

Министерство образования и науки Хабаровского края
Краевое государственное автономное нетиповое
общеобразовательное учреждение
«Краевой центр образования»

РАССМОТРЕНО
на заседании
Педагогического совета
КГАНОУ «Краевой центр
образования»
Протокол №1
от «25» августа 2025 г.

УТВЕРЖДАЮ
Генеральный директор
КГАНОУ «Краевой центр
образования»

Черемухин П.С.
«25» августа 2025 г.



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ
ПРОГРАММА ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ

«Основы программирования на С#.»

Уровень освоения: стартовый
Возраст учащихся: 9 – 11 лет
Общий объем программы в часах: 144 часа

Составители программы:
Демешко К.М., ПДО
Валетова М.А., методист

г. Хабаровск,
2025 г.

Раздел 1. Комплекс основных характеристик ДООП

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на С#» имеет техническую направленность (ИТ – технологии), стартовый уровень.

Программа разработана с учетом следующих нормативно-правовых документов:

- Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации»;
- Распоряжение Правительства Российской Федерации от 31 марта 2022 года № 678-р «Концепция развития дополнительного образования детей до 2030 года»;
- Приказ Минпросвещения Российской Федерации от 27 июля 2022 года № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Постановление Главного государственного санитарного врача Российской Федерации от 28 сентября 2020 года № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организации воспитания и обучения, отдыха и оздоровления детей и молодежи» в редакции от 01.03.2025 г.;
- Постановление Главного государственного санитарного врача Российской Федерации от 28 января 2021 года № 2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания»;
- Распоряжение Министерства образования и науки Хабаровского края от 26 сентября 2019 года № 1321 «Об утверждении методических рекомендаций «Правила персонифицированного финансирования дополнительного образования детей в городском округе, муниципальном районе Хабаровского края»;
- Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (Письмо Министерства образования и науки Российской Федерации от 18 ноября 2015 года № 09-3242);
- Методические рекомендации по формированию механизмов обновления содержания, методов и технологий обучения в системе дополнительного образования детей, направленных на повышение качества дополнительного образования детей, в том числе включение компонентов, обеспечивающих формирование функциональной грамотности и компетентностей, связанных с эмоциональным, физическим, интеллектуальным, духовным развитием человека, значимых для вхождения Российской Федерации в число десяти ведущих стран мира по качеству общего образования, для реализации приоритетных направлений научно-

технологического и культурного развития страны (Письмо Минпросвещения Российской Федерации от 29 сентября 2023 года № АБ-3935/06);

– Приказ краевого государственного автономного образовательного учреждения дополнительного образования «Центр развития творчества детей (региональный модельный центр дополнительного образования детей Хабаровского края)» от 27 мая 2025 года №220 П «Об утверждении Положения о дополнительной общеобразовательной общеразвивающей программе, реализуемой в Хабаровском крае»);

– Устав краевого государственного автономного нетипового образовательного учреждения «Краевой центр образования».

Актуальность программы

Развитие современного общества предполагает внедрение информационных технологий во все сферы жизни, именно поэтому возникает необходимость в более раннем обучении возрастным основам программирования и функционирования компьютерных программ для более глубокого изучения программирования и работе в IT сфере. Язык C# является одним из наиболее быстро растущих, востребованных и при этом «удобных» языков программирования. Язык входит в семью C-подобных языков. Язык C# практически универсален. Можно использовать его для создания любого ПО: продвинутых бизнес-приложений, видеоигр, функциональных веб-приложений.

Педагогическая целесообразность

Программа построена на основе системно-деятельностного и личностно-ориентированного подходов, что способствует формированию у обучающихся ключевых компетенций, необходимых для жизни в современном обществе. Активно используются наглядные, практические методы обучения, проектная деятельность и элементы геймификации для повышения мотивации и эффективности усвоения материала.

Отличительные особенности и новизна программы

Предлагаемый курс представляет собой программу, на которой обучающийся шаг за шагом узнает об основных понятиях языка C#, включая переменные, инструкции управления, функции, типы и массивы данных, классы и объекты.

В процессе знакомства с языком, обучающийся не только узнает теорию, но и приобретает навыки самостоятельной индивидуальной и групповой работы по практическому программированию. В результате, обучающийся может попробовать себя в роли программиста-разработчика простейших проектов и понять, насколько актуально для него дальнейшее развитие в сфере программирования.

Данный курс, являясь вводным, включает в себя разработку и защиту небольшого проекта, выполнение которого завершается его представлением и защитой. Таким образом, в процессе обучения будут развиваться исследовательские и проектные компетенции обучающихся.

Адресат программы: Программа ориентирована на обучающихся возрастной категории от 9 до 11 лет. Состав группы постоянный. Наполняемость в группах: до 15 человек.

Форма обучения: очная.

Объём реализации программы:

Период	Продолжительность занятия	Кол-во занятий в неделю	Кол-во часов в неделю	Кол-во недель	Кол-во часов в год
1 год	2 часа	2	4 ч	36	144 ч
Итого по программе:					144 ч

Режим организации занятий: Занятия в объединении рекомендуется проводить по 2 часа 2 раза в неделю. Продолжительности занятия - 45 минут. После 45 минут занятий организовывается перерыв длительностью 10 минут для проветривания помещения и отдыха обучающихся

1.2. Цель и задачи программы

Цель программы: Развитие познавательной активности, интеллектуальных и творческих способностей учащихся, в процессе обучения программированию на языке C#.

Задачи программы:

- сформировать навыки алгоритмического и логического мышления, грамотной разработки эффективных алгоритмов и программ;
- познакомить с принципами и методами структурного программирования;
- сформировать навыки работы в системе программирования C#;
- научить работать с основными структурами данных и типовыми методами обработки этих структур;
- сформировать навыки чтения чужого кода;
- развитие терпения, самоконтроля;
- развитие логического мышления, самостоятельности, познавательного потенциала обучающегося.
- воспитывать упорство в достижении результата;
- стимулировать интерес к изучению профессии, связанной с программированием.

