

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
«27» августа 2024 года

УТВЕРЖДАЮ

Генеральный директор
КГАНОУ «Краевой центр образования»

/Черёмухин П.С.
«27» августа 2024 года



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА ТЕХНИЧЕСКОЙ
НАПРАВЛЕННОСТИ

«Основы программирования на языке Java»

Уровень освоения: стартовый
Возраст учащихся: 13-18 лет
Общий объем программы в часах: 144 часа

Составители программы:
Валетова М.А., методист

Хабаровск
2024 г.

Раздел 1. Комплекс основных характеристик ДООП

1.1. Пояснительная записка

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

Программа разработана с учетом следующих нормативно-правовых документов:

- Федерального закона РФ от 29.12.2012г. № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями);
- Приказа Министерства Просвещения РФ от 27.07.2022 г. № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Приказа Министерства труда и социальной защиты РФ от 22.09.2021г. № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;
- Постановления Главного государственного санитарного врача РФ от 28.09.2020г. № 28 "Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Распоряжения Правительства РФ от 31.03.2022г. № 678-р «Концепция развития дополнительного образования детей до 2030 года»;
- Положения о дополнительной общеобразовательной программе, реализуемой в Хабаровском крае, утвержденного приказом КГАОУ ДО РМЦ от 26.09.2019 г. № 383П;
- Устав краевого государственного автономного нетипового образовательного учреждения «Краевой центр образования».

Актуальность программы. На сегодня разработка программного обеспечения является наиболее востребованным направлением в любых сферах применения. Кроме того, большое развитие мобильных платформ даёт более широкий выбор направлений разработки.

В ИТ-индустрии язык Java является одним из самых востребованных объектно-ориентированных языков программирования. Java стремительно развивается и охватывает практически все области программирования. Java используется для создания серверного программного обеспечения, прикладных программ, разработки веб-сайтов, игр и многого другого.

В современном мире Java как платформа является наиболее популярной в связи с тем, что не имеет требований к операционной системе для запуска своих приложений. Изучение языка программирования Java по данной программе обучения даёт возможность создавать программы в среде разработки на платформе Java.

Программа ориентирована на развитие у обучающихся навыков проектирования программ на языке программирования Java.

Педагогическая целесообразность. Программ является практико-ориентированной. Освоенный подростками теоретический материал закрепляется в виде опросов, задач и проектов. В ходе освоения учебных блоков обучающиеся получают практические навыки инженерной, творческой и технологической деятельности. В процессе обучения применяются актуальные педагогические методики, учитывающие необходимость формирования современных методов мышления и развития командообразования.

Адресат программы: Программа ориентирована на обучающихся возрастной категории от 13 до 18 лет. Состав группы постоянный, количество обучающихся от 8 человек.

Форма обучения: очная

Объем 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 Java.

Задачи программы:

Предметные:

- изучить конструкции языка программирования Java;
- познакомить с принципами и методами функционального и объектно-ориентированного программирования; основными структурами данных и типовыми методами обработки этих структур;
- научить работать в интегрированной среде разработки на языке Java.

Метапредметные:

- формирование и развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
- приобретение навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- развитие у обучающихся интереса к программированию, самостоятельности и творческого подхода к решению задач с

использованием средств вычислительной техники.

Воспитательные:

- воспитывать аккуратность и дисциплинированность при выполнении работы;
- способствовать формированию положительной мотивации к трудовой деятельности;
- способствовать формированию опыта совместного и индивидуального творчества при выполнении командных заданий;
- способствовать профессиональной ориентации обучающихся.

1.3. Учебный план

№ п/п	Название разделов, тем	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1.	Введение. Техника безопасности. Среда разработки	2	1	1	Опрос, беседа
2.	Основы программирования на Java	68	32	36	
2.1.	Структура программы	2	1	1	Опрос, наблюдение, практическая работа
2.2.	Переменные и константы	2	1	1	Опрос, наблюдение, практическая работа
2.3.	Типы данных	2	1	1	Опрос, наблюдение, практическая работа
2.4.	Консольный ввод/вывод Java	2	1	1	Опрос, наблюдение, практическая работа
2.5.	Класс Console	2	1	1	Опрос, наблюдение, практическая работа
2.6.	Арифметические операции	2	1	1	Опрос, наблюдение, практическая работа
2.7.	Математические вычисления и класс Math	2	1	1	Опрос, наблюдение, практическая работа
2.8.	Поразрядные операции	2	1	1	Опрос, наблюдение, практическая работа
2.9.	Условные выражения	2	1	1	Опрос, наблюдение, практическая работа
2.10.	Операции присваивания и приоритет операции	2	1	1	Опрос, наблюдение, практическая работа
2.11.	Преобразование базовых типов данных	2	1	1	Опрос, наблюдение, практическая работа

