

Государственное автономное нетиповое образовательное учреждение
Свердловской области «Дворец молодёжи»
Центр цифрового образования детей «IT-куб г. Верхняя Пышма»

Принята на заседании
научно-методического совета
ГАНОУ СО «Дворец молодёжи»
Протокол № 4 от 03.06.2021 г.

УТВЕРЖДАЮ:
Директор
ГАНОУ СО «Дворец молодёжи»
А. Н. Слизько
Приказ № 464-д от 04.06.2021 г.

Дополнительная общеобразовательная общеразвивающая программа
технической направленности

«Мобильная разработка»

Возраст обучающихся: 13–17 лет
Срок реализации: 1 год

СОГЛАСОВАНО:
Начальник центра цифрового образования
детей «IT-куб г. Верхняя Пышма»
Томшин М.С.
«19» мая 2021 г.

Авторы-составители:
Шанин М.М., педагог
дополнительного образования;
Куролина Т.Ю., методист;
Соболева М.В., методист

г. Екатеринбург, 2021 г.

I. Комплекс основных характеристик программы

1. Пояснительная записка

Задача инновационного развития программного обеспечения требует соответствующей образовательной среды, в том числе создания оптимальных условий детского технического творчества. Одной из наиболее инновационных областей в сфере детского технического творчества является мобильная разработка.

Для дальнейшего развития мобильных приложений существует широкий выбор направлений разработки. Каждому ребёнку интересно, как устроена платформа Android, как работает приложение на его смартфоне.

Изучение языка программирования Java по данной программе обучения даёт возможность пользователю мобильного устройства с ОС Android создавать простейшие программы в среде разработки, взаимодействующие с элементами графики, аудио- и видеофайлами, тестовыми форматами.

Java, в частности, используется для написания и загрузки программ на Arduino-совместимые платы. Arduino – интегрированная среда разработки (IDE) – кросс-платформенное приложение для ОС Windows, MacOS, Linux. С помощью Arduino можно конструировать различные интерактивные, обучающие, экспериментальные, развлекательные модели и устройства. Основой интерфейса является язык C++, поэтому освоить инструментарий могут даже начинающие программисты – обучающиеся по программе «Мобильная разработка».

Программа «Мобильная разработка» имеет техническую **направленность**, ориентирована на развитие навыков программирования и проектирования программ на языке программирования Java.

Основанием для проектирования и реализации данной общеразвивающей программы служит **перечень следующих нормативных правовых актов и государственных программных документов**: Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 N 273-ФЗ; Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р; Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242 «О направлении

Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»; Распоряжение правительства РФ от 04.09.2014 № 1726-р «Об утверждении Концепции развития дополнительного образования детей»; «Основы законодательства РФ об охране здоровья граждан», утвержденные Верховным советом РФ от 22.07.1993 № 5487 - (ред. от 25.11.2009); Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребёнка в РФ»; Федеральный закон от 21.11.2011 №323-ФЗ «Об основах охраны здоровья граждан в Российской Федерации»; Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Приказ №1008 отменен).

Актуальность программы обусловлена потребностью общества в технически грамотных специалистах и полностью отвечает социальному заказу по подготовке квалифицированных кадров в области мобильной разработки и машинного обучения, а также обусловлена высоким интересом подростков к ИТ-сфере, интернет-коммуникации, что характеризуется их активностью в информационном пространстве, стремлении к программированию, изучению устройства компьютерных программ и Интернет-ресурсов не на уровне обычного пользователя, а на уровне профессионала в данной сфере.

В процессе занятий по выбранной образовательной программе, обучающиеся будут постоянно сталкиваться с решением актуальных интересных задач, требующих творческого подхода и самостоятельности в принятии решений. Все полученные обучающимися на занятиях знания, умения и практические навыки подготовят их к самостоятельной проектно-исследовательской деятельности с применением современных технологий.

Также программа актуальна тем, что не имеет аналогов на рынке общеобразовательных услуг и является своего рода уникальным образовательным продуктом в области информационных технологий. Ведь данная программа является единственным в своём роде экспериментом в связи с востребованностью на рынке и отсутствием программ образования в данном направлении для

школьников. Особенность программы «Мобильная разработка» – в изучении основ языка программирования Java. Она строится в доступной и понятной для обучающихся среде, т. е. программирование ведётся в текстово-графическом режиме, что позволяет сразу задавать необходимый функционал для элементной базы приложения.

Прогностичность программы «Мобильная разработка» заключается в том, что она отражает требования и актуальные тенденции не только сегодняшнего, но и завтрашнего дня, а также имеет междисциплинарный характер, что полностью отражает современные тенденции построения как дополнительных общеобразовательных программ, так и образования в целом.

Также данная программа является базой для перехода на более сложные программы обучения. Так, по итогам успешного освоения программы «Мобильная разработка», обучающийся может сдать вступительное тестирование и быть зачислен на общеразвивающую программу «Мобильная разработка «IT школа Samsung» (продвинутый уровень)», которая представляет собой более углублённое и профессионально ориентированное изучение уже освоенного материала в рамках программы.

Обучающиеся приобретают знания по основам IT, которые будут востребованы для дальнейшего обучения в профильных средних специальных и высших учебных заведениях.

Отличительная особенность дополнительной общеразвивающей программы «Мобильная разработка» в том, что она является модульной, что предполагает наличие системы средств и приёмов, с помощью которых достигается интегрирующая дидактическая цель в совокупности всех модулей конкретной учебной дисциплины.

Модуль – структурная единица образовательной программы, имеющая определённую логическую завершённость по отношению к результатам обучения. (*Словарь рабочих терминов по предпрофильной подготовке*). Каждый модуль состоит из кейсов (не менее двух), направленных на формирование определённых компетенций (hard и soft). Результатом каждого кейса является «продукт»

(групповой, индивидуальный), демонстрирующий сформированность компетенций.

Кейс – история, описывающая реальную ситуацию, которая требует проведения анализа, выработки и принятия обоснованных решений (*Высшая школа экономики*).

Кейс включает в себя набор специально разработанных учебно-методических материалов. Кейсовые «продукты» могут быть самостоятельным проектом по результатам освоения модуля или общего проекта по результатам всей образовательной программы.

Модули и кейсы различаются по сложности и реализуются по принципу «от простого к сложному».

Для возрастной категории 15–16 лет при решении кейсов ставятся задания повышенного уровня сложности и применяется оборудование для соответствующей возрастной категории.

Адресат общеразвивающей программы

Дополнительная общеразвивающая программа «Мобильная разработка» предназначена для детей в возрасте 13–17 лет, без ограничений возможностей здоровья. Группы формируются по возрасту: 13–14 и 15–17 лет. Формы занятий групповые, количество обучающихся в группе до 14 человек. Состав групп постоянный.

Место проведения занятий: г. Верхняя Пышма, пр. Успенский, 2Г.

Возрастные особенности группы

Содержание программы учитывает возрастные и психологические особенности детей в возрасте 13–17 лет, указанные в ДООП и определяющие выбор форм проведения занятий с обучающимися. Выделенные нами возрастные периоды при формировании групп 13-17 лет базируются на психологических особенностях развития старшего подросткового возраста.

Содержание программы учитывает возрастные и психологические особенности детей 13–17 лет, которые определяют выбор форм проведения занятий с обучающимися. Дети этого возраста отличаются эмоциональностью, стремлением к активной практической деятельности, поэтому основной формой

проведения занятий выбраны практические занятия. Ребят также увлекает совместная, коллективная деятельность, так как резко возрастает значение коллектива, общественного мнения, отношений со сверстниками, оценки поступков и действий ребёнка со стороны не только старших, но и сверстников. Ребёнок стремится завоевать в их глазах авторитет, занять достойное место в коллективе. Поэтому в программу включены практические занятия соревновательного характера, которые позволяют каждому проявить себя и найти своё место в детском коллективе.

