


МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ НИЖЕГОРОДСКОЙ ОБЛАСТИ  
Государственное бюджетное профессиональное образовательное учреждение  
«НИЖЕГОРОДСКИЙ РАДИОТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

СОГЛАСОВАНО:  
Директор МАОУ № 186  
«Авторская академическая школа»

  
М.В.Буров



УТВЕРЖДАЮ:  
Директор ГБПОУ «НРТК»

  
И.А.Кормщикова  
Приказ от 31.08.2023г. № 96-ИТ/О



СОГЛАСОВАНО:  
Педагогический совет  
Протокол № 1 от 30.08.2023г.

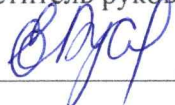
**ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ  
ПРОГРАММА  
«Работа с системой контроля версий»  
(базовый уровень)**

Направленность – техническая  
Возраст обучающихся: 12-17 лет  
Срок реализации программы: 85 часов

г.Нижний Новгород  
2023г.

Внутренняя экспертиза проведена. Программа рекомендована к рассмотрению на педагогическом совете учреждения.

Заместитель руководителя по ИМР

 /Русинова Е.Г.

«31» августа 2023 г.

Дополнительная общеразвивающая программа соответствует действующим федеральным, региональным нормативным документам Российской Федерации и локальным нормативным актам ГБПОУ «НРТК»

### Информационная карта

Полное наименование программы	Дополнительная общеразвивающая программа «Работа с системой контроля версий» (базовый уровень)
Направленность	техническая
Общий объем программы	85 часов
Целевая категория обучающихся	обучающиеся 12 – 17 лет
Аннотация программы	Программа направлена на изучение обучающимися системы контроля версий Git, а также на закрепление и углубление знаний и развитие навыков программирования на языке Java. Программа также направлена на развитие интереса обучающихся к проектной деятельности. Обучение по программе позволяет получить практические навыки и знания, необходимые для организации деятельности по разработке итоговой проектной работы по направлению «Программирование на Java».
Планируемые результаты реализации программы	Обучающиеся изучат особенности работы с системами контроля версий с использованием командной строки, а также с помощью среды разработки IntelliJ IDEA Community Edition. Обучающиеся смогут закрепить навыки программирования на языке Java, научатся писать сложный программный код для создания собственных приложений. Также обучающиеся смогут сформировать навыки критического и творческого мышления, навыки проектной деятельности в процессе работы над собственным итоговым проектом.
Авторы-составители	– педагог дополнительного образования Григорян Н.М.

## 1. Комплекс основных характеристик программы

### 1.1. Пояснительная записка

В современном мире большую популярность среди специалистов IT-компаний набирают программные инструменты, помогающие командам разработчиков управлять изменениями в исходном коде с течением времени. В свете усложнения сред разработки, а также необходимостью работать над кодом постоянно и в команде, большинство крупных компаний используют системы контроля версий (version control system – VCS). Помимо этого, данный инструмент полезен и обучающимся, которые уже имеют твердую базу знаний по направлению «Программирование на Java» и работают над объемными проектами с большим количеством кода. Поэтому данная программа направлена на ознакомление обучающихся с этим инструментом, получение первичных навыков работы с репозиторием, что поможет им в будущей работе с программным кодом.

Дополнительная общеразвивающая программа «Работа с системой контроля версий» (базовый уровень) реализуется посредством сетевого взаимодействия между ГБПОУ «Нижегородский радиотехнический колледж» и MAOY № 186 «Авторская академическая школа».

**Актуальность программы** обосновывается необходимостью изучения полезных инструментов работы с кодом, а также популяризацией языка программирования Java среди обучающихся, развитием мотивации к изучению IT-технологий с целью их будущего профессионального определения в сфере информационных технологий и искусственного интеллекта.

**Направленность программы.** Программа обладает технической направленностью и предназначена для использования в системе дополнительного образования детей. Обучающиеся в ходе занятий приобщаются к знаниям в области информационных технологий, формируют логическое мышление, развивают навыки программирования на языке Java, формируют навыки работы с репозиторием.