2.12.	Условные конструкции	2	1	1	Опрос, практическая работа
2.13.	Конструкция if/else	2	1	1	Опрос, практическая работа
2.14.	Конструкция switch/case	2	1	1	Опрос, наблюдение, практическая работа
2.15.	Тернарные операции	2	1	1	Опрос, практическая работа
2.16.	Циклы. Цикл for	2	1	1	Опрос, практическая работа
2.17.	Цикл do	2	1	1	Опрос, наблюдение, практическая работа
2.18.	Цикл while	2	1	1	Опрос, наблюдение, практическая работа
2.19.	Операторы continue и break	2	1	1	Опрос, практическая работа
2.20.	Управляющие структуры. Условные переходы	2	1	1	Опрос, практическая работа
2.21.	Массивы	2	1	1	Опрос, наблюдение, практическая работа
2.22.	Многомерный массив	2	1	1	Опрос, практическая работа
2.23.	Зубчатый массив	2	1	1	Опрос, наблюдение, практическая работа
2.24.	Списки	2	1	1	Опрос, наблюдение, практическая работа
2.25.	Работа со строками	2	1	1	Опрос, практическая работа
2.26.	Регулярные выражения	2	1	1	Опрос, наблюдение, практическая работа
2.27.	Методы	2	1	1	Опрос, наблюдение, практическая работа
2.28.	Параметры методов	2	1	1	Опрос, наблюдение, практическая работа
2.29.	Оператор return.Результат метода	2	1	1	Опрос, наблюдение, практическая работа
2.30.	Перегрузка методов	2	1	1	Опрос, наблюдение, практическая работа
2.31.	Рекурсивные функции	2	1	1	Опрос, наблюдение, практическая работа
2.32.	Введение в обработку исключений	2	1	1	Опрос, наблюдение, практическая работа
2.33.	Проектная работа	4	-	4	Самостоятельная работа
3.	Объектно- ориентированное программирование	42	19	23	
3.1.	Классы и объекты	2	1	1	Опрос, наблюдение, практическая работа

3.2.	Конструкторы	2	1	1	Опрос, наблюдение, практическая работа
3.3.	Пакеты	2	1	1	Опрос, наблюдение, практическая работа
3.4.	Импорт пакетов и классов	2	1	1	Опрос, наблюдение, практическая работа
3.5.	Модификаторы доступа и инкапсуляция	2	1	1	Опрос, наблюдение, практическая работа
3.6.	Статистические члены и модификатор static	2	1	1	Опрос, наблюдение, практическая работа
3.7.	Объекты как параметры методов	2	1	1	Опрос, наблюдение, практическая работа
3.8.	Внутренние и вложенные классы	2	1	1	Опрос, наблюдение, практическая работа
3.9.	Наследие	2	1	1	Опрос, наблюдение, практическая работа
3.10.	Абстрактные классы	2	1	1	Опрос, наблюдение, практическая работа
3.11.	Иерархия наследования и преобразования типов	2	1	1	Опрос, наблюдение, практическая работа
3.12.	Интерфейсы	2	1	1	Опрос, наблюдение, практическая работа
3.13.	Интерфейсы в механизме обратного вызова	4	2	2	Опрос, наблюдение, практическая работа
3.14.	Перечисления enum	2	1	1	Опрос, наблюдение, практическая работа
3.15.	Обобщения	2	1	1	Опрос, наблюдение, практическая работа
3.16.	Ограничения обобщений	2	1	1	Опрос, наблюдение, практическая работа
3.17.	Ссылочные типы и клонирование объектов	2	1	1	Опрос, наблюдение, практическая работа
3.18.	Класс Records	2	1	1	Опрос, наблюдение, практическая работа
3.19.	Проектная работа	4	-	4	Самостоятельная работа
4.	Коллекции	30	13	17	
4.1.	Типы коллекций. Интерфейс Collection	2	1	1	Опрос, наблюдение, практическая работа
4.2.	Класс ArrayList и интерфейс List	2	1	1	Опрос, наблюдение, практическая работа
4.3.	Очереди и класс ArrayDeque	2	1	1	Опрос, наблюдение, практическая работа
4.4.	Класс LinkedList	2	1	1	Опрос, наблюдение, практическая работа
4.5.	Интерфейс Set и класс HashSet	2	1	1	Опрос, наблюдение, практическая работа
4.6.	Интерфейс SortedSet	2	1	1	Опрос, наблюдение, практическая работа
4.7.	Интерфейс NavigableSet	2	1	1	Опрос, наблюдение, практическая работа
4.8.	Интерфейс TreeSet	2	1	1	Опрос, наблюдение, практическая работа
4.9.	Интерфейс Comparable и Compronation.Сортировка	2	1	1	Опрос, наблюдение, практическая работа