1.3. Учебный план

№ п/п	Название разделов, тем	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1	Вводное занятие. ТБ. Знакомство с программой.	2	2	-	Опрос, беседа
2	История языков программирования. Введение в язык программирования C#. Среда программирования C#.	2	1	1	Опрос, беседа, практическая работа

3	Типы данных и вывод. Определение переменной	2	1	1	Опрос, беседа, практическая работа
4	Арифметические выражения и операции	2	1	1	Опрос, беседа, практическая работа
5	Ввод данных и проверка ввода	2	1	1	Опрос, беседа, практическая работа
6	Примеры решения задач.	2	-	2	Практическая работа
7	Отработка навыков решения простейших задач.	2	-	2	Практическая работа
8	Строки: базовые операции	2	1	1	Опрос, беседа, практическая работа
9	Логические выражения и булевы операции.	2	1	1	Опрос, беседа, практическая работа
10- 11	Условные операторы (if/else, else if).	4	2	2	Опрос, беседа, практическая работа
12- 13	Примеры решения задач.	4	-	4	Практическая работа
14- 15	Отработка навыков решения простейших задач.	4	-	4	Практическая работа
16- 17	Цикл for и практические задачи	4	2	2	Опрос, беседа, практическая работа
18- 19	Цикл while и do-while Примеры решения задач.	4	2	2	Опрос, беседа, практическая работа
20- 21	Отработка навыков решения простейших задач.	4	-	4	Практическая работа
22	Операторы break и continue. Накопители (аккумуляторы)	2	1	1	Опрос, беседа, практическая работа
23	Отработка навыков решения простейших задач.	2	-	2	Практическая работа
24- 25	Массивы (одномерные)	4	1	3	Опрос, беседа, практическая работа
26- 27	Отработка навыков решения простейших задач	4	-	4	Практическая работа
28	Списки (List<T>) и работа с коллекциями	2	1	1	Опрос, беседа, практическая работа
29- 30	Отработка навыков решения простейших задач	4	-	4	Практическая работа

31	Функции (методы): объявление и вызов	2	1	1	Опрос, беседа, практическая работа
32	Отработка навыков решения простейших задач	2	-	2	Практическая работа
33	Область видимости и параметры (ref, out)	2	1	1	Опрос, беседа, практическая работа
34	Примеры решения задач.	2	-	2	Практическая работа
35	Рекурсия (введение)	2	1	1	Опрос, беседа, практическая работа
36	Примеры решения задач.	2	-	2	Практическая работа
37- 38	Отработка навыков решения задач.	4	-	4	Практическая работа
39	Строки: Substring, IndexOf, Replace, Split/Join	2	1	1	Опрос, беседа, практическая работа
40	Примеры решения задач текстовых задач	2	-	2	Практическая работ
41	Форматирование вывода и работа с вещественными числами	2	1	1	Опрос, беседа, практическая работа
42	Примеры решения задач.	2	-	2	Практическая работа
43	Ввод/вывод файлов (простая работа с файлами)	2	1	1	Опрос, беседа, практическая работа
44	Примеры решения задач	2	1	1	Опрос, беседа, практическая работа
45	Исключения и обработка ошибок (try/catch)	2	1	1	Опрос, беседа, практическая работа
46	Основы объектно- ориентированного программирования: классы и объекты	2	1	1	Опрос, беседа, практическая работа
47	Примеры решения задач	2	-	2	Практическая работа
48	Инкапсуляция и свойства (get/set)	2	1	1	Опрос, беседа, практическая работа
49	Примеры решения задач	2	-	2	Практическая работа
50	Наследование и полиморфизм (введение)	2	1	1	Опрос, беседа, практическая работа
51	Примеры решения задач	2	-	2	Практическая работа
52	Интерфейсы и их использование	2	1	1	Опрос, беседа, практическая работа
53	Создание интерфейса IPlayable для игровых объектов	2	-	2	Практическая работа
54	LINQ: основы и простые запросы	2	1	1	Опрос, беседа, практическая работа
55	Решение задач	2	-	2	Практическая работа
56	События и делегаты (простое знакомство)	2	1	1	Опрос, беседа, практическая работа

57	Простая имитация событий внутри программы (подписка/вызов)	2	-	2	Практическая работа
58	Итераторы и yield (введение)	2	1	1	Опрос, беседа, практическая работа
59-60	Решение задач	4	-	4	Практическая работа
61	Игровой мини-проект: текстовая игра (часть 1)	2	1	1	Опрос, беседа, практическая работа
62	Игровой мини-проект: текстовая игра (часть 2)	2	-	2	Практическая работа
63	Графическое введение (опционально): простая форма WinForms или Unity intro	2	1	1	Опрос, беседа, практическая работа
64	Алгоритмы: сортировка, поиск (вводный уровень)	2	1	1	Опрос, беседа, практическая работа
65	Модули и проектная структура. Документирование кода	2	1	1	Опрос, беседа, практическая работа
66	Безопасность и этика программирования, ресурсы для продолжения обучения	2	1	1	Опрос, беседа, практическая работа
67	Подготовка итогового проекта	2	1	1	Опрос, беседа, практическая работа
68-71	Реализация итоговых проектов	8	-	8	Практическая работа
72	Итоговое занятие. Рефлексия и планы на будущее	2	1	1	Итоговая аттестация
Итого:		144	39	105	

1.4. Содержание учебного плана

Тема 1. Вводное занятие. ТБ. Знакомство с программой.

Теория: цель курса, правила работы, безопасность в сети и за компьютером; что такое программы и алгоритмы.

