



Управление образования администрации г. Белгорода
Муниципальное бюджетное учреждение дополнительного образования
«Центр технологического образования и детского технического творчества»
г. Белгорода

Согласовано:
Руководителем МО
«Дополнительное образование»

Л.А.Баронова
Протокол № 5 от 31.05.2023 г.

Согласовано:
Заместитель директора
МБУДО ЦТОиДТТ
Ю.С.Феоктисова

«31» мая 2023 г.

Утверждаю:
Директор МБУДО ЦТОиДТТ
Ю.Н. Кумейко

«31» мая 2023 г.
Приказ № 67 от 31.05.2023 г.



Дополнительная
общеобразовательная (общеразвивающая) программа

«Логическое программирование»

Направленность: техническая
Уровень программы: стартовый
Возраст учащихся: 12-17 лет
Срок реализации: 1 год.

Автор – составитель:
педагог дополнительного образования
Пронькин Андрей Владимирович

г. Белгород,
2023 г.

**Дополнительная общеобразовательная (общеразвивающая) программа:
«Логическое программирование»** технической направленности

Автор программы: Пронькин Андрей Владимирович

**Программа рассмотрена и утверждена на заседании педагогического
совета МБУДО ЦТОиДТТ
от « 31 » мая 2023 г., протокол № 7.**

Председатель


(подпись)

Ю.Н. Кумейко
Ф.И.О.

Оглавление:

1. Комплекс основных характеристик программы	
1.1. Пояснительная записка	3
1.2. Учебный план	10
1.3. Содержание программы	15
1.4. Календарный учебный график	23
1.5. Формы аттестации	24
2. Комплекс организационно – педагогических условий реализации программы	
2.1. Система оценки образовательной результатов	25
2.2. Оценочные материалы	26
2.3. Материально - техническое обеспечение	32
2.4. Методическое обеспечение	33
2.5. Информационное обеспечение	33
2.6. Список методической литературы	34

Приложение

№ 1. Календарно – тематический план

1. Комплекс основных характеристик программы

1.1. Пояснительная записка

Совсем недавно компьютерное программирование казалось таинственным ремеслом, уделом специалистов. Мысль о том, что программирование может быть увлекательным занятием для каждого, большинству людей и в голову не приходила. Но мир изменился. Интернет, электронная почта, социальные сети, смартфоны и мобильные приложения, ворвавшись в нашу жизнь, преобразили ее всего за несколько лет. Мы охотно поглощаем любые плоды компьютеризации – от интернет-шопинга и развлечений до новостей и игр. Однако мы можем не только использовать эти технологии, научившись программировать, мы можем развивать их, создавать собственные произведения цифрового искусства.

Всем, что делает компьютер, управляют строки программного кода, введенные с клавиатуры. Компьютерный код похож на иностранный язык, но язык этот может освоить каждый. Учиться программировать очень интересно, ведь можно получить результаты сразу же. Более того, создание игр и программ – это отличная возможность для творчества, развития логики и интеллекта, которые важны в самых разных областях – от науки до юриспруденции. Количество вакансий, где нужно умение программировать растет с каждым днем.

Последовательное и планомерное обучение на пути освоения технического мастерства в области программирования дополняется знакомством с созданием трендов в программировании, в области сайтостроения, что имеет особое значение для развития и раскрытия своего мастерства в технической направленности в школьном возрасте.

Учащимся предоставляется возможность выбора творческой формы, художественных средств выразительности. Они приобретают опыт технической деятельности в строении сайтов, программировании, совмещают правила технической направленности с элементами фантазии.

Теоретические знания, практические занятия и развитие технического восприятия представлены в программе в их содержательном единстве.

Программа «Логическое программирование» разработана на основе авторской программы «Видеодизайн и компьютерная графика» (автор: Пронькин А.В.).

Программа «Логическое программирование» **авторская, технической направленности.**

Отличительные особенности данной дополнительной общеобразовательной (общеразвивающей) программы заключается в том, что она не только развивает общеучебные и общеинтеллектуальные умения и навыки, но и формирует у учащихся интерес к профессиям, связанных с программированием, способствующих и повышающих развитие техническо-творческих способностей детей. Расширение кругозора учащихся,

самостоятельного подхода к решению технических вопросов под контролем педагога.

Новизна дополнительной общеобразовательной (общеразвивающей) программы состоит в поэтапном изучении основ теории программирования и применении этих основ на практике.

Актуальность дополнительной общеобразовательной (общеразвивающей) программы заключается в том, что предложенный материал способствует:

- обеспечению необходимых условий для личностного развития;
- профессиональному самоопределению;
- адаптации их к жизни в обществе;
- организации содержательного досуга;
- способствует запросу современной жизни, требованиям учебно-воспитательного процесса.

Программа «Логическое программирование» актуальна еще в том, что занятия стимулируют стремление к самостоятельной деятельности, создают условия для развития личностных качеств учащихся, способствуют подготовке детей к новым тенденциям в программировании.

Предложенный вид деятельности вызывает мотивацию познания и творчества. Занятия развивают эстетический вкус, мышление, воображение, формируют конструктивные навыки. Повышают качество проводимого после школьных занятий времени, что развивает коммуникативные умения, содействуют профилактике асоциального поведения детей и подростков.

Педагогическая целесообразность дополнительной общеобразовательной (общеразвивающей) программы состоит в создании условий для получения учащимися новых знаний и актуализацию уже имеющихся теоретических сведений по основам рекламных веб-макетов и компьютерного дизайна.

Дополнительная программа направлена на:

- дизайнерское развитие учащихся в сфере веб-рекламы, дизайна и сайтостроения;
- ознакомление с современными и перспективными технологиями дизайна;
- творческое и эстетическое развитие учащихся;
- умение находить и использовать необходимую информацию;
- выдвигать идеи решения возникающих задач при разработке веб-рекламы.

Цель программы: формирование и развитие компетенций учащихся в области программирования, приобретение опыта использования информационных технологий в индивидуальной и коллективной учебной и познавательной, в том числе проектной деятельности.

Образовательные:

- сформировать навыки разработки конструирования несложных программ веб-сайта;
- дать базовые навыки программирования на языках HTML, CSS, JavaScript;

- реализовать коммуникативные, технические способности учащихся в ходе проектирования и конструирования сайтов;
- сформировать отношение к компьютеру как инструменту для творчества, созидания, реализации своих потребностей;
- сформировать навыки элементарного проектирования, конструирования, размещения и сопровождения мультимедийного проекта;
- ознакомление учащихся с основными направлениями истории рекламы дизайна.

Развивающие:

- развивать пространственное, образное мышление учащихся;
- развивать творческий подход к делу и поиску нестандартных решений в процессе создания Web-сайта;
- развивать эстетический вкус и дизайнерское мышление;
- развивать творческие авторские начала через создание самостоятельных проектов, участие в конкурсах и олимпиадах;
- способствовать обогащению жизненного опыта и формированию системы ценностей учащихся;
- повышать грамотность учащихся в области Интернет-технологий.