Также следует отметить, что дети данной возрастной группы характеризуются такими психическими процессами, как изменение структуры личности и возникновение интереса к ней, развитие абстрактных форм мышления, становление более осознанного и целенаправленного характера деятельности, проявление стремления к самостоятельности и независимости, формирование самооценки. Эти процессы позволяют положить начало формированию начального профессионального самоопределения обучающихся.

Режим занятий, объём общеразвивающей программы: длительность одного занятия составляет 2 академических часа, периодичность занятий – 2 раза в неделю.

Срок освоения общеразвивающей программы определяется содержанием программы и составляет 1 год (144 часа).

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

Виды занятий: беседы, обсуждения, мультимедийные презентации, игровые формы работы, практические занятия, метод проектов. Также программа курса включает групповые и индивидуальные формы работы обучающихся (в зависимости от темы занятия).

Учебный процесс строится таким образом, чтобы экспериментальная и практическая работа преобладала над теоретической подготовкой. Необходимые для работы теоретические сведения находятся на каждом персональном компьютере в специальной папке, даются педагогом перед началом практических занятий. Индивидуальная работа проводится во время практических занятий – при

выполнении задания у каждого обучающегося возникают свои вопросы. Групповая работа проводится во время теоретических занятий. Каждая тема по программированию сопровождается наглядной демонстрацией работы алгоритма для того, чтобы обучающиеся представляли работоспособность алгоритма, а также к чему им нужно стремиться при выполнении поставленной задачи. Учебный процесс организуется на основе постепенного усложнения учебного материала, как теоретического, так и практического.

Программой предусмотрены следующие виды деятельности обучающихся:

- освоение теоретического и практического материала на занятиях;
- разработка индивидуального проекта;
- участие в вебинарах;
- промежуточная аттестация в форме электронного тестирования;
- самостоятельная практическая работа: выполнение домашних заданий, мини-проектов (небольшие приложения, которые реализуются обучающимися преимущественно на занятиях совместно с педагогом с небольшими самостоятельными доработками в качестве домашнего задания).

По типу организации взаимодействия педагогов с обучающимися при реализации программы используются личностно ориентированные технологии, технологии сотрудничества.

Реализация программы предполагает использование здоровьесберегающих технологий.

Здоровьесберегающая деятельность реализуется:

- через создание безопасных материально-технических условий;
- включением в занятие динамических пауз, периодической смены деятельности обучающихся;
- контролем соблюдения обучающимися правил работы на ПК;
- через создание благоприятного психологического климата в учебной группе в целом.

Объём общеразвивающей программы составляет 144 часа.

Форма организации образовательной деятельности – групповая.

По уровню освоения программа общеразвивающая, *разноуровневая* (стартовый, базовый уровни). Она обеспечивает возможность обучения детей с любым уровнем подготовки.

«Стартовый уровень» рассчитан на детей в возрасте с 13 лет, проявляющих интерес к аналитической и исследовательской деятельности, ИТ-технологиям, приобретению навыков программирования.

Зачисление детей на первый год обучения производится без предварительного отбора (свободный набор). К концу стартового уровня обучающиеся приобретут навыки поиска, анализа и использования информации, а также безопасного поведения в сети Интернет; получат навыки программирования в среде разработки IntelliJ IDEA IDE на языке Java.

«Базовый уровень» рассчитан на детей, проявляющих интерес к созданию программ на языке Java для решения прикладных задач, желающих совершенствовать свои навыки программирования, имеющих опыт программирования в различных интегрированных средах разработки на языке Java.

Зачисление детей на базовый уровень после завершения стартового уровня производится по результатам успешной сдачи итогового тестирования.

Обучение по программе «Мобильная разработка» нацелено на углубление и структурирование знаний основ современного языка программирования Java; умение использовать данную технологию для решения сложных и актуальных практических задач; работать со специальными средствами и библиотеками языка Java.

К концу обучения подростки способны самостоятельно определять задачи и пути решения; владеют базовыми навыками программирования на языке Java; способны разрабатывать простые программы для решения практических задач на основе изученного языка программирования Java; способны самостоятельно изучать новые технологии. Зачисление детей на первый год обучения производится без предварительного отбора (свободный набор).

Педагогическая целесообразность программы

Программа «Мобильная разработка» составлена в виде модулей, позволяющих получить детям необходимый объём знаний в зависимости от уровня подготовки и потребности.

Стартовый уровень предполагает использование и реализацию общедоступных и универсальных форм организации материала, минимальную сложность предлагаемого для освоения содержания программы.

Базовый уровень предполагает использование и реализацию таких форм организации материала, которые допускают освоение специализированных знаний и языка, гарантированно обеспечивают трансляцию общей и целостной картины в рамках содержательно-тематического направления программы.

После успешного освоения программы «Мобильная разработка», обучающийся будет владеть необходимыми знаниями языка программирования Java на уровне, достаточном для успешного прохождения отборочного тестирования и зачисления на дополнительную общеобразовательную общеразвивающую программу «Мобильная разработка» «IT-школа Samsung».

2. Цель и задачи общеразвивающей программы

Цель программы – формирование технической грамотности средствами приобщения обучающихся к разработке программ под современную платформу Java.

Для успешной реализации поставленной цели необходимо решить ряд педагогических, развивающих и воспитательных *задач*:

Обучающие:

- расширение знаний о современных и популярных платформах;
- обучение языку программирования Java;
- обучение программированию технических устройств.

Развивающие:

- формирование алгоритмического мышления; навыков работы с информацией;
- формирование умения самостоятельно решать поставленную задачу, излагать мысли в чёткой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путём логических рассуждений;
- развитие логического и технического мышления.

Воспитательные:

- формирование активной жизненной позиции, гражданско-патриотической ответственности;
- воспитание этики групповой работы и отношений делового сотрудничества, взаимоуважения, развитие основ коммуникативных отношений внутри проектных групп и в коллективе в целом;
- создание условий для развития устойчивой потребности в самообразовании;
- воспитание упорства в достижении результата, целеустремленности, организованности;
- пропаганда здорового образа жизни;
- формирование ответственного отношения к труду, толерантности и уважительного отношения к окружающим.

3. Содержание общеразвивающей программы

Учебный план Стартовый уровень

Таблица 1

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации / контроля
		Теоретические занятия	Практические занятия	Всего	
1	Модуль 1. Логика	2	8	10	
1.1	«Событие». Понятие, группа	1	1	2	
1.2	Бинарная логика. Бинарные операторы	1	3	4	
1.3	Практика. Решение задач	0	4	4	
2	Модуль 2. Математика. Система исчислений	3	9	12	
2.1	Арифметика в информатике. DEC. СИ десятичная	1	1	2	
2.2	BIN. СИ двоичная. Bit. Byte	1	3	4	
2.3	HEX. СИ шестнадцатеричная	1	1	2	
2.4	Практика. Решение задач	0	2	2	
2.5	Итоговое занятие по разделу	0	2	2	Контр. работа
3.	Модуль 3. Java. IntelliJ IDEA IDE	25	33	58	
3.1	Техника безопасности. Среда разработки. Hello, world!	1	1	2	
3.2	System.in/out. Тип String	1	3	4	
3.3	Примитивные типы	5	5	10	
3.3.1	Представление целых чисел: прямой код, код со сдвигом, дополнительный код. byte, short, int, long	1	1	2	
3.3.2	Представление вещественных чисел. float, double	1	1	2	
3.3.3	Представление символов, таблицы кодировок. char	1	1	2	
3.3.4	Арифметика числовая и символьная	2	2	4	