Тема 2. История языков программирования. Введение в язык программирования C#. Среда программирования C#.

Теория: История языков программирования и место C#. Краткая хронология: от машинного кода → ассемблер → языки высокого уровня (Fortran, COBOL, C). Почему появились новые языки: удобство, безопасность, переносимость, новые платформы. Области применения C#

Практика: Установка IDE, создание первого проекта «Hello, World!», запуск и отладка

Тема 3. Типы данных и вывод. Определение переменной.

Теория: Срезы строк. Использование срезов. Метод IndexOf. Функции. Использование функций. Возврат значений. Локальные и глобальные переменные. Рекурсия. Использование рекурсии. Кортежи. Функция RANGE, цикл FOR. Списки. Метод Split и Join.

Практика: Примеры решения задач. Отработка навыков решения простейших задач. Решение несложных олимпиадных задач. Отработка навыков решения задач.

Тема 4. Арифметические выражения и операции.

Теория: +, -, *, / приоритет операций, преобразование типов.

Практика: Задачи: вычисления формул, задачи на деление/остаток.

Тема 5. Ввод данных и проверка ввода.

Теория: Console.ReadLine, int.Parse / int.TryParse, обработка ошибок ввода.

Практика: Простые вычисления, вывод форматированных строк.

Тема 6. Примеры решения задач.

Практика: Решение задач.

Тема 7. Отработка навыков решения простейших задач.

Практика: Решение задач.

Тема 8. Строки: базовые операции.

Теория: Конкатенация, интерполяция, Length, индексация.

Практика: Простые программы на обработку строк.

Тема 9. Логические выражения и булевы операции.

Теория: Операторы сравнения и логические &&, ||, !.

Практика: Примеры условий, валидация, булевы выражения.

Тема 10-11. Условные операторы (if/else, else if).

Теория: Построение ветвлений, тернарный оператор.

Практика: Задачи на ветвления (калькулятор оценок, проверка пароля).

Тема 12-13. Примеры решения задач.

Практика: Решение задач.

Тема 14-15. Отработка навыков решения простейших задач.

Практика: Решение задач.

Тема 16-17. Цикл for и практические задачи.

Теория: Синтаксис for, счетчики, диапазоны.

Практика: Вывод последовательностей, сумма чисел, таблица умножения

Тема 18-19. Цикл while и do-while.

Теория: Отличия от for, бесконечные циклы и безопасность.

Практика: Программы с повторным вводом до корректного значения, меню.

Тема 20-21. Отработка навыков решения простейших задач.

Практика: Решение задач.

Тема 22. Операторы break и continue. Накопители (аккумуляторы).

Теория: Управление потоком в циклах, подсчёты.

Практика: Задачи на фильтрацию и прерывание цикла

Тема 23. Отработка навыков решения простейших задач.

Практика: Решение задач.

Тема 24 - 25. Массивы (одномерные).

Теория: Объявление, инициализация, индексация, поиск, сортировка простым способом.

Практика: Задачи: поиск максимума/минимума, подсчёт элементов.

Тема 26-27. Отработка навыков решения простейших задач.

Практика: Решение задач.

Тема 28. Списки (List<T>) и работа с коллекциями.

Теория: отличие списка от массива, методы Add/Remove/Count.

Практика: обработка динамических данных, простые приложения (список дел).

Тема 29-30. Отработка навыков решения простейших задач.

Практика: Решение задач.

Тема 31. Функции (методы): объявление и вызов.

Теория: Зачем использовать методы, параметры, возвращаемые значения, static.

Практика: Рефакторинг задач в методы, написание полезных функций.

Тема 32. Отработка навыков решения простейших задач.

Практика: Решение задач.

Тема 33. Область видимости и параметры (ref, out).

Теория: Локальные и глобальные переменные, передача по значению и по ссылке.

Практика: Примеры с ref/out, исправление ошибок области видимости.

Тема 34. Примеры решения задач.

Практика: Решение задач.

Тема 35. Рекурсия (введение).

Теория: Понятие рекурсии, базовый случай, стек вызовов.

Практика: Факториал, простая рекурсивная сумма, объяснение на примерах.

Тема 36. Примеры решения задач.

Практика: Решение задач.

Тема 37-38. Отработка навыков решения простейших задач.

Практика: Решение задач.

Тема 39. Строки: Substring, IndexOf, Replace, Split/Join.

Теория: Расширенные методы работы со строками.

Практика: разбор простых парсеров, обработка текстовых задач.

Тема 40. Примеры решения задач.

Практика: Решение задач.

Тема 41. Форматирование вывода и работа с вещественными числами.

Теория: Тип double, округление (Math.Round), форматирование чисел.

Практика: Задачи на расчёты с вещественными числами и аккуратный вывод.

Тема 42. Примеры решения задач.

Практика: Решение задач.

Тема 43. Ввод/вывод файлов (простая работа с файлами).

Теория: Чтение/запись текстовых файлов (File.ReadAllText / WriteAllText / Append), безопасность.

Практика: Хранение данных (список игроков), чтение настроек.

Тема 44. Примеры решения задач.

Практика: Решение задач.

Тема 45. Исключения и обработка ошибок (try/catch).

Теория: Зачем нужны исключения, базовый синтаксис, finally.

Практика: Защита кода от невалидного ввода, логирование ошибок в файл.

Тема 46. Основы объектно-ориентированного программирования: классы и объекты.

Теория: Класс, объект, поля, методы, конструктор.

Практика: Создание класса "Ученик" или "Игрушка" с полями и методами.

Тема 47. Примеры решения задач.

Практика: Решение задач.

Тема 48. Инкапсуляция и свойства (get/set).

Теория: Закрытые поля, свойства, модификаторы доступа.

Практика: Реализация свойств с валидацией (например, возраст в диапазоне).

