

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
| Министерство Смоленской области по образованию и науке |
Управление образования и молодежной политики
Администрации города Смоленска|
муниципальное бюджетное общеобразовательное учреждение
«Средняя школа № 33" города Смоленска»

СОГЛАСОВАНО
Руководитель Кванториума

М.В. Богомолова

от «29» августа 2025 г.

ПРИНЯТО

Решение педагогического
совета МБОУ «СШ № 33»
Протокол № 1
от «29» августа 2025 г.

УТВЕРЖДЕНО

Директор
МБОУ «СШ № 33»



[Жойкин С.А.]

Приказ № 82-од
от «29» августа 2025 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА НАПРАВЛЕНИЯ
IT-КВАНТУМ

«Мастерская программирования: Python, C++, C#»

Возраст детей: 13 — 17 лет
Составитель:
Сапроненков Г.С.

СОДЕРЖАНИЕ

1. Пояснительная записка
2. Учебно-тематический план
3. Содержание программы
4. Воспитательный компонент
5. Организационно-педагогические условия реализации программы
6. Список литературы и иных источников

Пояснительная записка

Детский технопарк «Кванториум» призван обеспечить расширение содержания образования с целью развития у обучающихся современных компетенций и навыков, в том числе естественно-научной, математической, информационной грамотности, формирования критического и креативного мышления.

Детский технопарк «Кванториум» является частью образовательной среды общеобразовательной организации, на базе которой осуществляется дополнительное образование детей по программам естественно-научной и технической направленностей.

Нормативно-правовое обеспечение программы

Дополнительная общеобразовательная общеразвивающая программа «Мастерская программирования: Python, C++, C#» (далее — программа) разработана с учетом:

Федеральный Закон от 29.12.2012 № 273-ФЗ «Об образовании в РФ» (с изменениями от 25.12.2018г.).

Приказ Министерства образования и науки РФ от 09 ноября 2018 г. № 196 г. «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (с изменениями от 30.09.2020);

Приказ Министерства просвещения РФ от 03.09.2019 № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;

Распоряжение Правительства Российской Федерации от 04.09.2014 № 1726-р «Концепция развития дополнительного образования детей»

Федеральный закон от 29.12.2010 №436-ФЗ (ред.18.12.2018 г.) «О защите детей от информации, причиняющей вред их здоровью и развитию»

Приказ Минтруда и социальной защиты населения Российской Федерации от 5.05.2018 г. №298н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых».

Постановление Главного государственного санитарного врача РФ от 01.01.2021 № 628

«Об утверждении санитарных правил СП 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».

Письмо Минобрнауки России № 09-3242 от 18.11.2015 «О направлении информации» (вместе с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»

Приказ Министерства просвещения РФ от 3 сентября 2019 г. № 467 «Об утверждении Целевой модели развития систем дополнительного образования детей».

Направленность программы

Дополнительная общеобразовательная общеразвивающая программа «Мастерская программирования: Python, C++, C#» относится к программам технической направленности.

Цели и задачи образовательной программы

Цель - развить у участников навыки программирования на языках Python, C++ и C#, сформировать умение создавать собственные проекты и применять полученные знания для решения практических задач в различных областях информационных технологий.

Задачи программы:

Обучающие:

- познакомить учеников с основами синтаксиса и концепциями каждого из языков программирования (Python, C++, C#).
- обучить навыкам написания простых и сложных программ, алгоритмам и структурам данных.
- развивать умение проектировать и реализовывать собственные программные решения.
- формировать навыки работы с инструментами разработки и средами программирования.
- воспитывать интерес к дальнейшему обучению и развитию в области информационных технологий.
- создать условия для командной работы, обмена опытом и совместного решения задач.

Актуальность, новизна и значимость программы

Актуальность:

В современном мире информационных технологий программирование является ключевым навыком, востребованным в самых разных сферах — от разработки приложений и игр до искусственного интеллекта и робототехники. Знание языков Python, C++ и C# открывает широкие возможности для будущего профессионального роста и инновационных проектов. В условиях быстрого развития технологий обучение программированию становится необходимым для формирования конкурентоспособных специалистов.