Воспитательные:

- формирование у учащихся социальной активности, гражданской позиции, культуры общения и поведения в социуме, навыков здорового образа жизни;
- воспитание умения эффективно работать в команде и индивидуально над решением нестандартных задач;
- воспитание трудолюбия, коллективизма, обязательности, честности и культуры поведения, как в объединении, так и в обществе;
- воспитание бережного отношения к оборудованию.

Основные принципы обучения, используемые программой:

- принцип творческой активности,
- принцип сознательности,
- принцип последовательности,
- связь теории с практикой.

Использование разнообразных форм и методов обучения повышает продуктивность занятий, повышает интерес учащихся к учебному процессу. В процессе обучения применяются *групповые занятия, индивидуальные, теоретические и практические.*

При реализации программы используются следующие ***современные образовательные технологии***, основанные на личностно-ориентированном подходе: технология индивидуализации обучения, проблемное обучение, игровые технологии, технология коллективного взаимообучения, технология создания успеха, здоровьесберегающие технологии. Применение технологии создания успеха дает ребенку возможность осознать свою творческую ценность, продвигает к новым высотам творческих достижений.

Взаимодействие с родителями

Наибольшую эффективность работы в дополнительном образовании дает *способ совместной деятельности педагога и родителей.*

Формы работы с родителями:

- Родительское собрание.
- Совместное посещение выставок.
- Участие в мероприятиях, проводимых в рамках образовательной программы.

Возрастные особенности детей (данная программа рассчитана на детей среднего и старшего школьного возраста (12 - 17 лет)).

Средний школьный возраст - это возраст перехода от детства к юности. В этом возрасте происходит рост и развитие всего организма. Неравномерное физическое развитие детей оказывает влияние на их поведение: они часто жестикулируют, движения порывисты, плохо координированы. Характерная черта восприятия детей среднего школьного возраста – специфическая избирательность, поэтому содержание общеобразовательной (общеразвивающей) программы подобрано с учетом интересов и познавательных возможностей учащихся. В этом возрасте идет интенсивное нравственное и социальное формирование личности. Объем данной общеобразовательной (общеразвивающей) программы соответствует возможностям и уровню развития детей данного возраста.

Старший школьный возраст - все познавательные процессы, сформированные еще в подростковом возрасте, в старшем школьном возрасте будут только укрепляться и совершенствоваться. Главное для старшего школьника теперь - выход во взрослый мир, овладение профессией, а значит, нахождение своего места в мире. Доминантой становится выбор и овладение профессией, поскольку от этого зависит дальнейшая жизнь, которую избирает человек на пороге взрослой жизни. Соответственно новая доминанта изменяет отношение к учению, заставляя соотносить свои старания и практическое их применение.

Организация образовательного процесса

Срок реализации программы: 1 год

Рекомендуемый возраст учащихся: 12-17 лет

Количество часов: 144 часа

Режим занятий: 2 раза в неделю по 2 часа

Наполняемость групп: 12-15 человек

Программа может быть использована педагогами учреждений дополнительного образования.

Основная форма проведения занятий - *учебное занятие*. Занятия состоят из теоретической и практической частей. *Теоретическая часть* занятия включает просмотр видеоуроков, просмотр и анализ работ. *Практическая часть* занятия включает общие практические занятия, индивидуальные занятия.

Возможна реализация дополнительной общеобразовательной (общеразвивающей) программы «Логическое программирование» с применением электронного обучения и дистанционных образовательных технологий.

Уровень освоения программы – стартовый, предназначен для получения учащимися первоначальных знаний в области программирования и сопутствующих дисциплин (электроника и информатика).

Планируемые результаты программы

В результате завершения обучения по программе «Логическое программирование» учащиеся должны:

Знать:

- принципы построения Java документа и работы с его элементами;
- алгоритмы применения технологии Java
- сущность, назначение и структуру объектной модели браузера и документа;
- элементы и конструкции языка Java и способы их применения;
- принципы функционирования виртуального сервера;
- элементы и конструкции языка PHP и способы их применения для работы с сетью;
- современные технологии и средства разработки WEB-приложений;
- способы и средства публикации, поддержки, поисковой оптимизации и обновления WEB-документа.

Уметь:

- создавать описания стилей элемента Java документа различными способами;
- создавать разработки на языке Java
- создавать разработки серверных сценариев на языке PHP.
- создавать WEB-документ и работать с базовыми его элементами;
- изменять свойства документа и его элементов средствами технологии PHP
- создавать клиентские сценарии, осуществлять их внедрение в проект и тестирование;
- загружать в интернет рекламные веб-баннеры и веб-рекламу.

Личностные и метапредметные результаты освоения программы

Результаты	Формируемые умения	Средства формирования
<i>личностные</i>	<ul style="list-style-type: none"> - формирование у детей мотивации к обучению, о помощи им в самоорганизации и саморазвитии; - развитие познавательных 	организация на занятиях парно-групповой работы

	<p>навыков учащихся, умений самостоятельно конструировать свои знания, ориентироваться в информационном пространстве, развитие критического и творческого мышления</p>	
Метапредметные результаты		
регулятивные	<ul style="list-style-type: none"> - планировать свои действия в соответствии с поставленной задачей и условиями ее реализации, в том числе и во внутреннем плане; - осуществлять итоговый и пошаговый контроль по результату 	<ul style="list-style-type: none"> - в сотрудничестве с педагогом ставить новые учебные задачи; - преобразовывать практическую задачу в познавательную; - проявлять познавательную инициативу в учебном сотрудничестве
познавательные	<ul style="list-style-type: none"> - умения учиться навыкам решения творческих задач и навыкам поиска, анализа и интерпретации информации; - добывать необходимые знания и с их помощью проделывать конкретную работу; - осуществлять поиск необходимой информации для выполнения учебных заданий с использованием учебной литературы; - осуществлять анализ объектов с выделением существенных и 	<ul style="list-style-type: none"> - осуществлять расширенный поиск информации с использованием различных источников информации и Интернета

	несущественных признаков	
<i>коммуникативные</i>	<ul style="list-style-type: none"> - учиться выполнять различные роли в группе (лидера, исполнителя, критика); - умение координировать свои усилия с усилиями других; - формулировать собственное мнение и позицию; - договариваться и приходить к общему решению в совместной деятельности; - задавать вопросы; - допускать возможность существования у людей различных точек зрения, не совпадающих с собственной, и ориентироваться на позицию партнера в общении и взаимодействии; - учитывать разные мнения и стремиться к координации различных позиций в сотрудничестве 	<ul style="list-style-type: none"> - учитывать разные мнения и интересы и обосновывать собственную позицию; - понимать относительность мнений и подходов к решению проблемы; - аргументировать свою позицию и координировать ее с позицией партнеров в сотрудничестве при выработке общего решения в совместной деятельности; - продуктивно разрешать конфликты на основе учета интересов и позиций всех его участников; - с учетом целей коммуникации достаточно точно, полно и последовательно передавать необходимую информацию партнеру, как ориентир для построения действия