Тема 49. Примеры решения задач.

Практика: Решение задач.

Тема 50. Наследование и полиморфизм (введение).

Теория: Базовый класс и производный, override, виртуальные методы.

Практика: Пример: класс «Животное» → «Птица»/ «Рыба» с методом Speak/Move.

Тема 51. Примеры решения задач.

Практика: Решение задач.

Тема 52. Интерфейсы и их использование.

Теория: Что такое интерфейс, отличие от наследования, примеры.

Практика: Создание интерфейса IPlayable для игровых объектов.

Тема 53. Создание интерфейса IPlayable для игровых объектов.

Практика: Создание интерфейса.

Тема 54. LINQ: основы и простые запросы.

Теория: Идея LINQ, Where, Select, OrderBy на уровне школьника.

Практика: Фильтрация и сортировка коллекций с LINQ в задачах.

Тема 55. Решения задач.

Практика: Решение задач.

Тема 56. События и делегаты (простое знакомство).

Теория: Принцип событий, примеры (кнопка нажата), делегаты как указатели на метод.

Практика: Простая имитация событий внутри программы (подписка/вызов).

Тема 57. Простая имитация событий внутри программы (подписка/вызов).

Практика: Решение задач.

Тема 58. Итераторы и yield (введение).

Теория: IEnumerable<T>, ленивые последовательности, yield return.

Практика: Генерация последовательностей (например, генератор чисел Фибоначчи).

Тема 59-60. Решения задач.

Практика: Решение задач.

Тема 61. Игровой мини-проект: текстовая игра (часть 1).

Теория: планирование проекта, разбиение на задачи, роль структур данных.

Практика: начало проекта: структуры данных, ввод/вывод, базовая логика.

Тема 62 Игровой мини-проект: текстовая игра (часть 2).

Практика: Расширение логики, сохранение/загрузка прогресса, тестирование.

Тема 63. Графическое введение (опционально): простая форма WinForms или Unity intro.

Теория: GUI vs консоль, основы событийного интерфейса.

Практика: создание простой формы с кнопками или подключение к Unity (если доступно) для показа визуального результата.

Тема 64. Алгоритмы: сортировка, поиск (вводный уровень).

Теория: Линейный поиск, бинарный поиск (концепт), простая сортировка пузырьком.

Практика: Реализация и измерение времени выполнения на простых данных.

Тема 65. Модули и проектная структура. Документирование кода.

Теория: Организация проектов, разбиение на классы, комментарии, XML-комментарии.

Практика: Рефакторинг кода проекта, написание простой документации.

Тема 66. Безопасность и этика программирования, ресурсы для продолжения обучения.

Теория: Безопасное использование интернета, авторское право, честность в задачах.

Практика: Создание списка ресурсов, план дальнейшего обучения.

Тема 67. Подготовка итогового проекта.

Практика: Команды/индивидуальные проекты, постановка задач, распределение работы.

Тема 68-71. Реализация итоговых проектов.

Практика: Команды/индивидуальные проекты, постановка задач, распределение работы.

Тема 72. Итоговое занятие. Рефлексия и планы на будущее.

Теория: Повторение ключевых тем.

Практика: Мини-тест, разбор ошибок.

Планируемые результаты

Предметные:

- будут знать понятия «алгоритм», «программа», основные конструкции языка программирования C# (ветвления if, операторы цикла while, for, вспомогательные алгоритмы, функции, модули);
- будут знать структуру данных языка программирования C#;
- будут владеть основными приёмами составления программ на языке программирования C#
- смогут отлаживать программный код средствами среды разработки.

Метапредметные:

- развитие терпения, самоконтроля;
- развитие логического мышления, самостоятельности, познавательного потенциала обучающегося;
- смогут самостоятельно разрабатывать алгоритмы достижения целей (решения задач, выполнения проектов и т.д.).

Личностные:

- формирование навыков анализа и самоанализа;
- приобретение начальной профессиональной подготовки по данному направлению, что способствует повышению социальной адаптации учащихся после окончания школы.

Раздел 2. Комплекс организационно – педагогических условий

2.1. Календарный учебный график программы (Приложение 1)

Год обучения	Дата начала занятий	Дата окончания занятий	Кол-во учебных недель	Кол-во учебных дней	Кол-во учебных часов	Режим занятий
1 год обучения	04.09.2025 г.	30.05.2026 г.	36	72	144	2 раза в нед. по 2 часа

2.2. Условия реализации программы

Материально-техническое обеспечение:

Персональный компьютер — по 1 шт. на одного обучающегося:

- Не менее 4 ГБ оперативной памяти;
- Процессор с тактовой частотой не менее 1,2 ГГц;
- Диагональ мониторов не менее 12 дюймов;
- Наличие колонок или наушников;
- Свободные 50 ГБ на накопителях;
- компьютеры (ноутбуки) должны быть подключены к единой сети с доступом в Интернет не медленнее 10 Мбит/с;

– презентационное оборудование (проектор с экраном/телевизор с большим экраном) с возможностью подключения к компьютеру (ноутбуку) – 1 комплект;

– флипчарт с комплектом листов / маркерная доска, соответствующий набор письменных принадлежностей – 1 шт.

Программное обеспечение:

- Операционная система — любая, желательно Windows;
- Любой современный браузер (например, Яндекс.Браузер, Google Chrome, Mozilla Firefox, Safari);
- Microsoft VisualStudio;
- Microsoft VisualCode.

Информационное обеспечение:

1. С# для школьников: Учебное пособие / М. Дрейер. Перевод с англ. под ред. В. Биллига - М.: Интернет - Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2009 – 128 с.
2. Обучающий контент с сайта <https://learn.unity.com/>.
3. Обучающий контент с сайта <https://stepik.org> от всероссийского социально-образовательного проекта Порцион Язык программирования С#.

