонстрация продукта, 3 минуты ответы на вопросы).

Критерии оценки работ

Критерий	Индивидуальный проект	Групповой проект
Реализация MVP		
Полнота функционала	0-16	0-16
Отсутствие критических ошибок	0-12	0-12
Работа с данными	0-12	0-12
Качество кода		
Архитектура/модульность	0-12	0-8
Читаемость/комментарии	0-8	0-8
Обработка ошибок	0-8	0-6
Тесты (для групп)	-	0-8
Документация		
README.md	0-8	0-6
Демо-материалы	0-8	0-6
Автодокументирование	0-6	0-8
Презентация		
Структура выступления	0-5	0-5
Ответы на вопросы	0-5	0-5
Максимальная оценка	100	100

Уровни освоения программы:

Низкий: 0-49 баллов.

Средний: 50-79 баллов.

Высокий: 80-100 баллов.

Список литературы

1. Агалаков, А. А. Программирование на языке Python. Базовый уровень : учебное пособие / А. А. Агалаков, К. И. Дементьева. – Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2024. – 117 с. – Текст : непосредственный.
2. Галимуллин, Н.Р. Формирование IT-навыков детей и подростков в дополнительном образовании : учебно-методическое пособие / Н. Р. Галимуллин. – Казань : ИП Мухамеева М.С., 2024. – 49 с. – Текст : непосредственный.
3. Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учебное пособие / С. Р. Гуриков. – Москва : Общество с ограниченной ответственностью «Научно-издательский центр ИНФРА-М», 2025. – 343 с. – Текст : непосредственный.
4. Документация Python 3 : официальный сайт. – Текст : электронный . – URL : <https://docs.python.org/3/> (дата обращения 13.06.2024)
5. Доусон, М. Программируем на Python : учебное пособие / М. Доусон. – Санкт Петербург : Питер, 2020. – 416с. – текст : непосредственный.
6. Калитвин, В. А. Введение в программирование на Python : учебное пособие / В. А. Калитвин ; Липецкий государственный педагогический университет им. П.П. Семенова-Тян-Шанского. – Липецк : Липецкий государственный педагогический университет им. П.П. Семенова-Тян-Шанского, 2023. – 85 с. – Текст : электронный . – URL : <https://biblioclub.ru/index.php?page=book&id=714538> (дата обращения: 13.06.2024)
7. Канева, О. Н. Введение в программирование на языке Python : учебное пособие / О. Н. Канева, Т. Ю. Финк. – Омск : Омский государственный технический университет, 2024. – 149 с. – Текст : непосредственный.
8. Лутц, М. Изучаем Python. 5-е издание в 2-х томах : практическое пособие / М. Лутц. – Москва : Диалектика, 2020. – текст : непосредственный.
9. Чернышев, С. А. Основы программирования на Python : учебное пособие / С.А. Чернышев. – 2-е изд., пер. и доп. – Москва : Издательство Юрайт, 2024. – 349 с. – (Высшее образование). – Текст : непосредственный.
10. Шведов, Е. Г. Основы языка Python : учебное пособие / Е. Г. Шведов, О. Ю. Соловьева. – Волгоград : Волгоградский государственный технический университет, 2024. – 80 с. – Текст : непосредственный.