Новизна:

Данная программа объединяет изучение трёх популярных и мощных языков программирования — Python, C++ и C# — в рамках одного курса, что позволяет участникам получить комплексное представление о современных инструментах разработки. Такой подход способствует более глубокому

пониманию принципов программирования, алгоритмов и структур данных, а также расширяет возможности для создания разнообразных проектов. Использование актуальных методов обучения и практических заданий делает программу современной и ориентированной на реальные задачи.

Значимость:

Программа способствует развитию технического мышления, креативности и навыков решения проблем у молодёжи, что важно для формирования будущих инженеров, разработчиков и IT-специалистов. Освоение нескольких языков программирования в рамках одного курса повышает гибкость участников на рынке труда и стимулирует их самостоятельное обучение новым технологиям. В целом, программа способствует развитию инновационного потенциала молодого поколения и укреплению позиций страны в сфере информационных технологий.

Отличительные особенности образовательной программы

Многоязычный подход:

Программа охватывает три популярных языка программирования — Python, C++ и C#, что позволяет участникам получить разносторонние знания и навыки, а также понять особенности и области применения каждого языка.

Практическая ориентация:

Большая часть обучения сосредоточена на создании реальных проектов, решении практических задач и разработке собственных программ, что способствует закреплению теоретических знаний через практику.

Интерактивное обучение:

Использование современных методов обучения, включая мастер-классы, групповые проекты, код-ревью и командную работу, делает процесс более увлекательным и эффективным.

Развитие критического мышления и креативности:

Участники учатся не только писать код, но и анализировать задачи, искать оптимальные решения и реализовывать инновационные идеи.

Гибкая программа:

Возможность адаптировать содержание курса под уровень участников — от начинающих до продвинутых — и учитывать их интересы в области технологий.

Поддержка наставников и экспертов:

Обучение проходит под руководством опытных преподавателей и специалистов в области программирования, что обеспечивает высокий уровень поддержки и обратной связи.

Подготовка к профессиональной деятельности:

Курс включает изучение современных инструментов разработки, работы с средами IDE, системами контроля версий и другими важными аспектами профессиональной среды программиста.

Категория обучающихся

Данная образовательная программа разработана для работы с обучающимися от 13 до 17 лет (6-11 классы). Программа не адаптирована для обучающихся с ОВЗ.

Условия и сроки реализации образовательной программы

Наполняемость группы не более 15 человек.

Форма обучения — очная, очно-заочная с использованием дистанционных технологий, ИКТ.

Режим занятий. При очной форме обучения: 2 раз в неделю по 2 академических часа (по 30-45 минут в зависимости от формы обучения и вида занятий) с 10-минутным перерывом. При использовании дистанционных технологий занятия по 2-3 часа (по 30 минут) в виде онлайн-конференции. При использовании очно-заочной формы обучения не менее трети объема аудиторных часов должно быть реализовано в очной форме, остальные - заочно и с применением дистанционных технологий.

Объем учебной нагрузки в год — 144 часа, в неделю — 2 часа.

Занятия проводятся в кабинете Робототехники, оборудованном согласно санитарно-эпидемиологическим требованиям.

Форма занятий - групповая.

Уровень освоения — стартовый.

Примерный календарный учебный график

График формируется после утверждения расписания.

Планируемые результаты обучения

Приобретение базовых знаний и навыков программирования:

Участники смогут писать простые и сложные программы на языках Python, C++ и C#, понимать их синтаксис и основные конструкции.

Освоение алгоритмических подходов:

Участники научатся разрабатывать алгоритмы, использовать структуры данных и решать типовые задачи, что повысит их логическое мышление и аналитические способности.

Практическое создание проектов:

Участники смогут самостоятельно реализовывать небольшие программные проекты, применять полученные знания для решения реальных задач.

Работа с инструментами разработки:

Освоят работу с современными средами разработки (IDE), системами контроля версий и отладчиками, что подготовит их к профессиональной деятельности.

Развитие навыков командной работы:

Научатся взаимодействовать в группе, делиться знаниями, совместно решать задачи и реализовывать проекты.

Формирование интереса к дальнейшему обучению:

Участники получают мотивацию продолжать изучение программирования и технологий разработки программного обеспечения.