**Адресат программы.** Дополнительная программа «Работа с системой контроля версий» (базовый уровень) предназначена для детей в возрасте 12 – 17 лет, без ограничений возможности здоровья. Содержание программы составлено с учётом возрастных и психологических особенностей детей данного возраста.

**Форма проведения занятий.** По форме организации деятельности предполагаются групповые формы занятий, количество обучающихся в группе 7-17 человек.

**Место проведения занятий:** г. Нижний Новгород, ул. Генкиной, 84.

**Режим занятий.** Занятия проводятся 2 раза в неделю по 2-3 академических часа с перерывом на 10 минут для проветривания помещения и отдыха обучающихся. Продолжительность одного занятия – 40 минут.

**Условия реализации программы.** Зачисление детей производится по результатам освоения дополнительных общеразвивающих программ базового и продвинутого уровней по направлению «Программирование на Java».

**Срок освоения общеразвивающей программы** определяется содержанием программы и составляет 4 месяца (85 часов).

**Формы обучения:** очная, с применением дистанционных образовательных технологий (Закон №273-ФЗ, гл.2, ст.17, п.2.).

**Виды занятий:** беседы, мастер-классы, кейсы, деловые игры, практикумы, групповые занятия, самостоятельная работа, проект, консультация.

**Уровень освоения программы.** По уровню освоения программа общеразвивающая, базовая. Это обеспечивает возможность обучения с любым уровнем подготовки.

### 1.2. Цели и задачи программы

**Цель программы** – изучение инструментов работы с системой контроля версий GitHub, а также развитие мотивации к дальнейшему изучению языка программирования Java.

Для успешной реализации поставленной цели необходимо решить ряд поставленных задач:

**Обучающие (предметные):**

- ознакомить обучающихся с системами контроля версий;
- научить устанавливать Git на компьютер, работать в командной строке GitBash;
- изучить жизненный цикл файлов в репозитории, игнорирование изменений;

- изучить локальные и удаленные репозитории, подключение к ним через среду разработки IntelliJ IDEA Community Edition.
- научить работать с Git в команде с использованием веток;
- научить работать с версиями программного кода.

**Развивающие (метапредметные):**

- сформировать навыки критического и творческого мышления при решении задач, практических кейсов, выполнении индивидуальных и/или групповых заданий;
- развить познавательную активность, мотивацию к самостоятельному изучению дополнительных материалов по темам модулей;
- развить внимание, память, логику и концентрацию при работе над проектами;
- сформировать навыки проектной деятельности, в том числе коммуникативные навыки в процессе работы в команде;
- сформировать навык анализа промежуточных результатов деятельности, работы над ошибками, подбора разных способов улучшения эффективности своей деятельности.

**Воспитательные (личностные):**

- сформировать дисциплину и ответственность за свои действия и результаты деятельности;
- развить трудолюбие и уважение к результатам своего труда и труда окружающих;
- сформировать навык самоорганизации, самоконтроля и рефлексии;
- воспитать мотивацию к творчеству, созданию продуктов интеллектуальной собственности, инновационных разработок;
- воспитать чувство коллективизма, взаимопомощи и толерантности.

## 2. Содержание дополнительной общеразвивающей программы

### 2.1. Учебный план

№ п/п	Наименование раздела/модуля	Количество часов			Формы промежуточного контроля
		Всего	Теория	Практика	
1	Вводное занятие. Входной контроль.	5	2	3	Тестирование
2	Модуль 1. Система контроля версий Git.	25	11	14	Тестирование
3	Модуль 2. Проектная деятельность.	52	4	48	-
4	Итоговая аттестация	3	0	3	Защита проектов
<b>Итого:</b>		<b>85</b>	<b>17</b>	<b>68</b>	