1.2. Учебный план

№ п/п	Разделы программы и темы учебных занятий	Количество часов			Формы контроля (аттестаци и)
		Теори я	Практ ика	Все го	
1.	Вводное занятие. Инструктаж по технике безопасности. Знакомство с программами программирования	2	-	2	Беседа
2.	Язык программирования Java. Первое знакомство. Язык Python в игровой индустрии	10	10	20	
2.1.	Установка среды программирования Java. Выбор инструмента программирования Java.	1	1	2	Устный опрос, просмотр работ
2.2.	Путешествие в мир языка программирования Java. Создаем первую программу на Java.	1	1	2	
2.3.	Принципы работы программ на Java. Знакомство с библиотеками языка Java.	1	1	2	
2.4.	Основы программирования на Java. Сохранение и изменение информации в программе.	1	1	2	
2.5.	Работа со строками. Условные инструкции. Циклы.	1	1	2	
2.6.	Объекты и массивы на Java. Продвинутые методики программирования на Java.	1	1	2	
2.7.	Разработка графического интерфейса пользователя. Менеджеры компоновки.	1	1	2	

2.8.	Создание интернет-приложений. Чтение и запись файлов. Создание двухмерной графики. Использование HTTP-клиента. Создание модов для Minecraft с помощью Java.	1	1	2	
2.9.	Сервер и библиотеки Minecraft языка Python. Процедура запуска.	1	1	2	
2.10.	Архитектурные сооружения. Блоки и функции. Стены – циклы. Здания. Подвесные мосты (Теорема Пифагора). Создание улиц и городов Minecraft работа с функциями и списками. Создание транспорта.	1	1	2	
3	PHP 7	20	20	40	
3.1.	История языка PHP. Пример PHP программы. Использование PHP в Web.	1	1	2	Тестирование, просмотр работ
3.2.	Переменные, константы и типы данных.	1	1	2	
3.3.	Выражения и операции в PHP.	1	1	2	
3.4.	Ассоциативные массивы. Конструкции языка.	1	1	2	
3.5.	Функции в области видимости. Генераторы языка.	1	1	2	
3.6.	Строковые функции. Работа с массивами.	1	1	2	
3.7.	Работа с файлами и каталогами языка. Права доступа и атрибуты файлов. Запуск внешних программ.	1	1	2	
3.8.	Работа с датой и временем.	1	1	2	
3.9.	Основы объективно-ориентированного программирования PHP 7. Объекты и классы. Наследование. Обработка ошибок.	1	1	2	
3.10.	Предопределенные классы PHP 7. Календарные классы. Отражения.	1	1	2	
3.11.	Работа с сетью в PHP 7. Сетевые функции.	1	1	2	
3.12.	Посылка писем через PHP.	1	1	2	
3.13.	Расширения PHP. Фильтрация и проверка данных	1	1	2	
3.14.	Работа с СУБД MySQL. Администрирование баз данных.	1	1	2	
3.15.	Работа с изображениями при помощи PHP.	1	1	2	

3.16.	Сервер memcached и библиотеки PHP.	1	1	2	
3.17.	Приемы программирования на PHP.	1	1	2	
3.18.	Протокол SSH. Установка и создание виртуальных машин и их запуск	1	1	2	
3.19.	Система контроля Git. Web-сервер nginx.	1	1	2	
3.20.	PHP-FPM. Администрирование MySQL.	1	1	2	
4.	C++ Шаг за шагом	18	18	36	Тестирование, просмотр работ
4.1.	Краткая история C++. Основы языка.	1	1	2	
4.2.	Знакомство с данными, типами и операторами языка программирования C++	1	1	2	
4.3.	Предложение управления программой при помощи C++. Циклы.	1	1	2	
4.4.	Массивы, строки, указатели языка C++	1	1	2	
4.5.	Введение в функции языка программирования C++	1	1	2	
4.6.	Углубление в функции языка программирования C++.	1	1	2	
4.7.	Углубление в типы и операторы языка программирования C++.	1	1	2	
4.8.	Введение в классы и объекты языка программирования C++.	1	1	2	
4.9.	Углубление в классы и объекты языка программирования C++.	1	1	2	
4.10.	Наследование. Полиморфизм ООП языка программирования C++.	1	1	2	
4.11.	C++ и система ввода и вывода.	1	1	2	
4.12.	Исключения, шаблоны и другие дополнительные темы языка.	1	1	2	
4.13.	Фундаментальные алгоритмы C++.	1	1	2	
4.14.	Создание калькулятора на языке C++.	1	1	2	
4.15.	Гибкое программирование на C++.	1	1	2	

4.16.	Создание игр на C++. Игровой движок на Defold.	1	1	2	
4.17.	Программирование игр на C++ и Open GL. Создание игрового движка на Open GL.	1	1	2	
4.18.	Пишем Minecraft на C++. Окно. Прямоугольник с текстурами. Камера и матрица. Текстурный атлас. Объединяем чанки, строим и разрушаем предметы. Пишем Minecraft на C++. АО сохранение, mip-mapping, ландшафт, освещение, бесконечный мир.	1	1	2	
5.	Программирование на C #	10	10	20	
5.1.	Введение в C #. Обзор компиляторов языка программирования.	1	1	2	Тестирование, просмотр работ
5.2.	Базовая структура C# и вывод информации. Переменные и типы данных.	1	1	2	
5.3.	Математические действия. Условные операторы (if-else). Оператор switch-case.	1	1	2	
5.4.	Операторы циклов. Массивы данных. Одномерные и многомерные.	1	1	2	
5.5.	Цикл «foreach» для работы с массивами. Generics. Создание функций.	1	1	2	
5.6.	Обработчик исключений. Создание классов и объектов. Введение в ООП языка.	1	1	2	
5.7.	Конструкторы классов. Модификатор static. Указатель «this» в классах.	1	1	2	
5.8.	Аксессуары get и set. Наследование классов.	1	1	2	
5.9.	Операторы is и as. Массив указателей. Виртуальные методы.	1	1	2	
5.10.	Многопоточное программирование. Приложение с графическим интерфейсом.	1	1	2	
6.	Android-программирование – 8 часов	4	4	8	
6.1.	Установка Android Studio и первая программа. Введение в Java Android. Обзор программы и активные состояния. Простейший калькулятор. Работа с полями и Toast. Диалоговые окна и кнопки рейтинга. Переход между строками.	1	1	2	Тестирование, просмотр работ
6.2.	Красивый дизайн приложения. Работа с изображениями. Компоненты List View и	1	1	2	

	SeekBar. Фрагменты. Жесты и AutoComplete Text View. Создание звуков в приложении. Сохранение текста в файл. Простой браузер и его создание. Создание файла с расширением .apk				
6.3.	Создание простых мобильных игр на Android на языке программирования Java.	1	1	2	
6.4.	Язык программирования Kotlin. Kotlin Android установка и обзор создания мобильных приложений. Создание игры на Kotlin Android.	1	1	2	
7.	Аттестация	3	3	6	
7.1	Начальная аттестация	1	1	2	Тестирование, практическая работа
7.2	Промежуточная аттестация	1	1	2	Тестирование, практическая работа
7.3	Промежуточная аттестация	1	1	2	Тестирование, практическая работа
8.	Подготовка к выставкам и конкурсам	5	5	10	Просмотр
9.	Итоговое занятие	1	1	2	Просмотр работ
Всего часов:		73	71	144	

1.3. Содержание программы

1. Вводное занятие. Инструктаж по технике безопасности. Знакомство с программами программирования – 2 часа

Теоретические сведения: Техника безопасности, правила поведения в работе с компьютерной техникой. Правила безопасного труда при работе с приборами, питающимися от сети переменного тока. Оказание первой медицинской помощи при травмах и электротравмах. Правила противопожарной безопасности и эвакуации при возникновении пожара. Обязанности учащихся по окончании занятий. Общее знакомство учащихся с программами программирования.