3.4	Операции отношения. Логические операции	2	4	6	
3.5	Условные конструкции. Тип Boolean	2	2	4	
3.6	Итеративные конструкции. Массивы	2	4	6	
3.7	Функции. Синтаксис. Виды. Рекурсия	2	2	4	
3.8	Практика. Решение задач	0	2	2	
3.9	Кейс "Конвертер BIN<->DEC<->HEX"	8	10	18	
3.9.1	Постановка проблемы	2	0	2	
3.9.2	Идеи и формирование образа проекта	2	2	4	
3.9.3	Макетирование проекта и деление на функциональные подзадачи	2	2	4	
3.9.4	Прототипирование и тестирование. Подготовка презентации	2	6	8	
3.10	Защита/Итоговое занятие по разделу	2	0	2	Защита проекта. /Итоговое тестирование по стартовому уровню
	Итого стартовый уровень:	30	50	80	

Базовый уровень

Таблица 2

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации / контроля
		Теоретические занятия	Практические занятия	Всего	
4	Модуль 4. Java. Arduino IDE	22	42	64	
4.1	Техника безопасности. Среда разработки. "Hello, world!"	1	1	2	
4.2	Знакомство с ATmega328. Регистры. Примитивные типы. Арифметика. uintX t, X = 8.16.32.64	2	4	6	
4.3	Операции отношения. Логические операции. Работа со светодиодами.	1	3	4	
4.4	Условные конструкции. Работа с кнопками	2	4	6	
4.5	Итеративные конструкции. Массивы. Led-матрица	2	4	6	

4.6	Метод конечных автоматов	2	6	8	
4.7	Практика. Решение задач	0	4	4	
4.8	Кейс "Сейф"	10	16	26	
4.8.1	Постановка проблемы.	2	2	4	
4.8.2	Идеи и формирование образа проекта	2	2	4	
4.8.3	Макетирование проекта и деление на функциональные подзадачи	2	6	8	
4.8.4	Прототипирование и тестирование. Подготовка презентации	4	6	10	
4.9	Защита/Итоговое занятие по разделу	2	0	2	Защита проекта Итоговое тестирование по базовому уровню
Итого базовый уровень:		22	42	64	
Итого всего:		52	92	144	

Содержание учебно-тематического плана

Стартовый уровень

Модуль 1. Логика

Тема 1.1. 2 часа. «Событие». Понятие, группа

Тема 1.2. 4 часа. Бинарная логика. Бинарные операторы

Тема 1.3. 4 часа. Практика. Решение задач

Тема 1.1. «Событие». Понятие, группа

Теория: Понятия. Определение, виды понятий. Отношения между понятиями.

Практика: Импликация (если..., то...), рассмотреть отношения между понятиями по типу:

Если А, то Б

Б в том случае, если А

При А будет Б

Из А следует Б

В случае А произойдет Б

Б, так как А

Б, потому что А

А – достаточное условие для Б

Б – необходимое условие для А

Множество. Дать определение, показать группировку элементов во множестве.

Тема 1.2. Бинарная логика. Бинарные операторы

Теория: Знакомство с операторами и их применением, логическими операциями.

Операции с множествами (пересечение, дополнение, разность).

Пересечение множеств:

$$A \cup B := \{x \mid x \in A \vee x \in B\}$$

Объединение множеств:

$$A \setminus B := A \cap \overline{B} = \{x \mid x \in A \wedge x \notin B\}$$

Логическая операция И (конъюнкция).

Логическая операция ИЛИ (дизъюнкция).

Логическая операция НЕ (отрицание, унарный оператор).

Логическая операция «строгая дизъюнкция» (Исключающее ИЛИ).

Практика: Продемонстрировать выполнение логических операций.

Объяснение результатов. Описать или предоставить таблицу используемых символов.

Тема 1.3. Практика. Решение задач

Практика:

Задача 1.3.1. Петя, Вася и Маша остались дома одни. Кто-то из них ел варенье. На вопрос мамы, кто это сделал, они сказали: а) Петя: «Я не ел. Маша тоже не ела». б) Вася: «Маша действительно не ела. Это сделал Петя». в) Маша: «Вася врёт. Это он съел». + Выясните, кто ел варенье, если известно, что двое из них оба раза сказали правду, а третий один раз соврал, а один раз сказал правду.

Задача 1.3.2. В нарушении правил обмена валюты подозреваются четыре работника банка – А, В, С, Д. Известно, что:

1. Если А нарушил, то и В нарушил правила обмена валюты.
2. Если В нарушил, то и С нарушил или А не нарушал.
3. Если Д не нарушил, то А нарушил, а С не нарушил.
4. Если Д нарушил, то и А нарушил.

Кто из подозреваемых нарушил правила обмена валюты?

Задача 1.3.3. Формализуйте также ответ сына: «Если я буду говорить правду, то боги будут любить меня. Если я буду лгать, то люди будут любить меня. Но я должен говорить правду или лгать. Значит, меня будут любить боги или меня будут любить люди».

Задача 1.3.4. Три девочки – Роза, Маргарита и Анюта представили на конкурс цветоводов корзины выращенных ими роз, маргариток и анютиных глазок. Девочка, вырастившая маргаритки, обратила внимание Розы на то, что ни у одной из девочек имя не совпадает с названием любимых цветов. Какие цветы вырастила каждая из девочек?

Модуль 2. Арифметика

Тема 2.1. 2 часа. Арифметика в информатике. DEC. СИ десятичная

Тема 2.2. 4 часа. BIN. СИ двоичная. Bit. Byte

Тема 2.3. 2 часа. HEX. СИ шестнадцатеричная

Тема 2.4. 2 часа. Практика. Решение задач

Тема 2.5. 2 часа. Итоговое занятие по разделу

Тема 2.1. Арифметика в информатике. DEC. СИ десятичная

Теория: Формирование представлений о различных системах счисления, об операциях перевода между системами счисления и поисках основания системы счисления.

Практика:

Упражнение 2.1.1. Записать числа в различных системах счисления.

Необходимые знания: представление о том, что такое число, система счисления (нумерация), позиционная (однородная и смешанная) и непозиционная система счисления, основание системы счисления (нижний индекс), арифметические операции над цифрами.

Виды систем счисления. Что такое непозиционная система счисления. Что такое позиционная (однородная и смешанная) система счисления. Таблица соответствия чисел, записанных в различных системах счисления.

Правила записи чисел в системах счисления.

Знакомство с десятичной системой, её основанием и операциями перевода в другие системы счисления.

Необходимые знания: характеристики системы счисления, нерациональность использования в ЭВМ, иллюстрация работы позиционной системы счисления.

Упражнение 2.2.1. Сравнить число в непозиционной системе счисления позиционной десятичной. Возьмем число 503 в непозиционной и позиционной системе счисления. Каждую цифру числа необходимо умножить на основание системы, в данном случае число «10», возведённое в степень, равную номеру разряда. Получается, значение равно $5*10^2 + 0*10^1 + 3*10^0 = 500+0+3 = 503$.

Тема 2.2. BIN. СИ двоичная. Bit. Byte

Теория: знакомство с двоичной системой, её основанием и операциями перевода в другие системы счисления.

Необходимые знания: характеристики системы счисления, правила записи чисел, использование в ЭВМ (триггер, регистр, оперативная память, кодовая таблица символов (Unicode), единицы хранения информации), операции перевода в другие системы счисления, арифметические операции (сложение, вычитание, умножение, деление), двоичное кодирование информации.

Перевод двоичного числа в десятичное. При переводе предоставить таблицу степеней двойки.

Практика:

Упражнение 2.2.1. Перевод чисел из двоичной системы в шестнадцатеричную и наоборот.

Упражнение 2.2.2. Перевести число в двоичную систему счисления.

Упражнение 2.2.3. Перевести число в шестнадцатеричную систему счисления.