Готовность к профессиональному развитию:

Полученные знания и навыки создадут основу для дальнейшего обучения в области информационных технологий, робототехники, искусственного интеллекта и других современных направлений.

Способы отслеживания результатов освоения программы учащимися:

- педагогическое наблюдение в ходе занятий;
- презентации в конце изучения темы.

**Учебно-тематический план программы:
«Мастерская программирования: Python, C++, C#»**

№	Раздел и темы	Количество часов			Форма контроля
		Теория	Практика	Всего	
1	Вводное занятие.	1	1	2	Наблюдение, беседа
2	Основы Python	1	1	2	Наблюдение, беседа
3	Условные конструкции в Python	1	1	2	Наблюдение, беседа
4	Циклы	1	1	2	Наблюдение, беседа
5	Функции	1	1	2	Наблюдение, беседа
6	• Оператор return и возвращение результата из функции	1	1	2	Наблюдение, беседа
7	Лямбда-выражения	1	1	2	Наблюдение, беседа
8	Преобразование типов и область видимости переменных	1	1	2	Наблюдение, беседа
9	Списки	1	1	2	Наблюдение, беседа
10	Словари	1	1	2	Наблюдение, беседа
11	Обработка ошибок и исключений	1	1	2	Наблюдение, беседа, Защита проекта
12	Объектно-ориентированное программирование	1	3	4	Наблюдение, беседа, Защита проекта
13	Основы языка программирования C++	1	1	2	Наблюдение, беседа
14	Циклы	1	1	2	Наблюдение, беседа
15	Ссылки	1	1	2	Наблюдение, беседа
16	Массивы	1	1	2	Наблюдение, беседа
17	Указатели	1	3	4	Наблюдение, беседа, защита проекта.
18	Функции C++ ч.1.	1	3	4	Наблюдение,

					беседа, защита проекта.
19	Функции C++ ч.2.	1	3	4	Защита проекта
20	Динамическая память и smart-указатели	1	1	2	Наблюдение, беседа
21	Объектно-ориентированное программирование C++	1	1	2	Наблюдение, беседа
22	Управление доступом. Инкапсуляция	1	1	2	Наблюдение, беседа
23	Конструктор копирования	1	1	2	Наблюдение, беседа
24	Ключевое слово this	1	1	2	Наблюдение, беседа
25	Статические члены класса	1	1	2	Наблюдение, беседа
26	Пользовательские типы данных	1	1	2	Наблюдение, беседа
27	Исключения	1	1	2	Наблюдение, беседа
28	Шаблоны	1	1	2	Наблюдение, беседа.
29	Контейнеры	1	1	2	Наблюдение, беседа
30	Работа над индивидуальным проектом с использованием фреймворка программирования приложений (API) Windows Forms C++.	8	30	38	Наблюдение, беседа, защита проекта.
31	Основы программирования на языке C#.	1	3	4	Наблюдение, беседа.
32	Классы, структуры и пространства имен	2	3	5	Наблюдение, беседа.
33	Объектно-ориентированное программирование C#	2	3	5	Наблюдение, беседа.
34	Обработка исключений C#	2	2	4	Наблюдение, беседа.
35	Делегаты, события и лямбды	2	3	5	Наблюдение, беседа.
36	Интерфейсы	2	3	5	Наблюдение,

					беседа.
37	Многопоточность	2	3	5	Наблюдение, беседа.
38	Работа над индивидуальным проектом с использованием фреймворка программирования приложений (API) Windows Forms C#.	3	3	6	Подготовка к защите проекта. Защита проекта.
	Итого:	54	90	144	

Содержание образовательной программы

Тема 1. Вводное занятие (2 часа).

Теория:

- знакомство с лабораторией Робототехники
- правила поведения в лаборатории;
- инструктаж по технике безопасности при работе с оборудованием, ручным и электрическим инструментом, а также правила пользования ПК;
- противопожарная безопасность.

Практика:

- опрос учащихся по технике безопасности, противопожарной безопасности, а также правилах поведения.

Тема 2. Основы Python (2 часа).