2. Язык программирования Java. Первое знакомство. Язык Python в игровой индустрии – 20 часа

2.1. Установка среды программирования Java. Выбор инструмента программирования Java – 2 часа

Теоретическая часть: Познакомить учащихся как устанавливается среда программирования языка Java и какой выбрать инструмент программирования Java.

Практическая часть: Научиться устанавливать среду и инструмент программирования Java.

2.2. Путешествие в мир языка программирования Java. Создаем первую программу на Java – 2 часа

Теоретическая часть: Знакомство учащихся с миром языка программирования и созданием первой программы на Java.

Практическая часть: Сделать предложенные проекты веб-дизайна.

2.3. Принципы работы программ на Java. Знакомство с библиотеками языка Java – 2 часа

Теоретическая часть: Принципы работы программ на Java. Знакомство с библиотеками языка Java.

Практическая часть: Сделать предложенные проекты веб-дизайна.

2.4. Основы программирования на Java. Сохранение и изменение информации в программе – 2 часа

Теоретическая часть: Основы программирования на Java. Сохранение и изменение информации в программе.

Практическая часть: Сделать предложенные проекты веб-дизайна.

2.5. Работа со строками. Условные инструкции. Циклы – 2 часа

Теоретическая часть: Принципы работы со строками, условными инструкциями и циклами языка программирования Java.

Практическая часть: Сделать предложенные проекты веб-дизайна.

2.6. Объекты и массивы на Java. Продвинутое методиками программирования на Java – 2 часа

Теоретическая часть: Рассказать об объектах и массивах, а также о продвинутой методике программирования на Java.

Практическая часть: Сделать предложенные проекты веб-дизайна.

2.7. Разработка графического интерфейса пользователя. Менеджеры компоновки – 2 часа

Теоретическая часть: Разработка графического интерфейса пользователя и о менеджерах компоновки Java.

Практическая часть: Сделать предложенные проекты веб-дизайна.

2.8. Создание интернет-приложений. Чтение и запись файлов. Создание двумерной графики. Использование HTTP-клиента. Создание модов для Minecraft с помощью Java – 2 часа

Теоретическая часть: Создание интернет-приложений, чтение и запись файлов, создание двумерной графики с помощью Java.

Практическая часть: Сделать предложенные проекты веб-дизайна.

2.9. Сервер и библиотеки Minecraft языка Python. Процедура запуска – 2 часа

Теоретическая часть: Серверы и библиотеки Minecraft языка Python, и процедура запуска.

Практическая часть: Сделать предложенные проекты веб-дизайна.

2.10. Архитектурные сооружения. Блоки и функции. Стены – циклы. Здания. Подвесные мосты (Теорема Пифагора). Создание улиц и городов Minecraft работа с функциями и списками. Создание транспорта – 2 часа

Теоретическая часть: Создание архитектурных сооружений на языке Python.

Практическая часть: Сделать предложенные проекты веб-дизайна.

3. PHP 7 – 40 часов

3.1. История языка PHP. Пример PHP программы. Использование PHP в Web – 2 часа

Теоретическая часть: Рассказать об истории языка PHP, как создать PHP программу и об использовании PHP в Web.

Практическая часть: Сделать предложенные проекты.

3.2 Переменные, константы и типы данных. – 2 часа

Теоретическая часть: Изучение создания переменных, константы и типы данных.

Практическая часть: Сделать предложенные проекты.

3.3. Выражения и операции в PHP – 2 часа

Теоретическая часть: Изучение материала о выражениях и операциях в PHP.

Практическая часть: Сделать предложенные проекты.

3.4. Ассоциативные массивы. Конструкции языка – 2 часа

Теоретическая часть: Ассоциативные массивы и конструкции языка.

Практическая часть: Выполнение предложенных проектов.

3.5. Функции в области видимости. Генераторы языка – 2 часа

Теоретическая часть: Функции в области в видимости и генераторах языка.

Практическая часть: Выполнение предложенных проектов.

3.6. Строковые функции. Работа с массивами – 2 часа

Теоретическая часть: Создание строковых функций и работа с массивами.

Практическая часть: Выполнить предложенные проекты.

3.7. Работа с файлами и каталогами языка. Права доступа и атрибуты файлов. Запуск внешних программ – 2 часа

Теоретическая часть: Рассказать о работе с файлами и каталогами языка PHP, о правах доступа и атрибутах файлов, а также о запуске внешних программ языка.

Практическая часть: Изучение и выполнение проектов по теме.

3.8. Работа с датой и временем – 2 часа

Теоретическая часть: Изучение темы, как работать с датой и временем, как их создавать.

Практическая часть: Сделать предложенные проекты.

3.9. Основы объективно-ориентированного программирования PHP 7. Объекты и классы. Наследование. Обработка ошибок – 2 часа

Теоретическая часть: Основы ООП PHP, наследование, объекты и классы, обработка ошибок.

Практическая часть: Выполнить предложенные проекты.

3.10. Предопределенные классы PHP 7. Календарные классы. Отражения – 2 часа

Теоретическая часть: Рассказать о предопределенных классах PHP 7.

Практическая часть: Сделать предложенные проекты.

3.11. Работа с сетью в PHP 7. Сетевые функции – 2 часа

Теоретическая часть: Рассказать, как работать с сетью в PHP 7 и о сетевых функциях.

Практическая часть: Сделать предложенные проекты.

3.12. Посылка писем через PHP – 2 часа

Теоретическая часть: Как с помощью PHP посылаются письма.

Практическая часть: Сделать предложенные проекты.

3.13. Расширения PHP. Фильтрация и проверка данных – 2 часа

Теоретическая часть: Рассказать о расширениях PHP и фильтрации, и проверке данных.

Практическая часть: Сделать предложенные проекты.

3.14. Работа с СУБД MySQL. Администрирование баз данных – 2 часа

Теоретическая часть: Рассказать, как работать с СУБД MySQL и администрировать базы данных.

Практическая часть: Сделать предложенные проекты.

3.15. Работа с изображениями при помощи PHP – 2 часа

Теоретическая часть: Рассказать о работе с изображениями при помощи PHP.

Практическая часть: Сделать предложенные проекты.

3.16. Сервер memcached и библиотеки PHP – 2 часа

Теоретическая часть: Изучение темы о сервере memcached и библиотеках PHP.

Практическая часть: Сделать предложенные проекты.

3.17. Приемы программирования на PHP – 2 часа

Теоретическая часть: Рассказать о приемах программирования на PHP.

Практическая часть: Выполнить предложенные проекты.

3.18. Протокол SSH. Установка и создание виртуальных машин и их запуск – 2 часа

Теоретическая часть: Рассказать, как работает протокол SSH и как создавать, устанавливать, а также запускать виртуальные машины.

Практическая часть: Сделать предложенные проекты.

3.19. Система контроля Git. Web-сервер nginx – 2 часа

Теоретическая часть: Что такое система контроля Git и web-сервере nginx?