Кодовые таблицы ASCII. Форматы файлов. Определение кода символа. Ввод символа по коду. Назначение и особенности кодовых таблиц.

Упражнение 2.2.4. Определять числовой код символа. Кодировать и декодировать сообщение по кодовой таблице.

Тема 2.3. HEX. СИ шестнадцатеричная

Теория: Знакомство с шестнадцатеричной системой счисления, её основанием и операциями перевода в другие системы счисления.

Необходимые знания: характеристики системы счисления, правила записи чисел, операции перевода в другие системы счисления, использование для обозначения цветов.

Описать или предоставить таблицу используемых символов.

Перевод чисел из шестнадцатеричной системы в десятичную.

Практика:

Упражнение 2.3.1. Перевести число в десятичную систему счисления.

Тема 2.4. Практика. Решение задач

Практика:

Задание 2.4.1. Дано $A=101011002$, $B=101010102$. Какое из чисел C , записанных в двоичной системе, отвечает условию $A < c < B$?

- 1) 101011002
- 2) 101010102
- 3) 101010112
- 4) 101010002

Задание 2.4.2. Вычислите сумму чисел X и Y , если $X=1101112$, $Y=1358$.

Результат представьте в двоичном виде.

- 1) 110101002
- 2) 101001002
- 3) 100100112
- 4) 100101002

Задание 2.4.3. Найдите среднее арифметическое чисел 2368, 6C16 и 1110102.

Ответ представьте в десятичной системе счисления.

Задание 2.4.4. В саду 100 q фруктовых деревьев: из них 33 q яблони, 22 q груши, 16 q слив и 17 q вишнен. Найдите основание q системы счисления, в которой посчитаны деревья.

Задание 2.4.5. Найдите основание x системы счисления, если известно, что $2002_x = 13010$.

Тема 2.5. Итоговое занятие по разделу

Практика: Контрольная работа по теме.

Модуль 3. Java. IntelliJ IDEA IDE

Тема 3.1. 2 часа. Техника безопасности. Среда разработки. "Hello, world!"

Тема 3.2. 4 часа. System.in/out. Тип String

Тема 3.3. 10 часов. Примитивные типы

Тема 3.3.1. 2 часа. Представление целых чисел: прямой код, код со сдвигом, дополнительный код. byte, short, int, long

Тема 3.3.2. 2 часа. Представление вещественных чисел. float, double

Тема 3.3.3. 2 часа. Представление символов, таблицы кодировок. char

Тема 3.3.4. 4 часа. Арифметика числовая и символьная

Тема 3.4. 6 часов. Операции отношения. Логические операции

Тема 3.5. 4 часа. Условные конструкции. Тип Boolean

Тема 3.6. 6 часов. Итеративные конструкции. Массивы

Тема 3.7. 4 часа. Функции. Синтаксис. Виды. Рекурсия.

Тема 3.8. 2 часа. Практика. Решение задач

Тема 3.9. 18 часов. Кейс "Конвертер BIN<->DEC<->HEX"

Тема 3.9.1. 2 часа. Постановка проблемы

Тема 3.9.2. 4 часа. Идеи и формирование образа проекта

Тема 3.9.3. 4 часа. Макетирование проекта и деление на функциональные подзадачи

Тема 3.9.4. 8 часов. Прототипирование и тестирование. Подготовка презентации

Тема 3.10. 2 часа. Защита / Итоговое занятие по разделу

Тема 3.1. Техника безопасности. Среда разработки. Hello, world!

Теория: Техника безопасности. Первичная настройка среды разработки IntelliJ IDEA IDE.

Практика: Запуск первой программы "Hello, world!". Знакомство с внутренней структурой программы. Синтаксис языка Java. Разбор примеров.

Тема 3.2. System.in/out. Tun String

Теория: Знакомство с системой ввода / вывода среды разработки. Работа с консолью. Тип данных String. Работа с переменными. Чтение данных из консоли. Объект типа Scanner. Работа с System.in, String. Вывод данных в консоль. Работа с System.out, String. Методы toString(), print(), println(), printf().

Практика:

Упражнение 3.2.1. Программа-попугай. Вывод данных в систему вывода, прочитанных из системы ввода.

Тема 3.3. Примитивные типы

Тема 3.3.1. Представление целых чисел: прямой код, код со сдвигом, дополнительный код. byte, short, int, long

Теория: Знакомство с целыми числами и их использованием в информатике.

Прямой код. Достоинства и недостатки представления чисел с помощью прямого кода.

Код со сдвигом. Достоинства и недостатки представления чисел с помощью кода со сдвигом.

Дополнительный код (дополнение до единицы). Достоинства и недостатки представления чисел с помощью кода с дополнением до единицы.

Дополнительный код (дополнение до двух). Длинная арифметика для чисел, представленных с помощью кода с дополнением до двух.

Достоинства и недостатки представления чисел с помощью кода с дополнением до двух.

Практика: Знакомство с различными типами кодов, разбор примеров.

Тема 3.3.2. Представление вещественных чисел. float, double

Теория: Знакомство с вещественными числами и их использованием в информатике. Нормальная и нормализованная форма. Типы чисел с плавающей точкой (по IEEE 754). Число одинарной точности (Binary32, Single precision, float). Число двойной точности (Binary64, Double precision, double). Диапазон значений чисел с плавающей запятой. Особые значения чисел с плавающей точкой: Ноль (со знаком), Неопределенность (NaN), Бесконечности, Денормализованные числа. Алгоритм получения представления вещественного числа в памяти ЭВМ.

Практика: Знакомство с вещественными числами, разбор примеров.

Тема 3.3.3. Представление символов, таблицы кодировок. char

Теория: Знакомство с символами и их использованием в информатике. Представление символов в вычислительных машинах. Таблицы кодировок.

Кодировки стандарта ASCII. Структурные свойства таблицы.

Кодировки стандарта UNICODE. Кодовое пространство. Модифицирующие символы. Способы представления.

UTF-8. Принцип кодирования. Правила записи кода одного символа в UTF-8. Определение длины кода в UTF-8

UTF-16. UTF-16LE и UTF-16BE

UTF-32. Порядок байт. Варианты записи. Порядок от старшего к младшему. Порядок от младшего к старшему. Переключаемый порядок. Смешанный порядок. Различия. Маркер последовательности байт.

Проблемы Юникода.

Практика: Знакомство с символами и таблицами кодировок, разбор примеров.

Тема 3.3.4. Арифметика числовая и символьная

Теория: Знакомство с особенностями при использовании арифметических операторов с различными типами данных.

Практика: Разбор примеров по изучаемой теме.

Тема 3.4. Операции отношения. Логические операции

Теория: Рассмотреть операторы и их классификацию, поразрядные операции, логические выражения. Тип данных boolean. Логические значения true и false. Несовместимость типа boolean с int. Отметить, что приведение логических значений к целым и наоборот невозможно.

Логические операции и операции отношения. Операторы отношения: $>$, $<$, $>=$, $<=$, $!=$, $==$. Уяснение понятия значения операции отношения как ИСТИННО или ЛОЖНО. Логические операции: логическое И, логическое ИЛИ, логическое НЕ. Тернарная операция ? :

Выражения и операции. По итогу изучения различных операций рассмотрение понятия выражения в языке программирования; знаки операций; знаки-разделители. Классификация операций по количеству operandов: унарные и бинарные. Классификация операций по типу: арифметические, логические, присваивания, отношения и др.

Практика:

Упражнение 3.4.1. Программа, демонстрирующая выполнение логических операций и операций отношений. Объяснение результатов её выполнения.