## 2.2. Содержание учебного плана

№ п/п	Наименование разделов и тем	Содержание учебного материала, лабораторные и практические работы, самостоятельная работа обучающихся, проект	Количество часов			Основные виды деятельности обучающихся, оборудование
			Всего	Теория	Практика	
1.	Вводное занятие.	<b>Теория</b>	2	2	0	Беседа, ответы на вопросы. <i>Оборудование:</i> моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
		Инструктаж по технике безопасности. Обзор образовательной программы. Краткое повторение основных разделов базовой и продвинутой программ по направлению «Программирование на Java»				
2.	Входной контроль.	<b>Практика</b>	3	0	3	Тестирование, самостоятельная работа с инструментами среды IntelliJ IDEA. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство.
		Выполнение тестовых и практических заданий по темам базовой и продвинутой программ по направлению «Программирование на Java»				
<b>Модуль 1. Система контроля версий Git</b>						
3.	Знакомство с Git.	<b>Теория</b>	2	2	0	Лекция, наблюдение за работой педагога, ответы на вопросы. <i>Оборудование:</i> моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
		Что такое система контроля версий и зачем она нужна. Как выглядит работа с Git. Правила работы с Git. Частые проблемы и их решения.				
4.	Работа с Git в командной строке и в среде разработки.	<b>Теория</b>	3	1	2	Лекция, наблюдение за работой педагога, ответы на вопросы, самостоятельная работа с Git. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
		Работа с Git в командной строке. Жизненный цикл файлов в репозитории. Игнорирование изменений.				
		<b>Практика</b>				
		Установка Git на компьютеры. Выполнение практической работы по теме «Работа с Git в командной строке и в среде разработки».				

5.	Работа с удаленным репозиторием.	<b>Теория</b>	5	2	3	Лекция, наблюдение за работой педагога, ответы на вопросы, самостоятельная работа с инструментами среды IntelliJ IDEA и с репозиторием на GitHub. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
		Удаленный и локальные репозитории. Обзор систем GitLab и GitHub. Создание удаленного репозитория. Создание SSH-ключа.				
6.	Работа с локальным репозиторием.	<b>Теория</b>	5	2	3	Лекция, наблюдение за работой педагога, ответы на вопросы, самостоятельная работа с инструментами среды IntelliJ IDEA и с репозиторием на GitHub. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
		Обновление локального репозитория. Разрешение конфликтов при обновлении репозитория.				
7.	Командная работа в Git.	<b>Теория</b>	5	2	3	Лекция, наблюдение за работой педагога, ответы на вопросы, самостоятельная работа с инструментами среды IntelliJ IDEA и с репозиторием на GitHub. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
		Работа с удаленными ветками. Откладывание изменений. Слияние веток. Разрешение конфликтов. Модели ветвления. Pull request (запрос на слияние)				
8.	Сравнение версий и отмена изменений	<b>Теория</b>	5	2	3	Лекция, наблюдение за работой педагога, ответы на вопросы, самостоятельная работа с инструментами среды IntelliJ IDEA и с репозиторием на GitHub.
		Выполнение практической работы по теме «Работа с удаленным репозиторием».				
6.	Работа с локальным репозиторием.	<b>Теория</b>	5	2	3	Лекция, наблюдение за работой педагога, ответы на вопросы, самостоятельная работа с инструментами среды IntelliJ IDEA и с репозиторием на GitHub. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
		Обновление локального репозитория. Разрешение конфликтов при обновлении репозитория.				
7.	Командная работа в Git.	<b>Теория</b>	5	2	3	Лекция, наблюдение за работой педагога, ответы на вопросы, самостоятельная работа с инструментами среды IntelliJ IDEA и с репозиторием на GitHub. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
		Работа с удаленными ветками. Откладывание изменений. Слияние веток. Разрешение конфликтов. Модели ветвления. Pull request (запрос на слияние)				
8.	Сравнение версий и отмена изменений	<b>Теория</b>	5	2	3	Лекция, наблюдение за работой педагога, ответы на вопросы, самостоятельная работа с инструментами среды IntelliJ IDEA и с репозиторием на GitHub.
		Просмотр изменений. Удаление незакоммиченных изменений. Отмена закоммиченных изменений. Сброс коммитов. Отмена слияний, удаление веток.				