Практическая часть: Выполнить предложенные проекты.

3.20. PHP-FPM. Администрирование MySQL – 2 часа

Теоретическая часть: Рассказать о PHP-FPM и администрировании MySQL.

Практическая часть: Сделать предложенные проекты.

4. C++ Шаг за шагом – 36 часов

4.1. Краткая история C++. Основы языка – 2 часа

Теоретическая часть: Рассказать о краткой истории и об основах языка C++.

Практическая часть: Закрепление материала на практике.

4.2. Знакомство с данными, типами и операторами языка программирования C++. – 2 часа

Теоретическая часть: Познакомить с данными, типами и операторами языка C++.

Практическая часть: Сделать предложенные проекты.

4.3. Предложение управления программой при помощи C++. Циклы. – 2 часа.

Теоретическая часть: Рассказать, как управлять запрограммированной программой и создавать циклы.

Практическая часть: Сделать предложенные проекты.

4.4. Массивы, строки, указатели языка C++ – 2 часа

Теоретическая часть: Рассказать, как создаются массивы, строки и указатели языка программирования C++.

Практическая часть: Сделать предложенные проекты.

4.5. Введение в функции языка программирования C++ – 2 часа

Теоретическая часть: Введение в функции языка программирования C++.

Практическая часть: Сделать предложенные проекты.

4.6. Углубление в функции языка программирования C++ – 2 часа

Теоретическая часть: Рассказать о глубоком изучении функций языка программирования C++.

Практическая часть: Сделать предложенные проекты.

4.7. Углубление в типы и операторы языка программирования C++ – 2 часа

Теоретическая часть: Рассказать о глубоком изучении типов и операторов языка программирования C++.

Практическая часть: Выполнить предложенные проекты.

4.8. Введение в классы и объекты языка программирования C++ – 2 часа

Теоретическая часть: Рассказать о начальном изучении классов и объектов языка программирования C++.

Практическая часть: Сделать предложенные проекты.

4.9. Углубление в классы и объекты языка программирования C++ – 2 часа

Теоретическая часть: Рассказать о глубоком изучении классов и объектов языка программирования C++.

Практическая часть: Сделать предложенные проекты.

4.10. Наследование. Полиморфизм ООП языка программирования C++ – 2 часа

Теоретическая часть: Рассказать, как создается наследование и полиморфизм ООП языка программирования C++.

Практическая часть: Сделать предложенные проекты.

4.11. C++ и система ввода и вывода – 2 часа

Теоретическая часть: Что такое система ввода и вывода в языке программирования C++ и как она создается.

Практическая часть: Выполнить предложенные проекты.

4.12. Исключения, шаблоны и другие дополнительные темы языка – 2 часа

Теоретическая часть: Рассказать об исключениях, шаблонах и других дополнительных темах языка программирования.

Практическая часть: Сделать предложенные проекты.

4.13. Фундаментальные алгоритмы языка программирования C++ – 2 часа

Теоретическая часть: Фундаментальные алгоритмы языка программирования.

Практическая часть: Сделать предложенные проекты.

4.14. Создание калькулятора на языке C++ – 2 часа

Теоретическая часть: Рассказать, как создается калькулятор разных типов на C++.

Практическая часть: Сделать предложенные проекты.

4.15. Гибкое программирование на C++ – 2 часа

Теоретическая часть: Создание гибкого программирования на C++.

Практическая часть: Сделать предложенные проекты

4.16. Создание игр на C++. Игровой движок на Defold – 2 часа

Теоретическая часть: Рассказать, как создать двухмерные игры на движке Defold.

Практическая часть: Сделать предложенные проекты.

4.17. Программирование игр на C++ и Open GL. Создание игрового движка на Open GL – 2 часа

Теоретическая часть: Рассказать, как программировать игры при помощи связки C++ и Open GL.

Практическая часть: Сделать предложенные проекты.

4.18. Пишем Minecraft на C++. Окно. Прямоугольник с текстурами. Камера и матрица. Тектурный атлас. Объединяем чанки, строим и разрушаем предметы. Пишем Minecraft на C++. АО сохранение, mip-mapping, ландшафт, освещение, бесконечный мир – 2 часа

Теоретическая часть: Рассказать, как писать игру Minecraft на C++.

Практическая часть: Сделать предложенные проекты.

5. Программирование на C # – 20 часов

5.1. Введение в C #. Обзор компиляторов языка программирования. – 2 часа.

Теоретическая часть: Рассказать о компиляторах, работающих с этим языком программирования.

Практическая часть: Сделать предложенные проекты.

5.2. Базовая структура C# и вывод информации. Переменные и типы данных.

Теоретическая часть: Рассказать о структуре C# и выводе информации, а также о переменных и типах данных.

Практическая часть: Сделать предложенные проекты.

5.3. Математические действия. Условные операторы (if-else). Оператор switch-case.

Теоретическая часть: Рассказать, что такое условные операторы и как производятся в С# математические действия.

Практическая часть: Сделать предложенные проекты.

5.4. Операторы циклов. Массивы данных. Одномерные и многомерные

Теоретическая часть: Рассказать, что такое операторы циклов, одномерные и многомерные массивы данных.

Практическая часть: Сделать предложенные проекты.

5.5. Цикл «foreach» для работы с массивами. Generics. Создание функций

Теоретическая часть: Рассказать, как циклы работают с массивами и о создании функций.

Практическая часть: Сделать предложенные проекты.

5.6. Обработчик исключений. Создание классов и объектов. Введение в ООП языка

Теоретическая часть: Рассказать, что такое обработчик исключений, как создаются классы и объекты, а также введение в ООП языка.

Практическая часть: Сделать предложенные проекты.

5.7. Конструкторы классов. Модификатор statics. Указатель «this» в классах

Теоретическая часть: Рассказать, что такое конструкторы классов, модификаторы и указатели.

Практическая часть: Сделать предложенные проекты.

5.8. Аксессоры get и set. Наследование классов

Теоретическая часть: Рассказать, как работают аксессоры и что такое наследование классов.

Практическая часть: Сделать предложенные проекты.

5.9. Операторы is и as. Массив указателей. Виртуальные методы

Теоретическая часть: Рассказать, как работают операторы is и as, что такое массивы указателей и виртуальные методы.

Практическая часть: Сделать предложенные проекты.

5.10. Многопоточное программирование. Приложение с графическим интерфейсом

Теоретическая часть: Рассказать, что такое многопоточное программирование и как работать с графическим интерфейсом.

Практическая часть: Сделать предложенные проекты.

6. Android-программирование – 8 часов

6.1. Установка Android Studio и первая программа. Введение в Java Android. Обзор программы и активные состояния. Простейший калькулятор. Работа с полями и Toast. Диалоговые окна и кнопки рейтинга. Переход между строками. – 2 часа

Теоретическая часть: Рассказать о введении в разработку приложений на языке программирования Java в Android-программировании, также установку Android Studio.

Практическая часть: Сделать предложенные проекты.