Теория:

- Регистрозависимость;
- Комментарии;
- Основные функции;
- Переменные и типы данных;
- Управляющие последовательности в строке;
- Динамическая типизация;
- Арифметические операции с числами;

Практика:

- работа в IDE. Выполнение тестового задания;
- написание первых программ.

Тема 3. Условные конструкции в Python (2 часа)

Теория:

- Операции сравнения;
- Логические операции;
- Оператор in;
- Условная конструкция if;
- Блок else;
- Блок elif;
- Вложенные конструкции if.

Практика:

- работа с исходным кодом языка Python;
- использование базовых команд ветвления языка Python.

Тема 4. Циклы (2 часа)

Теория:

- Цикл while;
- Цикл for;
- Вложенные циклы;

- Выход из цикла. break и continue.

Практика:

- выполнение тестового задания.

Тема 5. Функции (2 часа)

Теория:

- Локальные функции;
- Организация программы и функция main;
- Параметры функции;
- Значения по умолчанию функций;
- Передача значений параметрам по имени. Именованные параметры;
- Неопределенное количество параметров.

Практика:

- выполнение тестового задания.

Тема 6. Оператор return и возвращение результата из функции (2 часа)

Теория:

- Возвращение результата;
- Выход из функции;
- Функция как тип, параметр и результат другой функции.

Практика:

- выполнение тестового задания.

Тема 7. Лямбда-выражения (2 часа)

Теория:

- Анонимные функции;
- возвращение лямбда-выражений из функций;

Практика:

- выполнение тестового задания.

Тема 8. Преобразование типов и область видимости переменных (2 часа)

Теория:

- Неявные преобразования;
- Явные преобразования
- Глобальный контекст
- Локальный контекст
- Скрытие переменных
- Замыкания

Практика:

- выполнение тестового задания.

Тема 9. Списки (2 часа)

Теория:

- Создание списка;

- Обращение к элементам списка;
- Разложение списка;
- Перебор элементов;
- Сравнение списков;
- Получение части списка;
- Методы и функции по работе со списками;
- Добавление и удаление элементов;
- Проверка наличия элемента;
- Удаление с помощью del;
- Изменение подсписка;
- Подсчет вхождений;
- Сортировка;
- Фильтрация списка;
- Проекция списка;
- Минимальное и максимальное значения;
- Копирование списков;
- Соединение списков;
- Списки списков;
- Списки и алгоритмы.

Практика:

- выполнение тестового задания.

Тема 10. Словари (2 часа)

Теория:

- Преобразование списков и кортежей в словарь;
- Получение и изменение элементов;
- Удаление;
- Копирование и объединение словарей;
- Перебор словаря;
- Комплексные словари.

Практика:

- выполнение тестового задания.

Тема 11. Обработка ошибок и исключений (2 часа).

Теория:

- Конструкция try...except...finally;
- except и обработка разных типов исключений.

Практика:

- выполнение тестового задания;
- Подготовка собственного проекта – разработка консольного приложения.

Тема 12. Объектно-ориентированное программирование (4 часа).

Теория:

- Классы и объекты;

- Конструкторы;
- Атрибуты объекта;
- Методы классов;
- Деструкторы.

Практика:

- выполнение тестового задания;
- подготовка и защита собственного проекта – разработка консольного приложения.

Тема 13. Основы языка программирования C++ (2 часа)

Теория:

- Структура программы;
- Типы данных, переменные;
- using. Подключение пространств имен и определение псевдонимов;
- Арифметические операции;
- Статическая типизация и преобразования типов;
- Поразрядные операции;
- Условные выражения;
- Конструкция switch-case.

Практика:

- выполнение тестового задания.

Тема 14. Циклы (2 часа)

Теория:

- for;
- while;
- do...while;

Практика:

- выполнение тестового задания.

Тема 15. Ссылки (2 часа)

Теория:

- reference;
- ссылки в цикле for;

Практика:

- выполнение тестового задания.

Тема 16. Массивы (2 часа)

Теория:

- Массивы;
- Индексы. Получение и изменение элементов массива;
- Константные массивы;
- Длина массива;
- Перебор массивов;

- Перебор элементов в стиле for-each;
- Ввод значений массива с консоли;
- Многомерные массивы.