4.10.	Интерфейс Map и класс Hash Map	2	1	1	Опрос, наблюдение, практическая работа
4.11.	Класс TreeMap	2	1	1	Опрос, наблюдение, практическая работа
4.12.	Интераторы	2	1	1	Опрос, наблюдение, практическая работа
4.13.	Классы StringBuffer и StringBuilder	2	1	1	Опрос, наблюдение, практическая работа
4.14.	Проектная работа	4	-	4	Самостоятельная работа
5.	Итоговое занятие	2	-	2	Презентация проектов
	Итого:	144	65	79	

1.4. Содержание учебного плана

Тема 1. Введение. Техника безопасности. Среда разработки.

Теория: Общие сведения о Java, знакомство с программным обеспечением JDK, знакомство со средой разработки IntelliJ IDEA.

Раздел 2. «Основы программирования на Java»

Тема 2.1. Структура программы.

Теория: Знакомство со структурой программы, понятие инструкции, выполнение программы.

Практика: Составление простой программы, запуск, остановка, поиск ошибок в программе.

Тема 2.2. Переменные и константы.

Теория: Понятие переменной, понятие инициализации, ключевые слова, понятие константы.

Практика: Присвоение переменным значений и операции с ними.

Тема 2.3. Типы данных.

Теория: Базовые типы данных, значения типов данных, целые числа, числа с плавающей точкой, символы и строки.

Практика: Решение задач.

Тема 2.4. Консольный ввод/вывод на Java.

Теория: Операторы ввода, вывода данных, объект класса Scanner.

Практика: Ввод и вывод данных в консоль, решение задач.

Тема 2.5. Класс Console.

Теория: Класс Console, методы класса Console.

Практика: Ввод и вывод данных с помощью класса Console и его методов.

Тема 2.6. Арифметические операции.

Теория: Операнды, операторы, префиксный инкремент, постфиксный инкремент, префиксный декремент, постфиксный декремент, приоритет арифметических операций, ассоциативность операций, операции с числами с плавающей точкой.

Практика: Решение арифметических задач с помощью кода.

Тема 2.7. Математические вычисления и класс Math

Теория: Класс Math, методы класса Math для решения математических вычислений.

Практика: Решение математических задач с помощью класса Math и его методов.

Тема 2.8. Поразрядные операции.

Теория: Понятие поразрядных операций, логические операции, операции сдвига

Практика: Решение задач

Тема 2.9. Условные выражения.

Теория: Понятие условных выражений, операции сравнения.

Практика: Решение задач.

Тема 2.10. Операции присваивания и приоритет операций.

Теория: Определение операции присваивания, приоритет операций присваивания.

Практика: Решение задач.

Тема 2.11. Преобразование базовых типов данных.

Теория: Явные и неявные преобразования, автоматические преобразования, потеря данных при преобразовании, усечение рациональных чисел до целых, преобразование при операциях.

Практика: Решение задач с преобразованием типов данных.

Тема 2.12. Условные конструкции.

Теория: Понятие условных конструкций, истинные и ложные условия.

Практика: Решение задач с условиями.

Тема 2.13. Конструкция if/else.

Теория: Парадигма ветвления в условном операторе, назначение и применение.

Практика: Простейшие программы с использованием ветвлений.

Тема 2.14. Конструкция switch/case.

Теория: Оператор выбора switch/case, синтаксис оператора, использование нескольких условий.

Практика: Решение задачи с несколькими условиями.

Тема 2.15. Тернарные операции.

Теория: Понятие тернарной операции, синтаксис тернарных операций.

Практика: Решение задач с применением тернарных операций.

Тема 2.16. Циклы. Цикл for.