6.2. Красивый дизайн приложения. Работа с изображениями. Компоненты List View и Seekbar. Фрагменты. Жесты и AutoComplete Text View. Создание звуков в приложении. Сохранение текста в файл. Простой браузер и его создание. Создание файла с расширением apk – 2 часа

Теоретическая часть: Рассказать, как делается красивый дизайн приложения, как работать с изображениями и компонентами, как создаются звуки и вставляются в приложение, как сохранить файл, как создать простой браузер. и файл с расширением apk.

Практическая часть: Сделать предложенные проекты.

6.3. Создание простых мобильных игр на Android на языке программирования Java

Теоретическая часть: Рассказать, как научиться делать простые мобильные игры на Android при помощи языка программирования Java.

Практическая часть: Сделать предложенные проекты.

6.4. Язык программирования Kotlin. Kotlin Android установка и обзор создания мобильных приложений. Создание игры на Kotlin Android – 2 часа.

Теоретическая часть: Рассказать, как работать с языком программирования Kotlin на Android и создавать мобильные приложения.

Практическая часть: Сделать предложенные проекты.

7. Аттестация – 6 часов

7.1. Начальная аттестация (сентябрь) – 2 часа

7.2. Промежуточная аттестация (декабрь) – 2 часа

7.3. Промежуточная аттестация (май) – 2 часа

8. Подготовка к выставкам и конкурсам – 10 часов

9. Итоговое занятие – 2 часа

Теоретическая часть: Подведение итогов работы.

Практическая часть: Просмотр лучших работ учащихся.

**1.4. Календарный учебный график реализации дополнительной
общеобразовательной (общеразвивающей)
программы «Логическое программирование»**

Дата начала занятий	Дата окончания занятий	Количество учебных недель	Количество учебных дней	Количество учебных часов	Режим занятий
сентябрь	май	32	72	144	2 раза в неделю по 2 часа

1.5. Формы аттестации учащихся

Для определения уровня усвоения программы учащимися, ее дальнейшей корректировки и определения путей достижения каждым ребенком максимального творческого и личностного развития предусмотрена *аттестация учащихся*.

Аттестация:

- начальная аттестация (сентябрь);
- промежуточная аттестация (декабрь, май).

При наборе учащихся проводится **начальная аттестация**, в ходе которой педагог проводит *тестирование и практическую работу*, по результатам которого узнает уровень подготовки учащихся к занятиям.

Формы промежуточной аттестации: теоретическая часть - *тестирование*, практическая часть - *практическая работа*.

Тестирование состоит из вопросов по содержанию разделов программы с вариантами ответов. *Практическая работа* предполагает задания по пройденному материалу.

Аттестация	Сроки	Теория	Практика
Начальная	сентябрь	Тестирование	Практическая работа
Промежуточная	декабрь	Тестирование	Практическая Работа
Промежуточная	май	Тестирование	Практическая Работа

2. Комплекс организационно – педагогических условий реализации программы

2.1. Система оценки образовательных результатов

Оценка теоретических знаний и практических умений и навыков учащихся по теории и практике проходит по трем уровням: **высокий, средний, низкий.**

Высокий уровень – учащиеся должны грамотно излагать программный материал, и самостоятельно выполнять практическое задание.

Средний уровень – учащиеся должны уметь правильно, грамотно и по существу излагать программный материал, не допуская существенных неточностей в работе.

Низкий уровень – учащиеся не знают значительной части материала, допускают существенные ошибки, с большими затруднениями выполняют практическое задание.

При обработке результатов учитываются **критерии** для выставления уровней:

Высокий уровень – выполнение 100% - 70%;

Средний уровень – выполнение от 50% до 70%;

Низкий уровень - выполнение менее 50%.

Подведение итогов реализации дополнительной общеобразовательной (общеразвивающей) программы осуществляется в форме конкурсов, участия в тематических выставках различного уровня, где выставляются лучшие видеопроекты.

2.2. Оценочные материалы

Вопросы для начальной аттестации (сентябрь)

Теоретическая часть: тестирование

1. Что такое Photoshop?

- 1) профессиональная программа редактирования фото
- 2) программа создания видеообложек
- 3) программа цифрового скульптинга
- 4) видеомонтажная программа

2. Что такое клавиша NumLock?

- 1) клавиша удаления текста
- 2) клавиша восстановления файла
- 3) клавиша вспомогательной панели введения цифр
- 4) клавиша восстановления компьютера к заводским настройкам

3. Что из себя представляет первоначально любой язык программирования?

- 1) структура кода
- 2) структура текста
- 3) объемное изображение

4. Чем отличается компилятор от IDE?

- 1) переводчик кода
- 2) параллельный интерфейс подключения накопителей (жёстких дисков и оптических приводов) к компьютеру.
- 3) обозначение системы непосредственного впрыска топлива в бензиновых двигателях от компании Renault.
- 4) система программных средств, используемая программистами для разработки программного обеспечения.

5. *Что такое Sata 3?*

- 1) последовательный интерфейс обмена данными с накопителями информации.
- 2) сокет
- 3) ЦПУ
- 4) Один из разделов BIOSa

6. *Что такое Boot-menu?*

- 1) Диспетчер загрузки системы Windows в BIOSe
- 2) название элемента транзистора
- 3) Реле советского лампового телевизора
- 4) название части электрического предохранителя

7. *Чему равен 1 Гб?*

- 1) 8 бит
- 2) 1024 байт
- 3) 1024 Мбайт
- 4) 1000 байт

8. *Чему равен 1 ТБ?*

- 1) 1024 Гб
- 2) 1024 Тб
- 3) 1024 байт
- 4) 8 бит

Практическая часть: практическая работа

Создать афишу в Photoshope.

**Вопросы для промежуточной аттестации
(декабрь)
1 полугодие**

Теоретическая часть: тестирование

1. Что такое Java?

- 1) язык программирования
- 2) музыкальный редактор
- 3) Блок питания

2. Назовите хотя бы одну из библиотек языка Java, связанную с названием племени индейцев?

- 1) Apache Commons
- 2) JUnit
- 3) JHipster

3. Что такое JShell?

- 1) название авиакомпании
- 2) названия автомобильного масла
- 3) оболочка языка

4. Что такое NeatBeans?

- 1) интегрированная среда разработки
- 2) название компании, выпускающая компьютерную технику
- 3) звукозаписывающая компания

5. Какой класс языка программирования Java используют для рисования окружностей

- 1) Ellipse2D.Float
- 2) Arc2D.Float
- 3) PiePanel

6. Как называется сервер для интеграции языка Python с известной игрой Minecraft?

- 1) Nuke
- 2) Neflex
- 3) Spigot

7. Как установить часовой пояс с помощью PHP?

- 1) через `date.timezone`
- 2) через проколы данных
- 3) через регулярные выражения языка.

8. Каким значком можно изобразить оператор присвоения в PHP?

- 1) знаком плюса
- 2) знаком умножения
- 3) знаком равно

9. Каких выражений и операций в PHP нет?

- 1) Логических выражений
- 2) Арифметических операций
- 3) Тригонометрические выражения

10. Что входит в полиморфизм языка программирования C++?

- 1) Наследование и многопоточность
- 2) Диффузия
- 3) Абберация оптической системы

11. С помощью какого класса можно начинать многопоточное программирование в C#?

- 1) класс `thread`.
- 2) блок `finaly`
- 3) инструкция `checked`

12. Что это такое Android Studio?