Кадровое обеспечение: Программа реализуется педагогом дополнительного образования.

2.3. Формы аттестации

Для отслеживания результативности и динамики освоения ребенком образовательной программы педагогу необходимо понимать, какой уровень развития тех или иных навыков имеет учащийся при зачислении на обучение. Поэтому проводится входящая диагностика по диагностической карте промежуточной аттестации, чтобы определить уровень развития ребенка по определенным параметрам, заложенным картой. Также используются такие формы и методы контроля, как:

- наблюдение;
- устный опрос;
- устный анализ практических работ;
- итоговая аттестация заключается написанием мини-теста.

Наблюдение – необходимый педагогу метод для осуществления промежуточной и текущей аттестации, применяется педагогом постоянно. *Устный опрос* – метод при котором педагог может оценить теоретически знания учащихся. Проводится в конце пройденной темы.

Устный анализ практических работ – дает возможность учащимся научиться логически мыслить и уметь высказать собственное суждение, поможет педагогу оценить логическое мышление учащихся. Проводится в конце пройденной темы.

Итоговая аттестация проводится в конце учебного года. Задачей итоговой аттестации является определение эффективности педагогического воздействия и полноты освоения образовательной программы.

Формы предъявления и демонстрации образовательных результатов: участие в предметных олимпиадах и конкурсах.

2.4. Оценочный материал

Для отслеживания и фиксации результатов предусмотрены следующие формы контроля: опрос, наблюдение, практическая работа. Опрос позволяют своевременно и быстро выявить сложности, возникающие у обучающихся, при освоении темы занятия. Практическая работа проверяет уровень владения практическими навыкам. Наблюдение позволяет оценить групповую и индивидуальную работу обучающихся без непосредственного вмешательства педагога, здесь отслеживаются не только знания и практические навыки, но и личностные результаты, достигнутые обучающимися.

В течение всего срока обучения ведется учет результатов участия каждого обучающегося в конкурсах и олимпиадах различного уровня.

Пример итоговой аттестации (Приложение 2).

Диагностическая карта итоговой аттестации

№	ФИО обучающегося	Параметры				
		Самостоятел ьность в выполнении задач	Участие в конкурсах	Специаль ные знания	Алгоритмическ ое логическое мышление	И Т О Г

1 балл – низкий уровень; 2 балла – средний уровень; 3 балла – высокий уровень.

При подсчете баллов по каждому учащемуся можно определить уровень освоения им программы, также по каждой группе и по объединению в целом.

Определение уровня каждого учащегося выводится после подсчета баллов по всем параметрам:

- от 1 до 5 баллов – низкий уровень освоения программы учащимся;
- от 6 до 9 баллов – средний уровень освоения программы учащимся;
- от 10 до 12 баллов – высокий уровень освоения программы учащимся.

2.5. Методическое обеспечение программы

Методика преподавания включает разнообразные формы, методы и приемы обучения и воспитания. Обоснованность применения различных методов обусловлена тем, что нет ни одного универсального метода для решения разнообразных творческих задач

Методы обучения

Методы обучения, применяемые в реализации программы, можно систематизировать на основе источника получения знания:

- словесные: рассказ, объяснение, беседа, дискуссия;
- наглядные: демонстрация дидактических материалов, видеофильмов; компьютерные игры.

– практические: работа с аудио- и видеоматериалами, тематические экскурсии, интернет-экскурсии, тренинги, участие в мероприятиях.

Вместе с традиционными методами на занятиях спешно используются активные методы обучения: мозговой штурм, моделирование, метод проектов, метод эвристических вопросов, игровые ситуации, анализ конкретных ситуаций (case-study) и др.

Выбор методов обучения зависит от дидактических целей, от характера содержания занятия, от уровня развития обучающихся.

Формы организации образовательного процесса

Занятия проводятся с использованием различных форм организации учебной деятельности (групповая, фронтальная, индивидуальная, индивидуальная дистанционная, групповая дистанционная). Разнообразные формы обучения и типы занятий создают условия для развития познавательной активности, повышения интереса детей к обучению.

Формы организации учебного занятия

Основной формой проведения учебных занятий является практическое занятие. Однако в ходе реализации программы, педагог вправе применять любую из доступных форм организации учебного занятия: беседа, встреча с интересными людьми, выставка, диспут, защита проектов, игра, конкурс, конференция, круглый стол, лекция, мастеркласс, «мозговой штурм», наблюдение, олимпиада, презентация, семинар, соревнование, чемпионат, экскурсия.

Типы занятий: изучение новой информации, занятия по формированию новых умений, обобщение и систематизация изученного, практическое применение знаний, умений, комбинированные занятия, контрольно-проверочные занятия.

Педагогические технологии, используемые в образовательном процессе

1. Проектная технология, учебно-исследовательская деятельность. На протяжении всего курса обучения учащиеся вовлечены в учебно-исследовательскую деятельность, которая позволяет им находить, обрабатывать, сравнивать и систематизировать информацию, полученную из встреч с интересными людьми, публикаций в сети Интернет. В ходе образовательного процесса учащиеся создают и защищают собственные проекты, учатся методам поиска информации, самопрезентации, которые необходимы им в дальнейшей жизни и профессиональной карьере, на практических занятиях учащиеся выполняют исследовательские проекты. Проектная деятельность позволяет учащимся принять активную гражданскую позицию, сформировать потребность в участии в общественно полезной деятельности, необходимость быть нужным обществу. На занятиях создаются и реализуются учебные мини-проекты, в которых учащиеся решают учебные задачи на основе построения последовательности этапов от цели к конкретному результату. В процессе обучения осуществляется знакомство учащихся с информационно-коммуникационными технологиями, достижениями науки техники в области инженерной мысли. Современные педагогические

технологии в сочетании с современными информационными технологиями могут существенно повысить эффективность образовательного процесса, решить стоящие перед педагогом задачи воспитания всесторонне развитой, творчески свободной личности.