Задание 3.4.1. Задачи на «ручное» написание логических выражений средствами языка Java:

- x лежит вне отрезка $[a, b]$;
- x принадлежит отрезку $[a, b]$ или отрезку $[c, d]$;

- x лежит вне отрезков $[a, b]$ и $[c, d]$;
- целое a является нечётным числом;
- целое a является трёхзначным числом, кратным пяти;
- из чисел a, b, c меньшим является c , а большим b ;
- среди чисел a, b, c, d есть взаимно противоположные;
- среди целых чисел a, b, c есть хотя бы два чётных;
- из отрезков с длинами a, b, c можно построить треугольник;
- год, заданный числом a , является високосным;
- год, заданный числом a , не является високосным;
- число a является простым;
- среди целых чисел a, b, c есть хотя бы два нечётных;
- отрезки длиной a, b и c могут образовать прямоугольный треугольник.

Тема 3.5. Условные конструкции. Тип Boolean

Теория: Изучить внутреннюю логику работы условных конструкций; приобрести навыки их использования в различных формах, предусмотренных синтаксисом языка. Закрепить навыки написания всех ранее изученных операторов путем написания и вычисления выражений.

Область действия блоков. Фигурные скобки для выделения блока. Вложенность блоков. На данный момент рассмотреть только ограничение на объявление переменных с одинаковым именем в одном и том же или вложенных блоках.

Конструкция if-else. Синтаксис оператора:

`if (cond_expression) TRUE_statement`

Или

`if (cond_expression) TRUE_statement
else FALSE_statement`

Разъяснить, что `statement` – это только один оператор или блок. Фундаментальное правило для сложных ветвлений, реализуемых с помощью вложенных конструкций if-else: `else` относится к ближайшему `if`, не имеющему `else`.

Конструкция switch-case. Синтаксис. Что может быть в качестве метки case. Мотивировка использования конструкции как упрощение сложных ветвлений. Логика выполнения, объяснение роли ключевых слов break и default в конструкции switch-case.

Практика:

Упражнение 3.5.1. Небольшие фрагменты кода, иллюстрирующие использование операторов ветвления, приоритетов вычисления операторов в выражении. Ускоренное вычисление логических выражений – прекращение вычислений, когда результат уже ясен.

Задание 3.5.1. Написать собственный пример на использование операторов ветвления. Например: нахождение максимума, минимума среди нескольких введенных переменных.

Тема 3.6. Итеративные конструкции. Массивы

Теория: Изучить внутреннюю логику работы итеративных конструкций; приобрести навыки их использования в различных формах, предусмотренных синтаксисом языка.

Цикл с предусловием while. Синтаксис. Объяснение логики работы, пример использования.

Цикл с постусловием do-while. Синтаксис. Объяснение логики работы, пример использования. Уяснение ключевого отличия от цикла while с предусловием: цикл с постусловием выполняется хотя бы один раз.

Операторы прерывания логики управления программой. Безусловные операторы перехода break, continue.

Практика:

Упражнение 3.6.1. Небольшие фрагменты кода, иллюстрирующие использование операторов цикла (без использования массивов). Например, вычисление НОД по алгоритму Евклида.

Задание 3.6.1. Написать собственный пример на использование операторов цикла и операторов безусловного перехода. Например, проверка числа на то, что оно является простым.

Теория: Оператор for, for each, одномерные массивы. Определение массива как совокупности элементов одного и того же типа, расположенных вплотную друг за другом в памяти. Объявление массива двумя способами. Подчеркнуть необходимость создания массива с помощью new. Значения, которыми инициализируется массив по умолчанию при создании. Инициализация массива без new – инициализация массива при объявлении. Доступ к отдельным элементам массива. Определение количества элементов в массиве через свойство length.

Цикл for. Синтаксис. Логика работы, роль каждой из составных частей. Частные формы записи оператора for: отсутствует инкрементальное выражение; отсутствует инкрементальное выражение и начальное выражение. Уяснение связи между for и while, эквивалентная запись for через while. Примеры некорректного использования операторов цикла, приводящего к зацикливанию. Вложенные циклы for.

Цикл for each. Синтаксис. Преимущества его применения при работе с массивами в сравнении с обычным for. Отметить, что переменная в цикле for each перебирает не индексы массива, а сами элементы массива.

Практика:

Упражнение 3.6.2. Фрагменты кода, иллюстрирующие на одномерном массиве решение задач нахождения максимального, минимального.

Задание 3.6.2. Написать программу по обработке массива с выводом на экран полученного результата:

- поиск заданного элемента простым перебором;
- переворот массива «задом наперед» без использования вспомогательного массива;
- вычисление суммы элементов массива;
- нахождение самого часто повторяющегося числа среди элементов массива;
- нахождение среднего арифметического числа элементов массива;
- заполнить массив числами арифметической прогрессии по заданной преподавателем формуле.

Тема 3.7. Функции. Синтаксис. Виды. Рекурсия

Теория: Фундаментальное понятие функции в программировании и проектировании программного обеспечения на примере методов Java, приобрести навыки их использования. Рассмотреть видимость переменных.

Основные понятия. Определение функции как логически самостоятельной именованной части программы, которой могут передаваться параметры, и которая может возвращать какое-то значение.

Определение функции. На примере объяснить понятия: тело метода, тип возвращаемого значения. Список формальных аргументов, список фактических аргументов. Методы с типом void и методы с пустым списком аргументов.

Практика:

Упражнение 3.7.1. На примере продемонстрировать ситуации, когда функции необходимы. Реализовать собственную функцию для считывания и вывода массива (int[] readIntArray(int length) и void printArray(int[] a, char delimiter)) с использованием уже существующих функций.

Область видимости переменных. Обзорная классификация переменных по области видимости: область класса, область метода, область блока.

Упражнение 3.7.2. На примере разобрать программы с ошибками, связанными с видимостью переменных.

Тема 3.8. Практика. Решение задач

Тема 3.9. Кейс "Конвертер BIN<->DEC<->HEX"

Тема 3.9.1. Постановка проблемы

Теория: Дается проблема и ряд ограничений по ней. Отталкиваясь от этих ограничений, нужно прийти к вектору развития идеи.

Задание: выполнение практического задания по теме кейса.

Компетенции:

Hard Skills – сбор данных, обработка информации.

Soft Skills – командная работа, коммуникативность, критическое мышление, аналитическое мышление, логическое мышление, исследовательские навыки,

способность выслушать и принять мнение собеседников, умение высказать и отстоять свою точку зрения.

Тема 3.9.2. Идеи и формирование образа проекта.

Теория: Опираясь на знание методов формирования идей и двигаясь по выбранному вектору развития, сгенерировать концепцию будущего проекта.

Практика: Выполнение практического задания по теме кейса.

Soft Skills – командная работа, коммуникативность, критическое мышление, аналитическое мышление, логическое мышление, способность выслушать и принять мнение собеседников, умение высказать и отстоять свою точку зрения.

Тема 3.9.3. Макетирование проекта и деление на функциональные подзадачи.

Теория: Основываясь на сформулированной ранее концепции решения проблемы, создать макет, демонстрирующий набор функций по решению проблемы. Разделение всей концепции на подзадачи с последующим выделением функций каждой части на макете и нахождение ее реализации. Построение плана работы над проектом «от простого, к сложному». Рефлексия.

Практика: Выполнение практического задания по теме кейса.

Hard Skills – работа с формообразованием, макетирование, объемно-пространственное мышление, интернет-серфинг, планирование рабочего процесса.

Soft Skills – командная работа, коммуникативность, аналитическое мышление, фокусировка, декомпозиция идей, логическое мышление, внимание, скрупулезность.

Тема 3.9.4. Прототипирование и тестирование. Подготовка презентации.

Теория: Следуя плану работ, заменить макетные части на реальные. По ходу продвижения по плану фиксировать на контрольных точках, какие функции удалось реализовать и в какой мере. Создание плана презентации. Формулировка тезисов. Верстка презентации. Рефлексия.