Теория: Понятие циклов, итерации, объявление цикла, простейшие программы с использованием оператора циклов for.

Практика: Решение задач с использованием цикла for.

Тема 2.17. Цикл do.

Теория: Понятие условного цикла, модель построения и принцип работы.

Практика: Решение задач с использованием условного цикла.

Тема 2.18. Цикл while.

Теория: Проверка истинности условия, принцип работы.

Практика: Разработка алгоритмов и программ. Решение задач.

Тема 2.19. Операторы continue и break.

Теория: Выход из цикла, продолжение цикла.

Практика: Решение задач.

Тема 2.20. Управляющие структуры. Условные переходы.

Теория: Получение навыков по работе с управляющими структурами в Java, умение применять логические операции при построении алгоритмов, разработка алгоритмов и их реализация,

Практика: Решение задач.

Тема 2.21. Массивы.

Теория: Определение массива, назначение массива, принцип создания и поиск элементов в массиве.

Практика: Решение задач.

Тема 2.22. Многомерный массив.

Теория: Определение многомерного массива, длина массива.

Практика: Решение задач.

Тема 2.23. Зубчатый массив.

Теория: Определение зубчатого массива, массив с различным количеством элементов.

Практика: Решение задач.

Тема 2.24. Списки.

Теория: развитие навыков применения массивов и списков для хранения и структурирования информации, методы списка, преобразование массивов, удаление из массива, динамические списки.

Практика: Решение задач.

Тема 2.25. Работа со строками.

Теория: получение навыков работы со строковыми данными в Java, Класс String, ключевые характеристики класса, создание строк, сложение строк, сравнение строк.

Практика: Решение задач

Тема 2.26. Регулярные выражения.

Теория: Определение регулярных выражений, синтаксис, метасимволы.

Практика: Решение задач.

Тема 2.27. Методы.

Теория: Определение метода, модификаторы метода public и static, название метода.

Практика: Решение задач.

Тема .2.28. Параметры методов.

Теория: Определение параметра метода, аргументы, параметры переменной длины.

Практика: Решение задач.

Тема .2.29. Оператор return. Результат метода.

Теория: Определение оператора return, возвращаемые значения, выход из метода.

Практика: Решение задач.

Тема 2.30. Перегрузка методов.

Теория: Определение механизма перегрузки метода, использование одного имени метода с разными параметрами.

Практика: Решение задач.

Тема 2.31. Рекурсивные функции.

Теория: Определение рекурсии, использование рекурсивных функций, базис рекурсии, шаг рекурсии.

Практика: Решение задач.

Тема 2.32. Введение в обработку исключений.

Теория: Определение исключений, обработка нескольких исключений, оператор throw.

Практика: Решение задач.

Тема 2.33. Проектная работа.

Теория: Введение в проектную деятельность. Основы проектной деятельности. Консультации экспертов. Подготовка

Практика: создание собственного приложения по индивидуальному замыслу на основе пройденного материала. Создание рабочей группы, проектирование, тестирование, отладка, запуск проекта.

Раздел 3. «Объектно-ориентированное программирование»

Тема 3.1. Классы и объекты.

Теория: Определение классов и объектов, ключевое слово this, инициализаторы.

Практика: Решение задач

Тема 3.2. Конструкторы.

Теория: Определение конструктора, оператор new, выделение памяти.

Практика: Решение задач.

Тема 3.3. Пакеты.

Теория: Определение пакетов, организация классов, логические наборы.

Практика: Решение задач с использованием пакетов.

Тема 3.4. Импорт пакетов и классов.

Теория: Получение доступа к пакетам, путь к файлу в пакете, директива import, статический импорт.

Практика: Решение задач с помощью импорта пакетов.

Тема 3.5. Модификаторы доступа и инкапсуляция.

Теория: Определение модификатора доступа, модификаторы public, private, protected, модификатор по умолчанию, определение инкапсуляции.

Практика: Решение задач.

Тема 3.6. Статические члены и модификатор static.

Теория: Статические поля, статические константы, статические инициализаторы, статические методы.

Практика: Решение задач.

Тема 3.7. Объекты как параметры методов.

Теория: Определение объекта, передача объектов в методы.

Практика: Решение задач.

Тема 3.8. Внутренние и вложенные классы.

Теория: Определение внутренних и вложенных классов, синтаксически вложенные классы.

Практика: Решение задач.