		<b>Практика</b> Выполнение практической работы по теме «Сравнение версий и отмена изменений».				<i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
<b>Модуль 2. Проектная деятельность</b>						
9.	Планирование проекта	<b>Теория</b> Понятие проекта и проектной деятельности. Виды проектов. Этапы работы над проектом. Понятия темы, цели, задач. Принципы формирования команды проекта, основные характеристики. <b>Практика</b> Выбор темы, постановка целей и задач, планирование проекта. Определение средств реализации проекта, способов контроля сроков и промежуточных результатов работы над проектом.	7	2	5	Беседа, работа в проектных группах, составление интеллект-карты. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
10.	Реализация проекта.	<b>Практика</b> Самостоятельная работа по реализации поставленных задач, поэтапное выполнение запланированной работы над проектом. Контроль промежуточных результатов, оценка эффективности действий, контроль сроков выполнения проекта.	38	0	38	Самостоятельная работа с инструментами среды IntelliJ IDEA и с репозиторием на GitHub по созданию собственного приложения, консультирование обучающихся. <i>Оборудование:</i> ноутбуки; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
11.	Оформление проекта.	<b>Практика</b> Оформление индивидуального проекта согласно техническим требованиям и критериям оценивания презентации.	5	0	5	Самостоятельная работа по оформлению проекта, консультирование обучающихся. <i>Оборудование:</i> ноутбуки; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
12.	Подготовка к защите.	<b>Теория</b> Общие рекомендации для подготовки к выступлению. Техника подготовки. Психологический аспект готовности к выступлению. Культура выступления.	2	2	0	Беседа, ответы на вопросы. <i>Оборудование:</i> моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.



13.	<b>Итоговая аттестация.</b>	Демонстрация проектов, рефлексия.	3	0	3	Защита проектов. <i>Оборудование:</i> ноутбуки; моноблочное интерактивное устройство; доска магнитно-маркерная настенная; доска-флипчарт магнитно-маркерная.
<b>Итого</b>			<b>85</b>	<b>17</b>	<b>68</b>	

### 2.3. Календарный учебный график

Номер учебной недели/ группа	сентябрь			октябрь					ноябрь				декабрь				
	04.09.2023-10.09.2023	11.09.2023-17.09.2023	18.09.2023-24.09.2023	25.09.2023-01.10.2023	02.10.2023-08.10.2023	09.10.2023-15.10.2023	16.10.2023-22.10.2023	23.10.2023-29.10.2023	30.10.2023-05.11.2023	06.11.2023-12.11.2023	13.11.2023-19.11.2023	20.11.2023-26.11.2023	27.11.2023-03.12.2023	04.12.2023-10.12.2023	11.12.2023-17.12.2023	18.12.2023-24.12.2023	25.12.2023-31.12.2023
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
<b>ПД-1</b>	У	У	У	У	У	П	У	У	У	У	У	У	У	У	У	У	И

Считать нерабочими праздничными днями: 4 ноября.

Условные обозначения:

- У – учебные занятия,
- П – промежуточный контроль,
- И – итоговая аттестация.

## 2.4. Планируемые результаты обучения

### *Обучающие (предметные):*

- обучающиеся ознакомились с системами контроля версий;
- обучающиеся научились устанавливать Git на компьютер, работать в командной строке GitBash;
- обучающиеся изучили жизненный цикл файлов в репозитории, игнорирование изменений;
- обучающиеся изучили локальные и удаленные репозитории, умеют подключаться к ним через среду разработки IntelliJ IDEA Community Edition;
- обучающиеся научились работать с Git в команде с использованием веток;
- обучающиеся умеют работать с версиями программного кода.

### *Развивающие (метапредметные):*

- сформированы навыки критического и творческого мышления при решении задач, практических кейсов, выполнении индивидуальных и/или групповых заданий;
- развита познавательная активность, обучающиеся самостоятельно изучают дополнительные материалы по темам модулей;
- развиты внимание, память, логика и концентрация при работе над проектами;
- сформированы навыки проектной деятельности, в том числе коммуникативные навыки в процессе работы в команде;
- сформирован навык анализа промежуточных результатов деятельности, работы над ошибками, подбора разных способов улучшения эффективности своей деятельности.