Практика: Выполнение практического задания по теме кейса.

Hard Skills – Работа с языком программирования Java, работа над прототипом, работа с планом презентации, работа с графическими редакторами, работа с фото/видео, работа с инфографикой, верстка, презентация.

Soft Skills – критическое мышление, концентрация, ответственность, аналитическое мышление, креативное мышление, логическое мышление, внимание, командная работа, навык защиты проекта, навык отстаивать свою точку зрения.

Тема 3.10. Итоговое занятие по разделу.

Теория: Защита проекта. Итоговое тестирование по стартовому уровню. Рефлексия.

Базовый уровень

Модуль 4. Java. Arduino IDE

Тема 4.1. 2 часа. Техника безопасности. Среда разработки. "Hello, world!"

Тема 4.2. 6 часов. Знакомство с ATmega328. Регистры. Примитивные типы. Арифметика. `uintX_t, X = 8.16.32.64`

Тема 4.3. 4 часа. Операции отношения. Логические операции. Работа со светодиодами

Тема 4.4. 6 часов. Условные конструкции. Работа с кнопками

Тема 4.5. 6 часов. Итеративные конструкции. Массивы. Led-матрица

Тема 4.6. 8 часов. Метод конечных автоматов

Тема 4.7. 4 часа. Практика. Решение задач

Тема 4.8. 26 часов. Кейс "Сейф"

Тема 4.8.1. 4 часа. Постановка проблемы.

Тема 4.8.2. 4 часа. Идеи и формирование образа проекта

Тема 4.8.3. 8 часов. Макетирование проекта и деление на функциональные подзадачи

Тема 4.8.4. 10 часов. Прототипирование и тестирование. Подготовка презентации

Тема 4.9. 2 часа. Защита / Итоговое занятие по разделу

Тема 4.1. Техника безопасности. Среда разработки. Hello, world!

Теория: Знакомство с микроконтроллером Arduino Uno. Первичная настройка среды разработки Arduino IDE. Техника безопасности при работе с микроконтроллером. Запуск первой программы Hello, world! Знакомство с внутренней структурой программы в среде разработки.

Практика: Разбор примеров.

Тема 4.2. Знакомство с ATmega328. Регистры. Примитивные типы.

Арифметика. $uintX_t, X = 8.16.32.64$

Теория: Знакомство с примитивными типами данных языка Java. $uintX_t, X = 8.16.32.64$. Простейшая арифметика над этими типами данных. Знакомство с регистрами микроконтроллера Arduino Uno Atmega328. Изменение состояния портов вывода и чтение состояния портов ввода-вывода.

Практика: Разбор примеров.

Тема 4.3. Операции отношения. Логические операции. Работа со светодиодами.

Теория: Рассмотреть операторы и их классификацию, поразрядные операции, логические выражения.

Тип данных boolean. Логические значения true и false. Несовместимость типа boolean с int. Отметить, что приведение логических значений к целым и наоборот невозможно.

Логические операции и операции отношения. Операторы отношения: $>$, $<$, $>=$, $<=$, $!=$, $==$. Уяснение понятия значения операции отношения как ИСТИННО или ЛОЖНО. Логические операции: логическое И, логическое ИЛИ, логическое НЕ. Тернарная операция $?:$

Выражения и операции. По итогу изучения различных операций рассмотрение понятия выражения в языке программирования; знаки операций; знаки-разделители. Классификация операций по количеству operandов: унарные и бинарные. Классификация операций по типу: арифметические, логические, присваивания, отношения и др.

Практика: Разбор примеров.

Тема 4.4. Условные конструкции. Работа с кнопками

Теория: Изучить внутреннюю логику работы условных конструкций; приобрести навыки их использования в различных формах, предусмотренных синтаксисом языка. Закрепить навыки написания всех ранее изученных операторов путем написания и вычисления выражений.

Область действия блоков. Фигурные скобки для выделения блока. Вложенность блоков. На данный момент рассмотреть только ограничение на объявление переменных с одинаковым именем в одном и том же или вложенных блоках.

Конструкция if-else. Синтаксис оператора:

if (cond_expression) TRUE_statement

Или

if (cond_expression) TRUE_statement

else FALSE_statement

Разъяснить, что statement – это только один оператор или блок. Фундаментальное правило для сложных ветвлений, реализуемых с помощью вложенных конструкций if-else: else относится к ближайшему if, не имеющему else.

Конструкция switch-case. Синтаксис. Что может быть в качестве метки case. Мотивировка использования конструкции как упрощение сложных ветвлений. Логика выполнения, объяснение роли ключевых слов break и default в конструкции switch-case.

Практика: Разбор примеров

Тема 4.5. Итеративные конструкции. Массивы. Led-матрица

Теория: Изучить внутреннюю логику работы итеративных конструкций; приобрести навыки их использования в различных формах, предусмотренных синтаксисом языка.

Оператор for, for each, одномерные массивы. Определение массива как совокупности элементов одного и того же типа, расположенных вплотную друг за другом в памяти. Объявление массива двумя способами. Подчеркнуть необходимость создания массива с помощью new. Значения, которыми инициализируется массив по умолчанию при создании. Инициализация массива

без new – инициализация массива при объявлении. Доступ к отдельным элементам массива. Определение количества элементов в массиве через свойство length.

Цикл for. Синтаксис. Логика работы, роль каждой из составных частей. Частные формы записи оператора for: отсутствует инкрементальное выражение; отсутствует инкрементальное выражение и начальное выражение. Уяснение связи между for и while, эквивалентная запись for через while. Примеры некорректного использования операторов цикла, приводящего к зацикливанию. Вложенные циклы for.

Цикл for each. Синтаксис. Преимущества его применения при работе с массивами в сравнении с обычным for. Отметить, что переменная в цикле for each перебирает не индексы массива, а сами элементы массива.

Практика: Работа со светодиодной матрицей, разбор примеров

Тема 4.6. Метод конечных автоматов

Теория: Диаграмма состояний. Таблица переходов. Детерминированность. Автоматы и регулярные языки. Минимизация автоматов. Что может «делать» конечный автомат и последовательностная машина?

Практика: Разработка моделей с использованием конечных автоматов.

Тема 4.7. Практика. Решение задач

Тема 4.8. Кейс "Сейф"

Сборка электронного замка для сейфа.

Тема 4.8.1. Постановка проблемы

Теория: Дается проблема и ряд ограничений по ней. Отталкиваясь от этих ограничений, нужно прийти к вектору развития идеи.

Практика: Выполнение практического задания по теме кейса.

Компетенции:

Hard Skills – сбор данных, обработка информации.

Soft Skills – командная работа, коммуникативность, критическое мышление, аналитическое мышление, логическое мышление, исследовательские навыки, способность выслушать и принять мнение собеседников, умение высказать и отстоять свою точку зрения.

Тема 4.8.2. Идеи и формирование образа проекта

Теория: Опираясь на знание методов формирования идей и двигаясь по выбранному вектору развития, сгенерировать концепцию будущего проекта.

Практика: Выполнение практического задания по теме кейса.

Soft Skills – командная работа, коммуникативность, критическое мышление, аналитическое мышление, логическое мышление, способность выслушать и принять мнение собеседников, умение высказать и отстоять свою точку зрения.

Тема 4.8.3. Макетирование проекта и деление на функциональные подзадачи

Теория: Основываясь на сформулированной ранее концепции решения проблемы, создать макет, демонстрирующий набор функций по решению проблемы. Разделение всей концепции на подзадачи с последующим выделением функций каждой части на макете и нахождение ее реализации. Построение плана работы над проектом “от простого, к сложному”. Рефлексия.

Практика: Выполнение практического задания по теме кейса.