Тема 3.9. Наследование

Теория: Определение наследования, ключевое слово `extends`, ключевое слово `super`, предопределение методов, запрет наследования, динамическая диспетчеризация методов.

Практика: Создание программы с использованием наследования.

Тема 3.10. Абстрактные классы

Теория: Определение абстрактных классов, ключевое слово `abstract`, абстрактные методы.

Практика: Решение задач.

Тема 3.11. Иерархия наследования и преобразования типов

Теория: Принципы работы иерархии наследования и преобразования классов, оператор `instanceof`, суперклассы.

Практика: Решение задач.

Тема 3.12. Интерфейсы

Теория: Определение интерфейсов, ключевое слово `implements`, интерфейсы в преобразовании типов, константы в интерфейсах, множественная реализация интерфейсов, наследование интерфейсов, вложенные интерфейсы.

Практика: Решение задач

Тема 3.13. Интерфейсы в механизме обратного вызова

Теория: Определение механизма обратного вызова и его взаимодействие с интерфейсами.

Практика: Решение задач.

Тема 3.14. Перечисления `enum`

Теория: Определение типа `enum`, методы перечислений, конструкторы перечислений.

Практика: Решение задач.

Тема 3.15. Обобщения

Теория: Определение обобщения, базисные типы, универсальные параметры, интерфейсы обобщений, обобщенные методы.

Практика: Решение задач с использованием нескольких универсальных параметров.

Тема 3.16. Ограничения обобщений.

Теория: Обобщенные типы в качестве ограничений, интерфейсы в качестве ограничений, множественные ограничения.

Практика: Решение задач.

Тема 3.17. Ссылочные типы и клонирование объектов.

Теория: Определение ссылочных типов, интерфейс `Cloneable`, исключение `CloneNotSupportedException`.

Практика: Решение задач.

Тема 3.18. Класс `Records`.

Теория: Определение функциональности `Records`, ключевое слово `record`, конструктор `record`, ограничения `records`.

Практика: Решение задач.

Тема 3.19. Проектная работа.

Теория: Подготовка к соревнованиям, конкурсам (разбор положений), оформление проекта.

Практика: Работа над проектом, проектирование, тестирование, отладка, запуск проекта, демонстрация отчёта в группе.

Раздел 4. «Коллекции»

Тема 4.1. Типы коллекций. Интерфейс Collection

Теория: Определение коллекций, интерфейсы коллекций, применение коллекций.

Практика: Решение задач

Тема 4.2. Класс ArrayList и интерфейс List

Теория: Определение класса ArrayList, методы интерфейса List.

Практика: Решение задач.

Тема 4.3. Очереди и класс ArrayDeque

Теория: Определение класса ArrayDeque, интерфейс Queue, интерфейс Deque.

Практика: Решение задач.

Тема 4.4. Класс LinkedList

Теория: Определение класса LinkedList, методы LinkedList и их применения.

Практика: Решение задач.

Тема 4.5. Интерфейс Set и класс HashSet

Теория: Определение интерфейса Set, определение обобщенного класса HashSet, хеш-таблица.

Практика: Решение задач.

Тема 4.6. Интерфейс SortedSet.

Теория: Определение интерфейса SortedSet, создание коллекций в отсортированном виде.

Практика: Решение задач.

Тема 4.7. Интерфейс NavigableSet

Теория: Определение интерфейса NavigableSet, извлечение элементов из коллекций.

Практика: Решение задач.

Тема 4.8. Интерфейс TreeSet

Теория: Определение интерфейса TreeSet, отсортированная структура данных.

Практика: Решение задач.

Тема 4.9. Интерфейсы Comparable и Comparator. Сортировка

Теория: Определение интерфейсов Comparable и Comparator, метод compare, сортировка по нескольким критериям.

Практика: Решение задач.

Тема 4.10. Интерфейс Map и класс HashMap

Теория: Определение интерфейса Map и класса HashMap, классы отображений, абстрактный класс AbstractMap.

Практика: Решение задач.

Тема 4.11. Класс TreeMap

Теория: Определение класса TreeMap, автоматическая сортировка объектов.

Практика: Решение задач.

Тема 4.12. Итераторы

Теория: Определение итератора, интерфейс Iterator, интерфейс ListIterator.

Практика: Решение задач.

Тема 4.13. Классы StringBuffer и StringBuilder

Теория: Определение классов StringBuffer и StringBuilder, получение и установка символов, добавление в строку, замена в строке, обрезка строки, удаление строки, изменение длины.