### *Воспитательные (личностные):*

- сформирована дисциплина и ответственность за свои действия и результаты деятельности;
- развиты трудолюбие и уважение к результатам своего труда и труда окружающих;
- сформирован навык самоорганизации, самоконтроля и рефлексии;
- сформирован интерес к творчеству, созданию продуктов интеллектуальной собственности, инновационных разработок;
- сформированы чувство коллективизма, взаимопомощи и толерантности.

## 2.5. Определение результативности реализации программы

### **Формы аттестации**

Система отслеживания результатов обучающихся выстроена следующим образом:

- текущий контроль;
- промежуточный контроль;
- итоговый контроль.

*Текущий контроль* осуществляется путём наблюдения, определения качества выполнения практических работ, отслеживания динамики развития обучающегося в рамках учебных занятий. Способы проверки уровня освоения тем: опрос, выполнение упражнений, наблюдение, оценка выполненных самостоятельных работ.

*Промежуточный контроль* осуществляется в конце модуля с целью определения уровня усвоения материалов модуля, оценить совокупность знаний и умений обучающихся с помощью тестирования.

*Итоговый контроль* проводится в виде общей рефлексии, анализа результатов прохождения обучающимися программы и итоговой аттестации. Итоговая аттестация осуществляется в форме защиты проекта, демонстрации работы созданного приложения и оценивается по 100-балльной шкале, которая переводится в один из уровней освоения образовательной программы (Таблица 1).

*Таблица 1*

### **Критерии оценивания уровня усвоения программы**

<b>Баллы, набранные обучающимся</b>	<b>Уровень освоения</b>	<b>Результат</b>
0–30 баллов	низкий	Обучающиеся демонстрируют низкий уровень заинтересованности в учебной, проектной деятельности, составляющих содержание программы. Итоговый проект не соответствует требованиям, содержит ошибки в коде, что делает код написанного приложения нефункциональным.

31–70 баллов	средний	Обучающиеся демонстрируют достаточную заинтересованность в учебной, проектной деятельности, составляющих содержание программы. Итоговый проект соответствует требованиям, код написанного приложения работает, но требует незначительной доработки.
71–100 баллов	высокий	Обучающиеся демонстрируют высокую заинтересованность в учебной, проектной деятельности, составляющих содержание программы. Итоговый проект соответствует требованиям, код написанного приложения работает и не требует доработки. Итоговый проект содержит реализацию собственной разработки, творческой идеи.

Итоговый проект оценивается специально формируемой комиссией. Состав комиссии (не менее 3-х человек): педагог (в обязательном порядке), представитель администрации учебной организации, приветствуется привлечение IT-профессионалов, представителей высших и других учебных заведений.

Критерии оценивания итоговых работ разрабатываются педагогом в зависимости от содержания программы, ее целей и задач. Критерии оценивания итоговых работ обучающихся по программе «Работа с системой контроля версий» (базовый уровень) представлены в таблице 2:

Таблица 2

**Критерии оценивания итоговых работ**

№ п/п	Критерий	Баллы
<b>Оценка итогового проекта</b>		
1.	Функциональность приложения	0 – приложение не работает, или работает некорректно; 5 – приложение работает, но требуются значительные доработки; 10 – приложение работает, требуются незначительные доработки, не влияющие на функциональность кода; 15 – приложение работает, написан функциональный код.
2.	Аккуратность и читабельность программного кода (Приложение 1)	0 – программный код нечитабельный, присутствует более 7 нарушений правил написания; 5 – программный код неаккуратный, присутствует от 4 до 6 нарушений правил написания; 10 – программный код читабельный, но присутствуют незначительные нарушения правил написания (1-3 нарушения); 15 – программный код написан аккуратно, правила написания читабельного кода не нарушены.
3.	Именованые проектов, классов, методов и переменных (Приложение 2)	0 – присутствует более 4 различных нарушений правил именования проектов, классов, методов и переменных; 5 – присутствует 2-3 различных нарушения правил именования проектов, классов, методов и переменных; 10 – присутствуют один тип нарушения правил именования проектов, классов, методов и переменных; 15 – полностью соблюдены правила именования проектов, классов, методов и переменных.
4.	Комментарии в программном коде	0 – большое количество излишних комментариев, которые мешают чтению кода; в программе есть закоментированные фрагменты кода, которые не используются; комментарии дублируют содержание кода; 5 – комментарии написаны четко и кратко; комментарии содержат важную для понимания кода дополнительную информацию.