2. Технология развития критического мышления помогает учащимся определять приоритеты, анализировать, оценивать, выявлять ошибки, повысить мотивацию. Осуществляется при совместной работе в группах, при взаимодействии во время выполнения заданий, при диалоге обучающихся между собой и с педагогом. Обязательным условием является сбор данных о динамике обучающегося и анализ его достижений и трудностей.

Алгоритм формирования критического мышления, предполагающий ответы на следующие вопросы:

1. Какова цель данной познавательной деятельности?
2. Что известно?
3. Что делать?
4. Достигнута ли поставленная цель?

Таким образом, критическое мышление - значит «искусство суждения, основанное на критериях». Результат - владение стратегиями критического мышления.

3. Технология имитационной игры – это моделирование реальной деятельности в специально созданных условиях, а её элементы включают в себя взаимосвязанные знаниевые и деятельностные компоненты обучения. Особенности:

- не моделируется труд конкретных работников;
- имитируются лишь некоторые хозяйственные, правовые, экономические, экологические, социально-психологические принципы, определяющие поведение людей и механизмы их действий (в экстремальных ситуациях);
- моделирование только среды, особенности среды знакомы играющим в основном понаслышке, что делает анализ информации более сложным и субъективным;
- общая цель всего игрового коллектива изначально не задана, и для ее достижения самими игроками может быть найден определенный механизм взаимодействия;
- отсутствуют альтернативы, участники должны действовать лишь в предложенных вариантах;
- не программируется конфликтная ситуация (как, например, в деловых играх), а представлены только различные личные (субъективные) интересы участников игры;
- описанные сценарии игр не включают технологии и механизмы специального обучения общению и коллективному принятию решений.

4. Технология проблемного обучения способствует развитию проблемного мышления учащихся и педагога.

Результаты:

- усвоение учащимися системы знаний и способов умственной

деятельности; – развитие интеллектуальных умений и навыков учащихся;

– усвоение способов организации познавательной деятельности и формирования познавательной самостоятельности;

– развитие интеллектуальных возможностей, включающих творческие способности и прошлый опыт учащихся.

Проблемный вопрос – это входящий в состав проблемной задачи или отдельно взятый учебный вопрос (вопрос-проблема), требующий ответа на него посредством мышления. Вопрос же, требующий воспроизведения по памяти, не является проблемным. Вопросы, стимулирующие мышление, начинаются с таких вопросительных слов и словосочетаний, как «почему», «отчего», «как (чем) это объяснить», «как это понимать», «как доказать (обосновать)», «что из этого следует (какой вывод)» и т.п. А вопросительные слова «кто», «что», «когда», «где», «сколько», «какой» всегда требуют ответа на основе памяти.

Проблемная задача – учебная проблема с четкими условиями, задаваемыми преподавателем (лектором) или выявленными и сформулированными кем-либо из обучаемых (студентов), и в силу этого получившую ограниченное поле поиска (в отличие от объективно возникающей перед человеком жизненной проблемы) и ставшую доступной для решения всеми обучаемыми (студентами).

Проблемная ситуация – это ситуация познавательного затруднения, вовлекающая учащихся в самостоятельное познание элементов новой темы.

5. Интерактивные технологии направлены на развитие готовности к организации группового общения.

Результаты:

– готовность воспринимать многомерность информацию, работать в режиме полилога;

– способность выбирать и обосновывать выбор методов, форм и техник организации коммуникационного процесса;

– владение психологическими техниками и методами организации коммуникационного процесса.

6. Технология дискуссионного общения включает в себя взаимосвязанные компоненты:

– мотивационный (готовность, желание принять участие в дискуссии);

– познавательный (знание о предмете спора, проблемная ситуация);

– операционно-коммуникативный (умение вести спор, отстаивать свою точку зрения, владеть способами осуществления логических операций);

– эмоционально-оценочный (эмоциональные переживания, потребности, отношения, мотивы, оценки, личностный смысл).

Календарный план воспитательной работы

1.Модуль «Учебное занятие»			
№	Наименование мероприятия	Сроки, дата	
1.1	Проведение тематических уроков: «Урок цифры», «Урок НТИ», «Технологический диктант», «Цифровой диктант», «День науки», «Урок Победы», «Урока безопасности и навыков безопасного поведения в Интернете, информационной безопасности, повышение правовой грамотности»	В течение учебного года согласно графика	
2.Модуль «Детское объединение»			
2.1	Формирование корпуса волонтеров	Сентябрь, далее – по мере необходимости	
2.2	Обеспечение работы корпуса волонтеров в течение года	Сентябрь-май	
2.3	Организация мероприятий в рамках дня открытых дверей каникулярных школ (мастер-классы, квесты, викторины, лагерные сборы)	Сентябрь-октябрь Июнь	
3. Модуль «Взаимодействие с родителями»			
№	Тема мероприятия	Форма организации	Сроки
3.1	Консультирование семей учащихся попавших в трудную жизненную ситуацию, а также по актуальным вопросам обучения, воспитания и личностного развития детей	Индивидуальные консультации по запросу	В течении года
3.2	Проведение мониторинга эффективности реализации проекта, с целью выявления и анализа степени удовлетворенности родителей.	Анкетирование и диагностика участников проекта	Апрель-Май
3.3	Проведение родительских собраний с включением в повестку следующих вопросов: особенности ранней профориентации и предпрофессиональной подготовки учащихся, роль родителей в профориентации учащихся; организация проектной деятельности обучающихся технической направленности; участие в конкурсных мероприятиях разного уровня в рамках формирования индивидуальных образовательных траекторий учащихся.	Открытые родительские собрания	Октябрь, Апрель
3.4	«День открытых дверей»	Экскурсия, мастер-классы	Сентябрь
4. Модуль «Наставничество»			
№	Наименование мероприятия	Сроки, дата	
4.1	Сопровождение учащихся на мероприятиях различного уровня	Согласно поступающим предложениям	
4.2	Организация системы наставничества	Сентябрь, далее – по мере приема на работу молодых педагогов	