- 1) программа программирования
- 2) программа цифрового скульптинга
- 3) программа мультипликации

Практическая часть: *практическая работа*

Создать простой одностраничник

**Вопросы для промежуточной аттестации
(май)
2 полугодие**

1. Что такое Kotlin?

- 1) язык программирования
- 2) автомобильная компания
- 3) марка производства компьютеров

2. С помощью какого компонента можно сделать захват видео в Android-программировании?

- 1) MediaRecorder
- 2) android.media.Camera
- 3) AndroidRGraph

3. С помощью какого компонента можно создать графики в Android-программировании?

- 1) AndroidRGraph
- 2) MediaRecorder
- 3) android.media.Camera

4. С помощью какого класса можно воспроизводить аудио и видео в Android-программировании?

- 1) MediaPlayer
- 2) LogCat
- 3) flixel-gdx

5. Что такое String?

- 1) Фреймворк для Java-платформы
- 2) Фреймворк для C++
- 3) Библиотека для Kotlin

6. Что такое Android SDK?

- 1) Трансформатор
- 2) Генератор
- 3) Эмулятор

7. Что такое android.animation?

- 1) Java-пакет классов для создания новой анимации
- 2) Java-пакет классов для управления учетными записями
- 3) Java-пакет набора классов для работы с технологией Bluetooth

8. Что такое Android Development Tools?

- 1) подключаемый модуль Eclipse, который помогает создавать Android-приложения
- 2) подключаемый модуль Android Studio, который помогает создавать Android-приложения
- 3) подключаемый модуль Visual Studio, который помогает создавать Android-приложения

9. Что такое НАХМ в Android Studio?

- 1) механизм аппаратной виртуализации
- 2) поршневой механизм
- 3) механизм турбины

10. Благодаря механизму НАХМ что происходит?

- 1) ускорение эмулятора Android
- 2) ускоряются электроны
- 3) происходит выброс радиоактивного мусора в безвоздушное пространство

Практическая часть: *практическая работа*

Создать простую игру на Android

2.3. Материально-техническое обеспечение

1. Компьютерное оборудование.
2. Программы веб-дизайна.
3. Компиляторы программирования.
4. Проектор.
5. Тетради и ручки для конспекта, флеш-накопители.

2.4. Методическое обеспечение программы

За основу программы «Логическое программирование» взяты такие методы и формы занятий, как *мотивация и стимулирование*, когда на первых занятиях педагог формирует интерес учащихся к обучению, используя при этом:

- словесные, наглядные, аудиовизуальные, практические занятия;
- методы эмоционального стимулирования;
- творческие задания;
- анализ, обобщение, систематизацию полученных знаний и умений;
- выполнение работ под руководством педагога; самостоятельную работу;
- контроль в виде анализа и коррекции работы.

Методы, используемые на занятии характеризуются постепенным смещением акцентов с *репродуктивных на саморазвивающиеся, продуктивные*, с фронтальных на *групповые и индивидуальные*.

Наряду с традиционными *формами организации занятий*, такими, как *рассказ, беседа*, программой предусмотрены и *нетрадиционные*:

- интегрированные занятия, основанные на межпредметных связях;

В основу всех форм учебных занятий заложены *общие характеристики*:

- каждое занятие имеет цель, конкретное содержание, определенные методы организации учебной деятельности;
- построение учебного занятия осуществляется по определенной логике, когда тип занятия соответствует его цели и задачам.

2.5. Информационное обеспечение программы

Интернет-ресурсы:

1. <https://www.helloworld.ru/texts/comp/lang/java/java5/vol1/ch5.html>
2. <https://revall.info/>
3. <https://codernet.ru/books/>
4. <https://obuchalka.org/knigi-po-programmirovaniu/>
5. <https://aldebaran.ru/genre/kompyuteryi/programmirovanie/>

2.6. Список методической литературы

1. Кейденхед, Роджерс Java за 24 часа, 8-е издание.: Пер. с англ. – СПб.: ООО «Диалектика», 2019. – 480 с.: ил.
2. Корнелл, Гари. Java. Библиотека профессионала Расширенные средства, 9-е изд.: Пер. с англ. – М.: ООО "ИД. Вильямс", 2014. – 1008 с.: ил.
3. Шилдт, Герберт. Java 8. Полное руководство; 9-е изд.: Пер. с англ. – М.: ООО "И.Д. Вильямс", 2015. – 1376 с.: ил.
4. Капель Евгений Гогаевич, Фрайман Зэев Java: Задачи по основам программирования. Более 600 задач, около 150 задач с решениями. Книга для школьников ... и не только. М.: ЛЕНАНД, 2019. - 208 с.
5. Лафоре Р. Структуры данных и алгоритмы в Java. Классика Computer Science. 2-е изд. — СПб.: Питер, 2013. — 704 с.: ил.
6. Бауэр К., Кинг Г., Грегори Г. Java Persistence API и Hibernate / пер. с англ. Д. А. Зинкевича; под науч. ред. А.Н.Киселева. – М.: ДМК Пресс, 2017.– 632 с.: ил.
7. Васильев, Алексей Николаевич. Программирование на C# для начинающих. Основные сведения / Алексей Васильев. – М.: Эксмо, 2019. – Т. 1. – 592 с.
8. Васильев, Алексей Николаевич. Программирование на C# для начинающих. Основные сведения / Алексей Васильев. – М.: Эксмо, 2019. – Т. 2. – 592 с.
9. Скин Джош, Гринхол Дэвид. Kotlin. Программирование для профессионалов. – СПб.: Питер, 2020. – 464 с.: ил.
10. Филлипс Б., Стюарт К., Марсикано К. Android-программирование для профессионалов. 3-е изд. – СПб.: Питер, 2019. – 688 с.: ил.
11. Шилдт Герберт. C++ для начинающих. Серия «Шаг за шагом» / Шилдт Г.; пер. с англ. – М.: ЭКОНОМ Паблишерз, 2013. – 640 с.: ил.
12. Котеров, Д.В. PHP 7 / Д.В. Котеров, И.В. Симдянов – СПб.: БХВ-Петербург, 2020 – 1088 с.: ил.
13. Жемеров Д., Исакова С. Kotlin в действии. / пер. с англ. Киселев А. Н. - М.: ДМК Пресс, 2018. - 402 с.: ил.