<b>Оценка демонстрации итогового проекта</b>		
1.	Целеполагание, планирование проектной деятельности	<p>0 – цель не сформулирована;</p> <p>3 – цель сформулирована нечетко, отсутствует план ее достижения;</p> <p>7 – цель сформулирована ясно, отсутствует план ее достижения;</p> <p>10 – цель сформулирована ясно, присутствует подробный план ее достижения.</p>
2.	Глубина раскрытия содержания темы проекта	<p>0 – тема проекта не раскрыта</p> <p>3 – тема проекта раскрыта фрагментарно, присутствует отклонение от выбранной темы;</p> <p>7 – тема проекта раскрыта, автор показал достойный уровень знаний по выбранной теме;</p> <p>10 – тема проекта раскрыта, автор продемонстрировал глубокие знания по выбранной теме.</p>
3.	Презентация итогового проекта	<p>0 – отсутствует презентация;</p> <p>3 – презентация есть, но отсутствует четкая структура, допущены ошибки в оформлении;</p> <p>7 – презентация обладает четкой структурой, присутствуют незначительные ошибки в оформлении;</p> <p>10 – презентация обладает четкой структурой, отличается грамотным оформлением.</p>
4.	Оценка выступления	<p>0 – автор читает текст, слабо владеет материалом, не умеет вести дискуссию и не может ответить на вопросы членов комиссии;</p> <p>3 – автор самостоятельно представляет свою итоговую работу, но присутствуют речевые ошибки, слабо владеет материалом, испытывает затруднения в умении вести дискуссию и отвечать на вопросы членов комиссии;</p> <p>7 – автор самостоятельно представляет свою итоговую работу, но присутствуют незначительные речевые ошибки, владеет материалом на достойном уровне, умеет вести дискуссию и отвечать на вопросы членов комиссии;</p> <p>10 – автор самостоятельно представляет свою итоговую работу, отсутствуют речевые ошибки, владеет материалом на достойном уровне, умеет вести дискуссию, доказательно и корректно защищать свою идею и отвечать на вопросы членов комиссии;</p>
5.	Рефлексия	<p>0 – отсутствуют выводы по результатам проделанной работы;</p> <p>3 – выводы по работе представлены неполно;</p> <p>7 – выводы по работе представлены полно, присутствует анализ сложностей в ходе работы над проектом;</p> <p>10 – выводы по работе представлены полно, присутствует анализ сложностей в ходе работы над проектом и предложен хотя бы один способ их преодоления в рамках работы над будущими проектами.</p>

### 3. Комплекс организационно-педагогических условий реализации дополнительной общеразвивающей программы

#### 3.1. Материально-техническое обеспечение

##### *Требования к помещению:*

- помещение для занятий, отвечающее требованиям СанПин для учреждений дополнительного образования;
- качественное освещение;
- столы, стулья по количеству обучающихся и 1 рабочее место для педагога.

##### *Общее оборудование:*

- ноутбук тип 1 – 1 шт.;
- ноутбук тип 2 – 14 шт.

##### *Презентационное оборудование:*

- моноблочное интерактивное устройство SMART MX265-V2 – 1 шт.;
- доска магнитно-маркерная настенная с полимерным покрытием (ДО-17б) – 1 шт.;
- доска-флипчарт магнитно-маркерная 70 x 100 см BRAUBERG Стандарт – 1 шт.;
- напольная мобильная стойка для интерактивных досок DIGIS DSM-P1060CL – 1 шт.