4.3	Знакомство с нормативными документами по организации образовательного процесса	Сентябрь - октябрь
4.4	Оказание помощи в овладении методами обучения и воспитания	В течение года
4.5	Практические семинары молодых педагогов: • знакомство и освоение оборудования; • правила заполнения документации; • формирование и оформление программ дополнительного образования; • типы, формы и структура занятий; • психофизиологические особенности детей разного возраста; • деятельностные образовательные практики	Сентябрь - ноябрь
4.6	Посещение учебных занятий и мероприятий	В течение года
4.7	Подготовка молодых педагогов к участию в конкурсных мероприятиях	В течение года
5. Модуль «Профессиональное самоопределение»		
№	Наименование мероприятия	Сроки, дата
5.1	Участие в работе всероссийских профориентационных проектов, созданных в сети интернет: просмотр лекций, решение учебно-тренировочных задач, участие в мастер-классах	В течение года
5.2	Экскурсии с последующим аналитическим обсуждением ИТ-компаний, промышленных предприятий в г. Хабаровске	В течение года
6. Модуль «Ключевые культурно-образовательные события»		
№	Наименование мероприятия	Срок, дата
6.1	День знаний	Сентябрь
6.2	Дни открытых дверей	Сентябрь, октябрь, май
6.3	День Учителя	Октябрь
6.4	День Народного единства	Ноябрь
6.5	День Матери в России	Ноябрь
6.6	День Героев Отечества	09 Декабря
6.7	Комплекс мероприятий «Новый год в ЦОД «ИТ- куб»»	Декабрь
6.8	День Конституции Российской Федерации	12 Декабря
6.9	День Российской науки	08 февраля
6.10	День защитника Отечества	23 февраля
6.11	Международный женский день	08 марта
6.12	День космонавтики	Апрель
6.13	День памяти о геноциде советского народа нацистами и их пособниками в годы Великой Отечественной войны	19 апреля
6.14	Праздник Весны и Труда	01 мая
6.15	День Победы	09 мая
6.16	День защиты детей	01 июня
6.17	День России	12 июня
6.18	День Памяти и скорби	Июнь
6.19	День любви, семьи и верности	Июль

Раздел 3. Список литературы

Для педагога:

1. Прайс. С# 7 и .NET Core. Кросс-платформенная разработка для профессионалов, 3-е издание, 5 мая 2021.
2. Троелсен и Джепикс. Язык программирования С# 7 и платформы .NET и .NET Core 7, 8-ое издание, 2020.
3. Скит. С# для профессионалов. Тонкости программирования, 1 января 2018.
4. 2020. 5. Албахари. С# 7.0. Справочник. Полное описание языка, 1 июля Албахари. С# 8.0. Карманный справочник, 30 апреля 2020.
5. Ферроне Х. Изучаем С# через разработку игр на Unity. СПб.: Питер. 2022.

Для обучающихся:

1. <https://stepik.org/course/62107/promo> - курса виртуальной и дополненной реальности в образовании».
2. «Технологии <https://unity.com/ru/learn> - интернет - портал unity education.
3. Unity 2020 By Example, 3 издание, 30 сентября 2020.
4. 4. Unity и С#. Геймдев от идеи до реализации | Гибсон Бонд
Гибсон Бонд Джереми 5. Троелсен и Джепикс. Язык программирования С# 7 и платформы .NET и .NET Core

Календарный учебный график

Месяц	Дата	Тема	Количество во часов	Форма контроля
Сентябрь		Вводное занятие. ТБ. Знакомство с программой.	2	Опрос, беседа
		История языков программирования. Введение в язык программирования C#. Среда программирования C#.	2	Опрос, беседа, практическая работа
		Типы данных и вывод. Определение переменной	2	Опрос, беседа, практическая работа
		Арифметические выражения и операции	2	Опрос, беседа, практическая работа
		Ввод данных и проверка ввода	2	Опрос, беседа, практическая работа
		Примеры решения задач.	2	Практическая работа
		Отработка навыков решения простейших задач.	2	Практическая работа
		Строки: базовые операции	2	Опрос, беседа, практическая работа
		Логические выражения и булевы операции.	2	Опрос, беседа, практическая работа
Октябрь		Условные операторы (if/else, else if).	2	Опрос, беседа
		Условные операторы (if/else, else if).	2	Практическая работа
		Примеры решения задач.	2	Практическая работа
		Примеры решения задач.	2	
		Отработка навыков решения простейших задач.	2	Практическая работа

		Отработка навыков решения простейших задач.	2	Практическая работа
		Цикл for и практические задачи	2	Опрос, беседа, практическая работа
		Цикл for и практические задачи	2	Практическая работа
		Цикл while и do-while Примеры решения задач.	2	Опрос, беседа, практическая работа
Ноябрь		Цикл while и do-while Примеры решения задач.	2	Практическая работа
		Отработка навыков решения простейших задач.	2	Практическая работа
		Отработка навыков решения простейших задач.	2	Практическая работа
		Операторы break и continue. Накопители (аккумуляторы)	2	Опрос, беседа, практическая работа
		Отработка навыков решения простейших задач.	2	Практическая работа
		Массивы (одномерные)	2	Опрос, беседа, практическая работа
		Массивы (одномерные)	2	Практическая работа
		Отработка навыков решения простейших задач	2	Практическая работа
Декабрь		Отработка навыков решения простейших задач	2	Практическая работа
		Списки (List<T>) и работа с коллекциями	2	Опрос, беседа, практическая работа
		Отработка навыков решения простейших задач	2	Практическая работа
		Отработка навыков решения простейших задач	2	Практическая работа
		Функции (методы): объявление и вызов	2	Опрос, беседа, практическая работа
		Отработка навыков решения простейших задач	2	Практическая работа
		Область видимости и параметры (ref, out)	2	Опрос, беседа,