Практика:

- выполнение тестового задания.

Тема 17. Указатели (4 часа)

Теория:

- Определение указателя;
- Получение адреса и оператор &;
- Получение значения по адресу;
- Присвоение указателю другого указателя;
- Нулевые указатели;
- Ссылки на указатели;
- Адрес указателя.

Практика:

- выполнение тестового задания;
- подготовка к защите собственного проекта в рамках консольного приложения.

Тема 18. Функции C++ ч.1. (4 часа)

Теория:

- Определение функции;
- Область видимости объектов;
- Параметры функции;
- Передача аргументов по значению и по ссылке;
- Константные параметры;
- Оператор return и возвращение результата;
- Указатели в параметрах функции;
- Массивы в параметрах функции;
- Возвращение указателей и ссылок.

Практика:

- выполнение тестового задания;
- подготовка к защите собственного проекта в рамках консольного приложения.

Тема 19. Функции C++ ч.2. (3 часа)

Теория:

- Перегрузка функций;
- Рекурсивные функции;
- Рекурсия на примере быстрой сортировки;
- Указатели на функции;
- Указатели на функции как параметры;
- Тип функции;
- Указатель на функцию как возвращаемое значение.

Практика:

- выполнение тестового задания;
- подготовка и защита собственного проекта в рамках консольного приложения.

Тема 20. Динамическая память и smart-указатели (2 часа)

Теория:

- Динамические объекты;
- Динамические массивы;
- Smart-указатели;
- `unique_ptr<T>`;
- `shared_ptr<T>`.

Практика:

- выполнение тестового задания.

Тема 21. Объектно-ориентированное программирование C++ (2 часа)

Теория:

- Определение классов;
- Указатели на объекты классов;
- Конструкторы и инициализация объектов;
- Определение нескольких конструкторов;
- Параметры по умолчанию;
- Инициализация констант и списки инициализации.

Практика:

- выполнение тестового задания.

Тема 22. Управление доступом. Инкапсуляция (2 часа)

Теория:

- Опосредование доступа;
- Объявление и определение функций класса;

Практика:

- выполнение тестового задания.

Тема 23. Конструктор копирования (2 часа)

Теория:

- Создание конструктора копирования;
- Удаление конструктора копирования.

Практика:

- выполнение тестового задания.

Тема 24. Ключевое слово `this` (2 часа)

Теория:

- Ключевое слово `this`;

Практика:

- выполнение тестового задания.

Тема 25. Статические члены класса (2 часа)

Теория:

- Статические поля;
- Статические функции;
- Обращение к статическим членам класса;
- Статические константы.

Практика:

- выполнение тестового задания.

Тема 26. Пользовательские типы данных (2 часа)

Теория:

- Структуры;
- Перечисления.

Практика:

- выполнение тестового задания.

Тема 27. Исключения (2 часа)

Теория:

- Обработка исключений;
- Создание объекта исключения;
- Обработка и генерация разных типов исключений;
- try-catch и деструкторы;
- Вложенные try-catch;
- Создание своих типов исключений.

Практика:

- выполнение тестового задания.

Тема 28. Шаблоны (2 часа)

Теория:

- Шаблоны функций;
- Явный вызов реализации функции для определенного типа;
- Перегрузка функций и параметризация;
- Использование нескольких параметров;
- Выведение типов и decltype(auto);
- Нетипизированные параметры;
- Выведение типов и результата;
- Шаблон класса.

Практика:

- выполнение тестового задания.

Тема 29. Контейнеры (2 часа)

Теория:

- Вектор;
- Array;

- List;
- Forward_list;
- Deque;
- Стек std::stack;
- Множества.

Практика:

- выполнение тестового задания.

Тема 30. Работа над индивидуальным проектом с использованием фреймворка программирования приложений (API) Windows Forms C++. (38 часов)

Теория:

- Знакомство со средой разработки;

Практика:

- Реализация собственного проекта в формате графического приложения;
- Защита проекта.