Практика: Решение задач.

Тема 4.14. Проектная работа

Теория: Подготовка к соревнованиям, конкурсам (разбор положений), оформление проекта.

Практика: Работа над проектом, проектирование, тестирование, отладка, запуск проекта, демонстрация отчёта в группе.

5. Итоговое занятие

Практика: Презентация проектов.

Планируемые результаты

Предметные:

- знают основные принципы объектно-ориентированного программирования, основные особенности построения программ на языке программирования Java;
- знают основные средства реализации принципов объектно-ориентированного программирования;
- умеют разрабатывать графический пользовательский интерфейс средствами языка программирования Java;
- умеют реализовывать программно-основные принципы объектно-ориентированного программирования средствами языка программирования Java.

Метапредметные:

- владение основами самоконтроля, способность к принятию решений;
- развитие логического мышления, самостоятельности, познавательного потенциала обучающегося;
- смогут самостоятельно разрабатывать алгоритмы достижения целей (решения задач, выполнения проектов и т.д.);
- умеют организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

– .

Личностные:

- формирование ответственного отношения к учению, готовности и способности обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию;

- готовность и способность учащихся к саморазвитию и реализации творческого потенциала в предметно-продуктивной деятельности за счет развития их образного, алгоритмического и логического мышления;
- приобретение начальной профессиональной подготовки по данному направлению, что поспособствует повышению социальной адаптации учащихся после окончания школы;
- умение и готовность работать в команде.

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	02.09.2024 г.	30.05.2025 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, Chrome, Mozilla Firefox, Safari);
- программное обеспечение JAVA (Java Development Kit (JDK));
- интегрированная среда разработки Eclipse IDE.

Информационное обеспечение:

- Сообщество IT специалистов <https://habr.com/ru/> ;
- Яков Файн «Программирование на Java для детей» http://yfain.github.io/Java4Kids/#_java_building_blocks;
- Шилдт, Герберт, Java: руководство для начинающих, 7-е изд.:

Пер. с англ. - СПб.: ООО "Диалектика", 2019. - 816 с.: ил.. PDF.

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 самостоятельных работ;
- самостоятельные работы;
- итоговая аттестация проводится в форме презентации

самостоятельно выполненного проекта.

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.

Самостоятельные работы – помогут учащимся проверить свои знания по пройденным темам и оценить приобретенные умения, проводятся в конце каждого раздела. Участие в проектной деятельности – это условие, необходимое для становления самостоятельности учащихся, которые сформируют установку на самостоятельность, индивидуальные цели и возможности их реализации.

Итоговая аттестация проводится в конце учебного года.

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, самостоятельная 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, олимпиадах и хакатонах различного уровня.

Пример итоговой аттестации (Приложение 2).

Диагностическая карта итоговой аттестации

№	ФИО обучающегося	Параметры				
		Самостоятельность в выполнении задач	Участие в конкурсах	Специальные знания	Алгоритмическое логическое мышление	ИТОГ

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

№ п/п	Наименование мероприятия	Форма проведения	Срок исполнения
1.	Проведение инструктажей с обучающимися по порядку действия в случае возникновения пожара, правилам поведения на воде и дорогах.	Час общения	сентябрь
2.	Единый урок по безопасности: «Безопасность на дорогах глазами детей»; «Безопасность в сети Интернет»; «Осторожно! Тонкий лёд. Падение снежных масс и наледи»; «К нам приходит Новый год!»; «Что мы знаем о терроризме»; «Я и мои виртуальные друзья»; «Действия при пожаре – правила пожарной безопасности»; «Осторожно! Загрязнение пластмассовыми материалами»; «Безопасное лето».	Дискуссия, устный журнал, встречи с сотрудниками МЧС, составление памяток и рекомендаций	ежемесячно
3.	Уроки истории нашей страны: «Моя малая родина»; «Край родной – навек любимый»; «Интересные и знаменитые люди нашего края».	Фотовыставка, устный журнал, поисково-исследовательская работа, встреча с интересными людьми	ноябрь январь апрель
4.	День воинской славы России: День героев Отечества России; День защитника Отечества; День Победы.	Военно-патриотическая беседа, встречи с участниками боевых действий, экскурсия в музей	декабрь февраль май
5.	«Как у наших у ворот...» – Масленица; – Пасха. Пасхальные торжества.	Игровая программа	март апрель
	Мы разные, но у нас равные права!	Занятие- обсуждение	ноябрь
	Честность прежде всего	Устный журнал	январь
	Что такое «хорошо» и что такое «плохо»?	Дискуссия	март
	«Моя семья - моя крепость»	Детско-родительская встреча	май
	«Что значит быть ответственным»	Занятие- обсуждение	май
6.	«Речная лента», «Покормите птиц зимой» и др.	акции	В течение года