##### *Дополнительное оборудование:*

- роутер TP-Link Archer C6U – 1 шт.;
- web-камера A4Tech PK-940HA – 1 шт.;
- ультрафиолетовый рециркулятор Milerd DZR-1 – 1 шт.;
- тележка для хранения и зарядки 20 ноутбуков – 1 шт.;
- лазерный цветной принтер HP Color LaserJet Pro M255dw (7KW64A) – 1 шт.;
- огнетушитель ОУ-3 (ВСЕ)

##### *Материалы:*

- подставка под огнетушитель П-15 – 1 шт.;
- комплект переходников (Набор Rexant 18-1203 USB 6 переходников+удлинитель) – 1 шт.;
- компьютерная мышь Logitech Mouse M100USB, проводная, черная – 15 шт.;
- наушники SVEN AP-860V – 14 шт.;
- кресло для ученика Нота (экокожа, цвет бежевый, крестовина «Паук», глайдеры) – 14 шт.

##### *Мебель:*

- кресло для учителя СН-883 (экокожа, цвет Слоновая кость, крестовина хром) – 1 шт.;
- шкаф (Размер(Ш\*Г\*В) 500мм\*415мм\*2000мм.Корпус и полки ЛДСП-16мм цвет: серый шиншилла – 1 шт.;
- стол преподавателя с тумбой, левый – 1 шт.;
- стол ученический тип 1 – 14 шт.

##### *Программное обеспечение:*

- среда разработки IntelliJ IDEA;
- система контроля версий GitHub.

#### 3.2. Методические материалы

Основной тип занятий — *комбинированный*, сочетающий в себе элементы теории и практики. Большинство заданий курса выполняется самостоятельно с помощью персонального компьютера и необходимых программных средств. Единицей учебного процесса является раздел, каждый из них охватывает отдельную информационную технологию или её часть. С учётом регулярного повторения ранее изученных тем темп изучения отдельных разделов блока определяется субъективными и объективными факторами. Каждая тема курса начинается с описание конкретных задач, определяющих содержание материал, которое предстоит изучить. С этой целью педагог проводит демонстрацию презентации или показывает саму программу, а также готовые работы, выполненные в ней. Закрепление знаний проводится с помощью практики отработки умений самостоятельно решать поставленные задачи, соответствующих минимальному уровню планируемых результатов обучения. Основные задания являются обязательными для выполнения всеми обучающимися в группе. Задания выполняются на компьютере с использованием интегрированной среды разработки IntelliJ IDEA Community Edition. При этом обучающиеся не только формируют теоретические практические знания, но и приобретают новые технологические навыки.

Методика обучения ориентирована на *индивидуальный* подход. Для того чтобы каждый обучающийся получил наилучший результат обучения, программой предусмотрены индивидуальные задания для самостоятельной работы. Такая форма организации обучения стимулирует интерес обучающегося к предмету, активность и самостоятельность учащихся, способствует объективному контролю глубины и широты знаний, повышению качества усвоения материала обучающимися, позволяет педагогу получить объективную оценку выбранной им тактики и стратегии работы, методики индивидуального обучения и обучения в группе, выбора предметного содержания.

В рамках учебных занятий больше часов отводится на самостоятельную работу обучающихся, реализуемую посредством выполнения практических заданий, решения кейсов. Это позволяет укрепить знания по пройденной теме и развить навык программирования и работы в среде разработки. Более того, это способствует реализации индивидуального подхода к обучению, поскольку каждый обучающийся решает задачу или кейс в своем темпе. Для обучающихся, которые быстро справляются с решением практической работы, у педагога всегда есть дополнительный материал повышенного уровня сложности для дополнительной практики.

В ходе обучения проводится проверочные задания, которые позволяют оценить уровень остаточных знаний обучающихся, а также способствует активизации учебно-познавательной деятельности, что служит индикатором успешности образовательного процесса.

Реализация программы предполагает использование *здоровьесберегающих* технологий. Специальных медицинских противопоказаний к занятиям не существует, но при выборе данного объединения родителям и педагогу необходимо с особым вниманием отнестись к обучающимся, относящимся к группе риска по зрению, так как на занятиях значительное время они пользуются компьютером. Педагогами проводится предварительная беседа с родителями, в которой акцентируется внимание на обозначенную проблему.