Тема 31. Основы программирования на языке C#. (4 часа)

Теория:

- Структура программы;
- Переменные и константы;
- Литералы;
- Типы данных;
- Консольный ввод-вывод;
- Арифметические операции языка C#;
- Поразрядные операции;
- Операции присваивания;
- Преобразования базовых типов данных;
- Условные выражения;
- Конструкция if..else и тернарная операция;
- Циклы;
- Массивы;
- Методы;
- Передача параметров по ссылке и значению. Выходные параметры;
- Массив параметров и ключевое слово params;
- Рекурсивные функции;
- Локальные функции;
- Конструкция switch;
- Перечисления enum.

Практика:

- выполнение тестового задания.

Тема 32. Классы, структуры и пространства имен (4 часа).

Теория:

- Классы и объекты;

- Конструкторы, инициализаторы и деструкторы;
- Класс Program и метод Main. Программы верхнего уровня;
- Структуры;
- Типы значений и ссылочные типы;
- Область видимости (контекст) переменных и констант;
- Пространства имен;
- Глобальные пространства имен;
- Подключение пространств имен по умолчанию;
- Создание библиотеки классов в Visual Studio;
- Модификаторы доступа;
- Свойства;
- Перегрузка методов;
- Статические члены и модификатор static;
- Null и ссылочные типы;
- Псевдонимы типов и статический импорт.

Практика:

- выполнение тестового задания.

Тема 33. Объектно-ориентированное программирование C# (4 часа).

Теория:

- Наследование;
- Преобразование типов;
- Виртуальные методы и свойства;
- Скрытие методов и свойств;
- Различия переопределения и скрытия методов;
- Абстрактные классы и члены классов;
- Класс System.Object и его методы;
- Обобщения.

Практика:

- выполнение тестового задания.

Тема 34. Обработка исключений C# (2 часа).

Теория:

- Конструкция try..catch..finally;
- Блок catch и фильтры исключений;
- Типы исключений. Класс Exception;
- Генерация исключения и оператор throw;
- Создание классов исключений;
- Поиск блока catch при обработке исключений.

Практика:

- выполнение тестового задания.

Тема 35. Делегаты, события и лямбды (4 часа).

Теория:

- Делегаты;
- События;
- Лямбды;
- Замыкания.

Практика:

- выполнение тестового задания.

Тема 36. Интерфейсы (4 часа).

Теория:

- Определение интерфейсов;
- Применение интерфейсов;
- Явная реализация интерфейсов.

Практика:

- выполнение тестового задания.

Тема 36. Многопоточность (4 часа).

Теория:

- Введение в многопоточность. Класс Thread;
- Создание потоков. Делегат ThreadStart;
- Потоки с параметрами и ParameterizedThreadStart;
- Синхронизация потоков;
- Мониторы.

Практика:

- выполнение тестового задания.

Тема 37. Работа над индивидуальным проектом с использованием фреймворка программирования приложений (API) Windows Forms C#. (6 часов).

Теория:

- Знакомство со средой разработки с использованием иного языка программирования.

Практика:

- Реализация собственного проекта в формате графического приложения;
- Защита проекта.

Воспитательный компонент

Цель, задачи, целевые ориентиры воспитания детей

Целью воспитания является развитие личности, самоопределение и социализация детей на основе социокультурных, духовно-нравственных ценностей и принятых в российском обществе правил и норм поведения в интересах человека, семьи, общества и государства, формирование чувства патриотизма, гражданственности, уважения к памяти защитников Отечества и подвигам Героев Отечества, закону и правопорядку, человеку труда и старшему поколению, взаимного уважения, бережного отношения к культурному наследию и традициям многонационального народа Российской Федерации, природе и окружающей среде

(Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации», ст. 2, п. 2).

Задачами воспитания по программе являются:

- усвоение детьми знаний норм, духовно-нравственных ценностей, традиций; информирование
- детей, организация общения между ними на содержательной основе целевых ориентиров воспитания;
- формирование и развитие личностного отношения детей к художественно-эстетическим занятиям, к собственным нравственным позициям и этике поведения в учебном коллективе;
- приобретение детьми опыта поведения, общения, межличностных и социальных отношений в составе учебной группы, применение полученных знаний, организация активностей детей, их ответственного поведения, создание, поддержка и развитие среды воспитания детей, условий физической безопасности, комфорта, активностей и обстоятельств общения, социализации, признания, самореализации, творчества при освоении предметного и метапредметного содержания программы.