7.	Подготовка и участие в конкурсах, олимпиадах, хакатонах и других мероприятиях	Образовательное мероприятие	В течение года
	Цикл мероприятий «Участвуй в НТО»	Образовательное мероприятие	Сентябрь – октябрь
	Экскурсии в IT-компаниях	экскурсия	По согласованию
	Ярмарка проектов и достижений	День открытых дверей	Декабрь, май

Раздел 3. Список литературы

Список литературы для педагога:

1. Блох Д. Java. «Эффективное программирование» – Лори., 2014 г.
2. Седжвик Р. , Уэйн К.«Алгоритмы на Java» –Санкт-Петербург, Вильямс, 2016г.
3. Шилдт Г. «Java. Полное руководство» –Санкт-Петербург, Вильямс, 2015 г.
4. Орам Э., Уилсон Г. «Идеальный код» – Санкт-Петербург, Вильямс, 2011г.
5. Эккель Б. «Философия Java»–Москва, Питер, 2009 г.
6. Аккуратов Е. Е. «Знакомьтесь: Java»–Санкт-Петербург, Вильямс, 2006 г.
7. Сьерра К., Бэйтс Б. «Изучаем Java» – Москва, Эксмо, 2012 г.
8. Васильев А. Н. «Java. Объектно-ориентированное программирование» – Санкт-Петербург, Питер, 2011 г.
9. Машнин Т. «Современные Java-технологии на практике» – Москва, БХВПетербург, 2010 г.
10. Хабибуллин И. «Самоучитель Java» – Санкт-Петербург, БХВ-Петербург, 2008 г

Список литературы для обучающихся:

1. Босова, Л.Л. Занимательные задачи по информатике [Электронный ресурс] / Л.Л. Босова, А.Ю. Босова, Ю.Г. Коломенская. – 5-е изд. – М.: БИНОМ. Лаборатория знаний, 2013.
2. Златопольский, Д. М. Сборник задач по программированию [Электронный ресурс] / Д.М. Златопольский. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2011.
3. Шень, А. Программирование: теоремы и задачи [Электронный ресурс] / А. Шень. - 6-е изд., дополненное. М.: МЦНМО, 2017.

Календарный учебный график

Месяц	Дата	Тема	Формы контроля	Кол-во часов
Сентябрь		Введение. Техника безопасности. Среда разработки	Опрос, беседа	2
		Структура программы	Опрос, наблюдение, практическая работа	2
		Переменные и константы	Опрос, наблюдение, практическая работа	2
		Типы данных	Опрос, наблюдение, практическая работа	2
		Консольный ввод/вывод Java	Опрос, наблюдение, практическая работа	2
		Класс Console	Опрос, наблюдение, практическая работа	2
		Арифметические операции	Опрос, наблюдение, практическая работа	2
		Математические вычисления и класс Math	Опрос, наблюдение, практическая работа	2
Октябрь		Поразрядные операции	Опрос, наблюдение, практическая работа	2
		Условные выражения	Опрос, наблюдение, практическая работа	2
		Операции присваивания и приоритет операции	Опрос, наблюдение, практическая работа	2
		Преобразование базовых типов данных	Опрос, наблюдение, практическая работа	2
		Условные конструкции	Опрос, практическая работа	2
		Конструкция if/else	Опрос, практическая работа	2
		Конструкция switch/case	Опрос, наблюдение, практическая работа	2
		Тернарные операции	Опрос, практическая работа	2
		Циклы. Цикл for	Опрос, практическая работа	2
		Цикл do	Опрос, наблюдение, практическая работа	2
Ноябрь		Цикл while	Опрос, наблюдение, практическая работа	2
		Операторы continue и break	Опрос, практическая работа	2
		Управляющие структуры. Условные переходы	Опрос, практическая работа	2
		Массивы	Опрос, наблюдение, практическая работа	2