				практическая работа
Январь		Примеры решения задач.	2	Практическая работа
		Рекурсия (введение)	2	Опрос, беседа, практическая работа
		Примеры решения задач.	2	Практическая работа
		Отработка навыков решения задач.	2	Практическая работа
		Отработка навыков решения задач.	2	Практическая работа
		Строки: Substring, IndexOf, Replace, Split/Join	2	Опрос, беседа, практическая работа
		Примеры решения задач текстовых задач	2	Практическая работ
Февраль		Форматирование вывода и работа с вещественными числами	2	Опрос, беседа, практическая работа
		Примеры решения задач.	2	Практическая работа
		Ввод/вывод файлов (простая работа с файлами)	2	Опрос, беседа, практическая работа
		Примеры решения задач	2	Опрос, беседа, практическая работа
		Исключения и обработка ошибок (try/catch)	2	Опрос, беседа, практическая работа
		Основы объектно-ориентированного программирования: классы и объекты	2	Опрос, беседа, практическая работа
		Примеры решения задач	2	Практическая работа
		Инкапсуляция и свойства (get/set)	2	Опрос, беседа, практическая работа
Март		Примеры решения задач	2	Практическая работа
		Наследование и полиморфизм (введение)	2	Опрос, беседа, практическая работа
		Примеры решения задач	2	Практическая работа
		Интерфейсы и их использование	2	Опрос, беседа,

				практическая работа
		Создание интерфейса IPlayable для игровых объектов	2	Практическая работа
		LINQ: основы и простые запросы	2	Опрос, беседа, практическая работа
		Решение задач	2	Практическая работа
		События и делегаты (простое знакомство)	2	Опрос, беседа, практическая работа
Апрель		Простая имитация событий внутри программы (подписка/вызов)	2	Практическая работа
		Итераторы и yield (введение)	2	Опрос, беседа, практическая работа
		Решение задач	2	Практическая работа
		Решение задач	2	Практическая работа
		Игровой мини-проект: текстовая игра (часть 1)	2	Опрос, беседа, практическая работа
		Игровой мини-проект: текстовая игра (часть 2)	2	Практическая работа
		Графическое введение (опционально): простая форма WinForms или Unity intro	2	Опрос, беседа, практическая работа
		Алгоритмы: сортировка, поиск (вводный уровень)	2	Опрос, беседа, практическая работа
Май		Модули и проектная структура. Документирование кода	2	Опрос, беседа, практическая работа
		Безопасность и этика программирования, ресурсы для продолжения обучения	2	Опрос, беседа, практическая работа
		Подготовка итогового проекта	2	Опрос, беседа, практическая работа
		Реализация итоговых проектов	2	Практическая работа
		Реализация итоговых проектов	2	Практическая работа
		Реализация итоговых проектов	2	Практическая работа

		Реализация итоговых проектов	2	Практическая работа
		Итоговое занятие. Рефлексия и планы на будущее	2	Итоговая аттестация
ИТОГО:			144	

Образец итоговой аттестации

1. Что выполняет команда `Console.WriteLine("Hello")` в C#?

- A) Сохраняет текст в файл
- B) Выводит текст в консоль и переводит курсор на новую строку
- C) Читает строку с клавиатуры
- D) Останавливает программу

2. Какой тип данных в C# лучше всего подходит для хранения целого числа без дробной части?

- A) `string`
- B) `double`
- C) `int`
- D) `bool`

3. Что делает конструкция `if (x > 0) { ... }` в программе?

- A) Повторяет код, пока `x > 0`
- B) Выполняет код внутри фигурных скобок только если `x > 0`
- C) Объявляет переменную `x`
- D) Сравнивает `x` с нулём и всегда выводит `true`

4. Какой оператор используется для получения остатка от деления в C#?

- A) `/`
- B) `%`
- C) `*`
- D) `//`

5. Что делает метод `int.TryParse(s, out n)`?

- A) Всегда превращает `s` в число без ошибок
- B) Пытается преобразовать строку `s` в целое число и возвращает `true` при успехе
- C) Меняет тип переменной `n` на `string`
- D) Удаляет пробелы в строке `s`

6. Какой цикл лучше использовать, если заранее известно количество повторений (например, 10 раз)?

- A) `while`
- B) рекурсию
- C) `for`
- D) `if`

7. Что такое массив в C#?

- A) Набор функций

- В) Последовательность однотипных элементов, доступных по индексу
- С) Способ вывода текста
- Д) Особый вид комментария

8. Какой модификатор доступа делает поле или метод доступным только внутри того же класса?

- А) public
- В) private
- С) protected
- Д) internal

9. Что означает ключевое слово static в объявлении метода static void Main()?

- А) Метод можно вызвать без создания объекта класса
- В) Метод всегда возвращает число
- С) Метод выполняется в отдельном окне
- Д) Метод компилируется отдельно

10. Что из перечисленного лучше всего подходит для обработки ошибки во время выполнения (например, некорректный ввод)?

- А) Использовать комментарии
- В) try { ... } catch { ... }
- С) Удалить программу
- Д) Писать код без проверки

Ключ ответов:

- 1 — В
- 2 — С
- 3 — В
- 4 — В
- 5 — В
- 6 — С
- 7 — В
- 8 — В
- 9 — А
- 10 — В