Целевые ориентиры воспитания детей по программе:

- интерес к технической деятельности, истории техники в России и мире, к достижениям российской и мировой технической мысли;
- понимание значения техники в жизни российского общества;
- навыков определения достоверности и этики технических идей;
- уважения к достижениям в технике своих земляков;
- воли, упорства, дисциплинированности в реализации проектов.

Формы и методы воспитания

Решение задач информирования детей, создания и поддержки воспитывающей среды общения и успешной деятельности, формирования межличностных отношений на основе российских традиционных духовных ценностей осуществляется на каждом из учебных занятий. Ключевой формой воспитания детей при реализации программы является организация их взаимодействий в подготовке и проведении календарных праздников с участием родителей (законных

представителей), организация, проведение и выступление на мероприятиях детского центра. В воспитательной деятельности с детьми по программе используются методы воспитания: метод убеждения (рассказ, разъяснение, внушение), метод положительного примера (педагога и других взрослых, детей); метод упражнений (приучения); методы одобрения и осуждения поведения детей, педагогического требования (с учётом преимущественного права на воспитание детей их родителей (законных представителей), индивидуальных и возрастных особенностей детей младшего возраста) и стимулирования, поощрения (индивидуального и публичного); метод переключения в деятельности; методы руководства и самовоспитания, развития самоконтроля и самооценки детей в воспитании; методы воспитания воздействием группы, в коллективе.

Условия воспитания, анализ результатов

Воспитательный процесс осуществляется в условиях организации деятельности обучающихся на основной учебной базе реализации программы в организации дополнительного образования детей в соответствии с нормами и правилами работы организации, а также на выездных базах, площадках, мероприятиях в других организациях с учётом установленных правил и норм деятельности на этих площадках.

Анализ результатов воспитания проводится в процессе педагогического наблюдения за поведением детей, их общением, отношениями детей друг с другом, в коллективе, их отношением к педагогам, к выполнению своих заданий по программе. Косвенная оценка результатов воспитания, достижения целевых ориентиров воспитания по программе проводится путём опросов родителей в процессе реализации программы (отзывы родителей, интервью с ними) и после её завершения (итоговые исследования результатов реализации программы за учебный период, учебный год). Анализ результатов воспитания по программе не предусматривает определение персонифицированного уровня воспитанности, развития качеств личности конкретного ребёнка, обучающегося, а получение общего представления о воспитательных результатах реализации программы, продвижения в достижении определённых в программе целевых ориентиров воспитания, влияния реализации программы на коллектив обучающихся: что удалось достичь, а что является предметом воспитательной работы в будущем. Результаты, полученные в ходе оценочных процедур — опросов, интервью используются только в виде агрегированных усреднённых и анонимных данных.

Организационно-педагогические условия реализации программы

Формы оценивания для отслеживания динамики освоения данной дополнительной общеобразовательной программы и анализа результатов образовательной деятельности разработан педагогический мониторинг. Мониторинг осуществляется в течение всего учебного года и включает первичную диагностику, а также промежуточную и итоговую аттестацию.

Вводный контроль (первичная диагностика) проводится в начале учебного года (сентябрь) для определения уровня подготовки обучающихся и впервые поступивших учеников. Форма проведения — беседа.

Текущий контроль осуществляется в процессе проведения каждого учебного занятия. Направлен на закрепление теоретического и практического материала по изучаемой теме. Форма проведения — практические работы или состязание.

Промежуточный контроль (промежуточная аттестация) проводится по итогам разделов. Форма проведения — практические работы или состязание.

Итоговая аттестация проводится в конце обучения для демонстрации достигнутого результата.

Контроль знаний проводится в форме тестовых заданий и творческих работ.

Итоговая оценка выставляется с учетом результатов итоговой аттестации, результативности участия в конкурсах разного уровня.