		Многомерный массив	Опрос, практическая работа	2
		Зубчатый массив	Опрос, наблюдение, практическая работа	2
		Списки	Опрос, наблюдение, практическая работа	2
		Работа со строками	Опрос, практическая работа	2
Декабрь		Регулярные выражения	Опрос, наблюдение, практическая работа	2
		Методы	Опрос, наблюдение, практическая работа	2
		Параметры методов	Опрос, наблюдение, практическая работа	2
		Оператор return.Результат метода	Опрос, наблюдение, практическая работа	2
		Перегрузка методов	Опрос, наблюдение, практическая работа	2
		Рекурсивные функции	Опрос, наблюдение, практическая работа	2
		Введение в обработку исключений	Опрос, наблюдение, практическая работа	2
		Проектная работа	Самостоятельная работа	2
Январь		Проектная работа	Самостоятельная работа	2
		Классы и объекты	Опрос, наблюдение, практическая работа	2
		Конструкторы	Опрос, наблюдение, практическая работа	2
		Пакеты	Опрос, наблюдение, практическая работа	2
		Импорт пакетов и классов	Опрос, наблюдение, практическая работа	2
		Модификаторы доступа и инкапсуляция	Опрос, наблюдение, практическая работа	2
Февраль		Статистические члены и модификатор static	Опрос, наблюдение, практическая работа	2
		Объекты как параметры методов	Опрос, наблюдение, практическая работа	2
		Внутренние и вложенные классы	Опрос, наблюдение, практическая работа	2
		Наследие	Опрос, наблюдение, практическая работа	2
		Абстрактные классы	Опрос, наблюдение, практическая работа	2
		Иерархия наследования и преобразования типов	Опрос, наблюдение, практическая работа	2
		Интерфейсы	Опрос, наблюдение, практическая работа	2
		Интерфейсы в механизме	Опрос, наблюдение,	2

		обратного вызова		
Март		Интерфейсы в механизме обратного вызова	Практическая работа	2
		Перечисления enum	Опрос, наблюдение, практическая работа	2
		Обобщения	Опрос, наблюдение, практическая работа	2
		Ограничения обобщений	Опрос, наблюдение, практическая работа	2
		Ссылочные типы и клонирование объектов	Опрос, наблюдение, практическая работа	2
		Класс Records	Опрос, наблюдение, практическая работа	2
		Проектная работа	Самостоятельная работа	2
		Проектная работа	Самостоятельная работа	2
Апрель		Типы коллекций. Интерфейс Collection	Опрос, наблюдение, практическая работа	2
		Класс ArrayList и интерфейс List	Опрос, наблюдение, практическая работа	2
		Очереди и класс ArrayDeque	Опрос, наблюдение, практическая работа	2
		Класс LinkedList	Опрос, наблюдение, практическая работа	2
		Интерфейс Set и класс HashSet	Опрос, наблюдение, практическая работа	2
		Интерфейс SortedSet	Опрос, наблюдение, практическая работа	2
		Интерфейс NavigableSet	Опрос, наблюдение, практическая работа	2
		Интерфейс TreeSet	Опрос, наблюдение, практическая работа	2
		Интерфейс Comparable и Comparator. Сортировка	Опрос, наблюдение, практическая работа	2
Май		Интерфейс Map и класс HashMap	Опрос, наблюдение, практическая работа	2
		Класс TreeMap	Опрос, наблюдение, практическая работа	2
		Интераторы	Опрос, наблюдение, практическая работа	2
		Классы StringBuffer и StringBuilder	Опрос, наблюдение, практическая работа	2
		Проектная работа	Самостоятельная работа	2
		Проектная работа	Самостоятельная работа	2
		Итоговое занятие	Презентация проектов	2
		Итого:		144

Образец итоговой аттестации

Краткое описание математической задачи:

Решение:

Напишите программу, которая выводит на консоль простые числа в промежутке от [2, 100]. Используйте для решения этой задачи оператор "%" (остаток от деления) и циклы.

Пример выполнения:

```
for(int i = 2; i <= 100; i++){
    boolean isPrime = true;

    for(int j = 2; j < i; j++){
        if(i % j == 0){
            isPrime = false;
            break;
        }
    }

    if(isPrime){
        System.out.println(i);
    }
}
```

Или, используя [циклы с метками](#):

```
out_for:
for (int i = 2; i <= 100; i++) {
    for (int j = 2; j < i; j++) {
        if (i % j == 0) {
            continue out_for;
        }
    }
    System.out.println(i);
}
```