*Здоровьесберегающая* деятельность реализуется:

- созданием безопасных материально-технических условий;
- включением в занятие динамических пауз, периодической смены деятельности обучающихся;
- выполнение комплекса упражнений для релаксации глаз, а также увеличения остроты зрения (Приложение 3);
- контролем педагога за соблюдением обучающимися правил работы за персональным компьютером;
- созданием благоприятного психологического климата в учебной группе в целом.

### Список использованной литературы и интернет-ресурсы

#### *Основная литература*

1. Блинов И.Н., Романчик В.С.. Java 2 Практическое руководство. // Мн.: Универсал Пресс, 2005 – 400 с.
2. Блох Д. Java. Эффективное программирование. // М.: Лори, 2002 – 224 с.
3. Макконнелл С. Совершенный код.// СПб: Питер, 2005 – 868 с.
4. Седжвик Роберт, Уэйн Кевин. Алгоритмы на Java., пер. с англ. – 4-е изд. - М.: Вильямс, 2013 – 848 с.
5. Скиена С.С., Ревилла М.А. Олимпиадные задачи по программированию. Руководство по подготовке к соревнованиям. Пер. с англ. – М.: КУДИЦ-ОБРАЗ, 2005 – 416 с.
6. Сухов С. А. Учебное пособие Основы программирования на Java. Ульяновск: Ул ГТУ, 2006 – 88 с.
7. Сьерра К., Бейтс Б. Изучаем Java. изд. Эксмо, 2020 – 720 с.
8. Харди Б., Филлипс Б. Android. Программирование для профессионалов. 2-е изд. – СПб.: Питер, 2016 – 640 с.
9. Шилдт Герберт. Полный справочник по Java. пер. с англ. – 7-е изд. – М.: Вильямс, 2007 – 1040 с.
10. Эккель Б. Философия Java. 4-е изд. СПб.: Питер, 2009 – 637с.

### *Интернет-источники*

1. Интегрированная среда разработки с использованием Java «Online Java Compiler»: <https://www.jdoodle.com/online-java-compiler/>
2. Интерактивная среда разработки на языке Java «Browxy»: <https://www.browxy.com/>
3. Веб-сервис <https://github.com/>

### *Литература, рекомендованная для учащихся и родителей*

1. Арсак Ж. Программирование игр и головоломок. — М.: Наука, 1990 — 224 с.
2. Бентли Д. Жемчужины творчества программистов: пер. с англ. — М.: Радио и связь, 1990 — 224 с.
3. Окулов С.М., Пестов А.А. 100 задач по информатике. Киров: Изд-во ВГПУ, 2000 — 272с.
4. Уэзерелл Ч. Этюды для программистов. — М.: Мир, 1982 — 288 с.

### Правила написания аккуратного и читабельного программного кода:

1. Определение переменных нужно располагать в начале блока, а не «ждать» первого использования переменной. Инициализация должна производиться, по возможности, сразу.
2. Между именем метода и скобками для списка параметров нет пробела.
3. Параметры разделяются пробелом.
4. Пробелы окружают любой оператор.
5. Ключевое слово и следующая за ним скобка ( должны разделяться пробелом.
6. Открывающаяся фигурная скобка располагается на той же строке, что и сигнатура метода/заголовок if, while-блока и т.д.
7. Закрывающаяся фигурная скобка выровнена по строке начала данного блока.
8. Методы разделяются пустой строкой, объявления свойств класса располагаются по одному на строку.
9. На строке располагается только один оператор.

### Правила именования проектов, классов, методов и переменных:

1. Имена проектам, классам, методам и переменным даются на английском языке.
2. Имена классов должны быть существительными, первые буквы всех слов — заглавные.
3. Названия методов должны быть глаголами, отражать свойства метода, первая буква должна быть строчной, первые буквы внутренних слов — заглавные.
4. Имена переменных должны начинаться со строчной буквы, внутренние слова — с заглавной.
5. Имена проектов, классов, методов и переменных должны отражать их содержание (их названия должны быть понятны для любого разработчика, который будет читать код).