Критериями оценки освоения программы являются:

— соответствие уровня теоретических знаний, практических умений

обучающихся программным требованиям;

— самостоятельность работы;

— осмысленность действий;

— соответствие практической деятельности программным требованиям;

Программа предполагает выполнение обучающимися самостоятельных заданий, что позволит оценить уровень освоения материала и понимание структуры и функционирования изучаемых механизмов.

Критерии уровня освоения программного материала

Низкий уровень	Учебный материал усваивается бессистемно. Обучающийся овладел менее $\frac{1}{2}$ объема теоретических знаний и практических умений, навыков, предусмотренных программой. Работоспособность крайне низкая. Осваивает легкие задания. Есть недостатки в личностных качествах: обучающийся эмоционально неустойчив, проявляет недоверие к окружающим, боится общения. Часто наблюдаются негативные реакции на просьбы взрослых, капризы.
Средний уровень	Обучающийся овладел не менее $\frac{1}{2}$ объема теоретических знаний и практических умений, навыков, предусмотренных программой. Осваивает задания средней сложности. Личностные качества соответствуют «средним», «нормальным»: у ребёнка преобладает эмоционально-положительное настроение, приветлив с окружающими, проявляет активный интерес к словам и действиям сверстников и взрослых.
Высокий уровень	Обучающийся показывает высокий уровень знаний теоретического материала, овладел всеми умениями и навыками, предусмотренными программой. Осваивает задания повышенной трудности. Личностные характеристики соответствуют нормам поведения детей данного возраста: Обучающийся сохраняет жизнерадостное настроение, проявляет активность

Материально-техническое обеспечение программы Условия реализации программы

Учебно-методические:

- фото-видеоматериалы;
- технологические инструкции по сборке конструкций

Материально—технические:

№	Наименование	Количество
1	Столы	10
2	Стулья	15
3	Ноутбук/Персональный компьютер	15
4	Базовый робототехнический набор	15
5	Образовательный конструктор с комплектом датчиков	15

Список литературы

Список литературы для педагогов:

1. Брайан К. В. «Программирование для начинающих. Учебное пособие» / Пер. с англ. — М.: Вильямс, 2018.
2. Гудтейн А., Бенджамин Р. «Методика преподавания программирования» / Пер. с англ. — М.: Академический проект, 2020.
3. Иванов И.И., Петров П.П. «Методика обучения программированию в школе и колледже» — М.: Просвещение, 2019.
4. Мартин О. «Изучаем Python» / Пер. с англ. — М.: Диалектика, 2021.
5. Липпман Дж., Ладжой Б., Ладжой Л. «C++ Primer» (5-е издание) — М.: Питер, 2019.
6. Эндрю Троелсен, Филип Джепикс «C# 9 и .NET 5 — полное руководство» — М.: Вильямс, 2021.
7. Григорьев А.А., Смирнова Е.В. «Методика преподавания информационных технологий и программирования» — М.: Академия, 2018.
8. Иванова Е.В., Кузнецов В.П. «Практическое руководство по обучению программированию школьников и студентов» — СПб.: Питер, 2020.

Официальные документации языков:

Python: <https://docs.python.org/3/>

C++: <https://isocpp.org/std/the-standard>

C#: <https://docs.microsoft.com/en-us/dotnet/csharp/>

Список литературы для обучающихся:

1. Мартин О. «Изучаем Python» / Пер. с англ. — М.: Диалектика, 2021.
2. Бенджамин Р., Гудтейн А. «Python для начинающих» — М.: Вильямс, 2019.
3. <https://docs.python.org/3/> — Официальная документация по Python.
4. Липпман Дж., Ладжой Б., Ладжой Л. «C++ Primer» / Пер. с англ. — М.: Питер, 2019.
5. Страуструп Б. «Язык программирования C++» / Пер. с англ. — М.: Вильямс, 2014.
6. <https://isocpp.org/std/the-standard> — Официальная документация по C++.
7. Троелсен А., Джепикс Ф. «C# 9 и .NET 5 — полное руководство» / Пер. с англ. — М.: Вильямс, 2021.
8. Эндрю Троелсен, Филип Джепикс «C# 8 и .NET Core — практическое руководство» — М.: Вильямс, 2019.
9. <https://docs.microsoft.com/en-us/dotnet/csharp/> — Официальная документация по C#.