Hard Skills – работа с формообразованием, макетирование, объемно-пространственное мышление, интернет-сёрфинг, планирование рабочего процесса.

Soft Skills – командная работа, коммуникативность, аналитическое мышление, фокусировка, декомпозиция идей, логическое мышление, внимание, скрупулезность.

Тема 4.8.4. Прототипирование и тестирование. Подготовка презентации

Теория: Следуя плану работ, заменить макетные части на реальные. По ходу продвижения по плану фиксировать на контрольных точках, какие функции удалось реализовать и в какой мере. Создание плана презентации. Формулировка тезисов. Вёрстка презентации. Рефлексия.

Практика: Выполнение практического задания по теме кейса.

Hard Skills – работа с языком программирования Java, работа над прототипом, работа с планом презентации, работа с графическими редакторами, работа с фото / видео, работа с инфографикой, вёрстка, презентация.

Soft Skills – критическое мышление, концентрация, ответственность, аналитическое мышление, креативное мышление, логическое мышление,

внимание, командная работа, навык защиты проекта, навык отстаивать свою точку зрения.

Тема 4.9. Итоговое занятие по разделу

Теория: Защита презентации. Итоговое тестирование по базовому уровню. Рефлексия.

4. Планируемые результаты

Предметные результаты:

- знание и соблюдение требований техники безопасности и санитарно-гигиенических норм;
- знание основ языка программирования Java;
- умение использовать разные алгоритмы в приёмах программирования,
- умение пользоваться ПК и IDE-разработки для программирования устройства;
- умение читать готовую программу и находить ошибки в готовых программах.

Личностные результаты:

- формирование ответственного отношения к учению, готовности и способности обучающихся к саморазвитию и самообразованию, средствами информационных технологий;
- формирование универсальных способов мыслительной деятельности (абстрактно-логического мышления, памяти, внимания, творческого воображения, умения производить логические операции);
- формирование опыта участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- формирование ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Метапредметные результаты:

- ориентироваться в своей системе знаний: отличать новое знание от известного;
- умение производить анализ поставленной задачи, самостоятельно решать её;
- умение формулировать, аргументировать и отстаивать свое мнение;
- умение извлекать нужную информацию из открытых источников;
- умение составлять примерный алгоритм работы.

II. Комплекс организационно-педагогических условий реализации общеразвивающей программы

1. Календарный учебный график на 2021–2022 учебный год

Таблица 3

№ п/п	Основные характеристики образовательного процесса	
1.	Количество учебных недель	36
2.	Количество учебных дней	72
3.	Количество часов в неделю	4
4.	Количество часов на учебный год	144
5.	Недель в I полугодии	16
6.	Недель во II полугодии	20
7.	Начало занятий	13 сентября
8.	Выходные дни	30 декабря – 09 января
9.	Окончание учебного года	31 мая

2. Условия реализации общеразвивающей программы

Материально-техническое обеспечение

Требования к помещению:

- помещение для занятий, отвечающее требованиям СанПиН для учреждений дополнительного образования;
- качественное освещение.

Оборудование:

- столы и стулья по количеству обучающихся и 1 рабочим местом для педагога;

- ноутбуки, объединенные в локальную сеть;
- наушники;
- акустическая система;
- многофункциональное устройство (принтер, сканер и копир);
- планшеты (для отладки);
- смартфон и планшет на системе iOS (для отладки);
- комплекты с платформой Arduino UNO;
- моноблочное интерактивное устройство;
- напольная мобильная стойка для интерактивных досок или универсальное настенное крепление;
- доска магнито-маркерная настенная;
- флипчарт.

Расходные материалы:

- whiteboard маркеры;
- бумага писчая;
- шариковые ручки;
- permanent маркеры.

Информационное обеспечение:

- Программное обеспечение: IntelliJ IDEA IDE, Ardiono IDE, Notepad++;

- учебный материал по теме;
- демонстрационные программы;
- инструкции по настройке среды разработки;
- справочные материалы по терминам ПО.

Кадровое обеспечение:

Реализовывать программу могут педагоги дополнительного образования, обладающие достаточными знаниями в области педагогики и психологии, методологии, знающие особенности обучения основам мобильной разработки.

3. Формы аттестации и оценочные материалы

Система контроля знаний и умений обучающихся представляется в виде учёта результатов по итогам выполнения заданий отдельных кейсов и посредством наблюдения (Приложение 3), отслеживания динамики развития обучающегося (Приложение 2). В конце учебного года, обучающиеся проходят защиту индивидуальных / групповых проектов (Приложение 4, 5).

Индивидуальный / групповой проект оценивается формируемой комиссией. Состав комиссии (не менее 3-х человек): педагог (в обязательном порядке), администрация учебной организации, приветствуется привлечение ИТ-профессионалов, представителей высших и других учебных заведений.

Компонентами оценки индивидуального / группового проекта являются (по мере убывания значимости): качество ИП, отзыв руководителя проекта, уровень презентации и защиты проекта. Если проект выполнен группой обучающихся, то при оценивании учитывается не только уровень исполнения проекта в целом, но и личный вклад каждого из авторов. Решение принимается коллегиально. Для оценки проекта членам комиссии рекомендуется использовать «Бланк оценки ИП» (Приложение 5).

Итоговая аттестация учащихся осуществляется по 100-балльной шкале, которая переводится в один из уровней освоения образовательной программы согласно таблице:

Таблица 4

Набранные баллы обучающимся	Уровень освоения
0–50 баллов	низкий
51–75 баллов	средний
76–100 баллов	высокий

Формы проведения итогов по каждой теме и каждому разделу общеобразовательной программы соответствуют целям и задачам ДООП.

4. Методические материалы

Образовательный процесс осуществляется в очной форме.

В образовательном процессе используются следующие *методы обучения*:

- устные (беседы, объяснение);
- поисковые (изменение программы для приобретения устройством новых свойств);
- демонстрационные (демонстрация возможностей устройства);
- практические (написание программы, проведение мини-соревнований).

Программа обучения состоит из нескольких основных блоков:

- обучение основам языка Java;
- знакомство с Arduino.

Программой предусмотрены следующие виды деятельности обучающихся:

- работа с технической и справочной литературой;
- программирование;
- эксперимент, испытание.

Выбор методов обучения осуществляется исходя из анализа уровня готовности обучающихся к освоению содержания модуля, степени сложности материала, типа учебного занятия.

Формы обучения:

- *фронтальная* – предполагает работу педагога сразу со всеми обучающимися в едином темпе и с общими задачами. Для реализации обучения используется компьютер педагога с мультимедиа проектором, посредством которых учебный материал демонстрируется на общий экран. Активно используются Интернет-ресурсы;
- *групповая* – предполагает, что занятия проводятся с подгруппой. Для этого группа распределяется на подгруппы не более 6 человек, работа в которых регулируется педагогом;
- *индивидуальная* – подразумевает взаимодействие преподавателя с одним обучающимся. Как правило, данная форма используется в сочетании с

фронтальной. Часть занятия (объяснение новой темы) проводится фронтально, затем обучающийся выполняют индивидуальные задания или общие задания в индивидуальном темпе;

– **дистанционная** – взаимодействие педагога и обучающихся между собой на расстоянии, отражающее все присущие учебному процессу компоненты. Для реализации дистанционной формы обучения весь дидактический материал размещается в свободном доступе в сети Интернет, происходит свободное общение педагога и обучающихся в социальных сетях, по электронной почте, посредством видеоконференции или в общем чате. Кроме того, дистанционное обучение позволяет проводить консультации обучающегося при самостоятельной работе дома. Налаженная система сетевого взаимодействия подростка и педагога, позволяет не ограничивать процесс обучения нахождением в учебной аудитории, обеспечить возможность непрерывного обучения в том числе, для часто болеющих детей или всех детей в период сезонных карантинов (например, по гриппу) и температурных ограничениях посещения занятий.