14. Сомон. П.-И. С61 Волшебство Kotlin / пер. с англ. А. Н. Киселева. – М.: ДМК Пресс, 2020. – 536 с.: ил.
15. Цехнер Марио Программирование игр под Android. — СПб.: Питер, 2013. — 688 с.: ил.
16. Майер Р. Android 4. Программирование приложений для планшетных компьютеров и смартфонов / Рето Майер; [пер. с англ. ООО «Аудио-номикс»]. – М. : Эксмо, 2013. –816 с. – (Мировой компьютерный бестселлер)
17. Дарвин, Ян. Ф. Android. Сборник рецептов: задачи и решения для разработчиков приложений. 2-е изд.: Пер. с англ. – СПб.: ООО Альфа-книга, 2018. – 768 с.: ил.
18. Дейтел П., Дейтел Х., Уолд А. Android для разработчиков. 3-е изд. – СПб.: Питер, 2016. – 512 с.: ил. – (Серия «Библиотека программиста»).
19. Нейл Тереза Мобильная разработка. Галерея шаблонов. – СПб.: Питер, 2013. — 208 с.: ил.
20. Льюис Ш., Данн М.Нативная разработка мобильных приложений / пер. с англ. А. Н. Киселева. – М.: ДМК Пресс, 2020. – 376 с.: ил.
21. Дэрси Л. Android за 24 часа. Программирование приложений под операционную систему Google/ Дэрси Л., Кондер Ш. – М.: Рид Групп, 2011. – 464 с.
22. Бурнет Э. Привет, Android! Разработка мобильных приложений. – СПб.: Питер, 2012. – 256 с.: ил
23. Ёрансон Андерс Эффективное использование потоков в операционной системе Android. / пер. с англ. А.В. Снастина. – М.: ДМК-Пресс, 2015. – 304 с.: ил.
24. Клифтон, Я. Проектирование пользовательского интерфейса в Android / пер. с англ. А.Н. Киселева 2-е изд. – М.: ДМК-Пресс, 2017. – 452 с.: ил.
25. Прохоренок Н. А. OpenCV и Java. Обработка изображений и компьютерное зрение. – СПб.: БХВ-Петербург, 2018. – 320 с.: ил. – (Профессиональное программирование)
26. Шилдт, Герберт. Ш57 Java. Полное руководство, 10-е изд.: Пер. с англ. – СПб. ООО Альфакнига; 2018. – 1488 с. : ил.
27. Фрайман Зев Создание приложений для смартфонов и планшетов под ОС Android: Практический курс. Для школьников и не только. – М.: ЛЕНАНД, 2019. – 504 с.
28. Прохоренок Н. А. Java FX – СПб.: БХВ-Петербург, 2020 – 768с.: ил.
29. Уоллс К. Spring в действии. – М.: ДМК Пресс, 2013. – 752 с.: ил.
30. Лонг Д., Бастани К. Java в облаке. Spring Boot, Spring Cloud, Cloud Foundry. – СПб.: Питер, 2019. – 624 с.: ил.
31. Роберт Седжвик, Кевин Уэйн Алгоритмы на Java / пер. с англ. – М.: ООО И.Д.Вильямс, 2013. – 848 с.: ил.
32. Шефер, Крис, Хо, Кларенс, Харроп, Роб. Spring 4 для профессионалов, 4-е изд.: Пер. с англ. – М. ООО "И.Д. Вильямс", 2015. – 752 с.: ил.

33. Коматинени, Сатия, Маклин, Дэйв. K63 Android 4 для профессионалов. Создание приложений для планшетных компьютеров и смартфонов.: Пер. с англ. – М. : ООО “И.Д. Вильямс”, 2012. – 880 с.: ил.
34. Гутьеррес Фелипе Spring Boot 2: лучшие практики для профессионалов. – СПб.: Питер, 2020. – 464 с.: ил. – (Серия «Библиотека программиста»).
35. Шилдт, Герберт. С# 4.0: полное руководство.: Пер. с англ. – М.: ООО "И.Д. Вильямс", 2011. – 1056 с.: ил.
36. Бонд Джереми Гибсон Unity и С#. Геймдев от идеи до реализации. 2-е изд. – СПб.: Питер, 2019. – 928 с.: ил. – (Серия «Для профессионалов»)
37. Макеффри, Митч Unreal Engine VR для разработчиков / Митч Макеффри [пер. с англ. Н.И. Веселко, О.В. Максименковой, А.А. Незнанова] – М.: ЭКСМО, 2019. – 256 с. – (Мировой компьютерный бестселлер)
38. Колисниченко Д. Н. Программирование для Android 5. Самоучитель. – СПб.: БХВ-Петербург, 2015. – 303 с.: ил.
39. Голощапов А. Л. Google Android. Создание приложений для смартфонов и планшетных ПК. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2014. – 928 с.: ил.
40. Колисниченко Д. Н. Android для пользователя. Полезные программы и советы. – СПб.: БХВ-Петербург, 2013. – 256 с.: ил.

**Календарный план воспитательной работы
детского объединения «Логическое программирование»
на 2023-2024 учебный год**

Педагог дополнительного образования – Пронькин А.В.

№	Мероприятие	Задачи	Сроки проведения
1.	Организационное родительское собрание	Знакомство родителей с целями и задачами обучения по данной ДООП, особенностями организации учебного процесса, режимом работы и учебным графиком занятий в объединении.	сентябрь
2.	Индивидуальные консультации для родителей	Решение вопросов социального и педагогического характера.	в течение учебного года
3.	Проведение инструктажа по действиям в случае террористического акта	Формирование социальной компетентности.	сентябрь
4.	Онлайн выставка «Открытка любимому учителю» , посвященная Всемирному дню учителя и Году педагога и наставника	Воспитание чувства уважения к педагогической профессии, развитие духовно-нравственных качеств личности (уважения и доброжелательности, взаимопомощи), совершенствование творческих умений и навыков работы в коллективе.	октябрь
5.	Выставка компьютерной видеографики «Осенняя пора, очей очарованье...» .	Развитие эстетических качеств личности, творческой активности обучающихся, трудолюбия, обеспечение возможности самореализации и развития талантов.	октябрь
6.	Мастер-класс по созданию баннеров	Развитие эстетических качеств личности, творческой активности обучающихся, трудолюбия, самореализация и совершенствование своих умений и навыков в работе с IT-технологиями	ноябрь
7.	Творческая мастерская в техники «коллаж» при изготовлении новогодней открытки	Содействие развитию творческой активности обучающихся, самореализация и совершенствование своих умений и навыков, воспитание чувства уважения и вежливого отношения к своим родным и близким.	декабрь

8.	Творческая мастерская при использовании ИТ-технологий «Подарки ко Дню защитника Отечества»	Формирование у детей общероссийской гражданской идентичности, патриотизма, гражданской ответственности, чувства гордости за историю России, воспитание чувства уважения к родным и близким.	февраль
9.	Мастерская «Открытки к 8 Марта» при использовании ИТ-технологий	Формирование у детей чувства любви и уважения к матери, развитие эстетических качеств личности, творческой активности обучающихся, трудолюбия, развитие эмоциональной отзывчивости, чувства гордости и заботливого отношения к маме.	март
10.	Викторина «Что ты знаешь о космосе?»	Развитие интеллектуальных способностей детей, воспитание умения работать в команде, сопереживать друг другу; создание эмоциональной благоприятной атмосферы.	апрель
11.	Мастер-класс по изготовлению открыток по созданию видеосюжета ко Дню Победы	Формирование у детей общероссийской гражданской идентичности, патриотизма, гражданской ответственности, чувства гордости за историю России, воспитание чувства уважения к родным и близким.	май
12.	Итоговое родительское собрание	Подведение итогов работы объединения, знакомство с результатами итоговой аттестации обучающихся	май
13.	Участие в муниципальных, региональных, Всероссийских и Международных конкурсах изобразительного искусства и декоративно-прикладного творчества	Выявление, поддержка и развитие способностей и талантов у обучающихся совершенствование творческих умений и навыков.	В течение учебного года