Формы организации учебного занятия:

В образовательном процессе помимо традиционного учебного занятия используются многообразные формы, которые несут учебную нагрузку и могут использоваться как активные способы освоения детьми образовательной программы, в соответствии с возрастом обучающихся, составом группы, содержанием учебного модуля: беседа, лекция, мастер-класс, практическое занятие, защита проектов, конкурс, викторина, диспут, круглый стол, «мозговой штурм», воркшоп, квиз.

Некоторые формы проведения занятий могут объединять несколько учебных групп или весь состав объединения, например, экскурсия, викторина, конкурс и т. д.

Методы воспитания: мотивация, убеждение, поощрение, упражнение, стимулирование, создание ситуации успеха и др.

Педагогические технологии: индивидуализации обучения; группового обучения; коллективного взаимообучения; дифференциированного обучения; разноуровневого обучения; проблемного обучения; развивающего обучения;

дистанционного обучения; игровой деятельности; коммуникативная технология обучения; коллективной творческой деятельности; решения изобретательских задач; здоровьесберегающая технология.

Дидактические материалы:

Методические пособия, разрабатываемые преподавателем с учётом конкретных условий. Техническая библиотека объединения, содержащая справочный материал, учебную и техническую литературу. Индивидуальные задания.

Методическое обеспечение учебного процесса включает разработку преподавателем методических пособий, вариантов демонстрационных программ и справочного материала.

Список литературы

Нормативные документы:

1. Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 N 273-ФЗ.
2. Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р.
3. Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242. «О направлении Методических рекомендаций по проектированию дополнительных обще развивающих программ (включая разноуровневые)».
4. Распоряжение правительства РФ от 04.09. 2014 № 1726-р «Об утверждении Концепции развития дополнительного образования детей».
5. «Основы законодательства РФ об охране здоровья граждан», утвержденные Верховным советом РФ от 22.07.1993 № 5487 - (ред. от 25.11.2009);
6. Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребёнка в РФ»;
7. Федеральный закон от 21.11.2011 №323-ФЗ «Об основах охраны здоровья граждан в Российской Федерации».
8. Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Приказ №1008 отменен).
9. Санитарно-эпидемиологические правила и нормативы СанПиН 2.4.4.3172-14.

Список литературы, использованной при написании программы:

1. Блох Джошуа. Java. Эффективное программирование. Effective Java. Programming Language Guide. изд. «Лори». 2014. – 310 с. ISBN 978-5-85582-347-9.
2. Гослинг Джеймс, Билл Джой, Гай Л. Стил, Гилад Брача, Алекс Бакли. Язык программирования Java SE 8. Подробное описание. The Java Language Specification:

Java SE8 Edition. изд. «Вильямс». 2015 – 672 с. ISBN 978-5-8459-1875-8, 978-0-13-390069-9.

3. Зигард Медникс, Лайрд Дорнин, Блейк Мик, Масуми Накамура. Программирование под Android. Programming Android. изд. Питер. 2012 – 496 с. ISBN 978-5-459-01115-9, 978-1-449-38969-7.

4. Филиппов С.А. Робототехника для детей и их родителей. СПб, «Наука», 2013. – 319 с.

5. Эльконин Б.Д. Детская психология: учеб. пособие для студ. высш. учеб. заведений / Д. Б. Эльконин; ред.-сост. Б. Д. Эльконин. — 4-е изд., стер. — М.: Издательский центр «Академия», 2007. — 384 с

Электронные ресурсы:

6. Науменко О.М. Творчествоведение на современном этапе. Академия творческоведческих наук и учений. <http://atnu.narod.ru/tvorit.html> (06.2015).

7. Портал обучения Университета ИТМО. <https://de.ifmo.ru/>

8. Портал Хабрахабр. <https://habrahabr.ru>

9. Руководство пользователя платформы Arduino ENG. <http://arduino.cc/>

10. Руководство пользователя платформы Arduino RUS. <http://arduino.ru/>

Итоговая контрольная работа № 1

1. Записать дату своего дня рождения в формате:

DDMMYYYY, YYYYMMDD.

Пример: 05 августа 3245 г. – 05083245, 32450805

2. Перевести полученные числа в BIN(2), OCT(8) и HEX(16).

3. Записать 3 числа в формате DDMM, MMDD, YYYY.

Пример: 05 августа 3245 г. – 0508, 0805, 3245.

4. Вычислить следующие выражения:

a) DDMM | MMDD, DDMM | YYYY, MMDD | YYYY.

b) DDMM & MMDD, DDMM & YYYY, MMDD & YYYY.

c) DDMM ^ MMDD, DDMM ^ YYYY, MMDD ^ YYYY.

d) DDMM / MMDD, DDMM / YYYY, MMDD / YYYY.

e) DDMM % MMDD, DDMM % YYYY, MMDD % YYYY.

Приложение 2

Критерии оценивания обучающихся

№ группы: _____

Дата: _____

№ п/п	ФИО обучающегося	Сложность продукта (по шкале от 0 до 5 баллов)	Соответствие продукта поставленной задаче (по шкале от 0 до 5 баллов)	Презентация продукта. Степень владения специальными терминами (по шкале от 0 до 5 баллов)	Степень увлеченности процессом и стремления к оригинальности (по шкале от 0 до 5 баллов)	Кол-во вопросов и затруднений (шт. за одно занятие)
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						

Бланк наблюдения на обучающимися

Группа _____

Педагог _____

№ п/п	ФИО	ПОКАЗАТЕЛИ				РЕЗУЛЬТАТ
		Внимателен в течение занятия	Использует базовую систему понятий	Проявляет инициативу, интерес в течение занятия	Идет на деловое сотрудничество	
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						

За каждое согласие с утверждением 1 – балл.

План рассказа о проекте

1. Поприветствовать аудиторию. Представиться. Озвучить тему проекта.
2. Озвучить тему, актуальность, цели и задачи проекта.
3. Рассказать о выбранном наборе данных: источник, структура, размер.
4. Рассказать об использованных подходах, моделях и методах: причины выбора, структура, принцип работы.
5. Дать оценку качества работы модели по выбранным критериям.
6. Привести примеры работы модели.
7. В выводах озвучить, насколько достигнута поставленная цель и как усовершенствовать модель.
8. Поблагодарить за внимание.
9. Ответить на вопросы аудитории.

Бланк оценки индивидуальных/групповых проектов

Проект является одним из видов самостоятельной работы, предусмотренной в ходе обучения по программе. Педагог оказывает консультационную помощь в выполнении проекта.

В комплект обязательных материалов, которые представляются обучающимся, входит: исходный код программы в архиве, презентация проекта.

Бланк оценки индивидуальных/групповых проектов

ФИО члена комиссии _____

Дата _____

подпись расшифровка

Аннотация

Задача инновационного развития программного обеспечения требует соответствующей образовательной среды, в том числе создания оптимальных условий детского технического творчества. Одной из наиболее инновационных областей в сфере детского технического творчества является мобильная разработка.

Для дальнейшего развития мобильных приложений существует широкий выбор направлений разработки. Каждому ребёнку интересно, как устроена Java платформа, как работает Java приложение на любой платформе и на смартфоне в том числе.

Изучение языка программирования Java по данной программе обучения даёт возможность пользователю освоить базовые навыки использования языка программирования, понять его особенности использования и выполнения на различных платформах.

Разработка мобильных приложений на базе Android на сегодняшний день. очень востребована ввиду высокой популярности данной ОС. Поэтому обучение по данной программе – это самый первый, но важный шаг в изучении основ программирования на языке Java, для создания проектов и простейших программ в среде разработки на его основе.

Программа рассчитана на обучающихся 13–17 лет.