

Управление образования администрации Кемеровского городского округа
Муниципальное бюджетное образовательное
учреждение дополнительного образования
«Центр развития творчества детей и юношества Кировского района»

Принята на заседании
методического совета

от «04» 08 2023г.
Протокол № 12

Утверждаю:
Директор МБОУДО «ЦРТДиЮ
Кировского района»
/Т.В. Вяткина/
«04» 08 2023г.



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«Мобильная разработка»
техническая направленность
стартовый/базовый уровень
Возраст обучающихся: 12-16 лет
Срок реализации: 2 года

Разработчик:
Алексеева М.С.,
методист

Кемерово 2023 год

Оглавление

Пояснительная записка	3
Цель и задачи программы	7
Цель и задачи программы стартового уровня обучения	8
Учебный план программы стартового уровня обучения	8
Содержание программы стартового уровня обучения	11
Цель и задачи программы базового уровня обучения	19
Учебный план программы базового уровня обучения	19
Содержание программы базового уровня обучения	21
Планируемые результаты	24
Комплекс организационно-педагогических условий	26
Список литературы	30

КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ДОПОЛНИТЕЛЬНОЙ ОБЩЕОБРАЗОВАТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Мобильная разработка» относится к программам **технической направленности**.

Программа основывается на положениях основных законодательных, нормативных и рекомендательных актах Российской Федерации.

1. Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации».
2. Указ Президента РФ от 07.05.2012 № 599 «О мерах по реализации государственной политики в области образования и науки».
3. [Концепции развития дополнительного образования детей до 2030 года](#) (утверждена распоряжением Правительства РФ от 31.03.2022 № 678-р).
4. Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (утвержден Приказом Минпросвещения России от 27.07.2022 № 629)
5. Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (Приложение к письму Департамента государственной политики в сфере воспитания детей и молодежи Министерства образования и науки РФ от 18.11.2015 г. № 09-3242).
6. Стратегия развития информационного общества в Российской Федерации, утв. Указом Президента РФ от 09.05.2017 № 203 «О Стратегии развития информационного общества в Российской Федерации на 2017-2030 годы».
7. Постановление от 28.09.2020 г. № 28 об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».
8. Приказ Министерства просвещения Российской Федерации от 03.09.2019 г. № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;
9. Устав МБОУДО «ЦРТДиЮ Кировского района»; «Положением о дистанционном обучении учащихся МБОУДО «ЦРТДиЮ Кировского района».

Количество пользователей мобильными телефонами на операционных системах Android, iOS и Windows Phone растет с каждым днем. Человек с помощью смартфона получает доступ к неограниченной информации: может

вести бухгалтерию, планировать мероприятия, развлекаться, просматривая медиаконтент, устанавливать полезные программы и игры. За счет этого рынок мобильных приложений можно смело назвать перспективной сферой, в которой уже работает большое количество людей.

Новизна и актуальность.

Актуальность программы. Современный подросток проводит со своим смартфоном основную часть дня. Сегодня специалистами в области информационных технологий разрабатываются мобильные приложения, которые позволяют решать огромное количество задач. Некоторые служат для того, чтобы устанавливать соединение с сетью. Другие помогают оптимизировать маршрут. Третьи предназначены для тех, кто ищет самые выгодные магазины. Есть и такие, с помощью которых можно заказать еду на дом. В связи с этим разработка мобильных приложений является актуальным и целесообразным направлением в современном мире. Программа «Мобильная разработка» научит подростков создавать мобильные разработки, определять значимость и полезность разработки.

Занятия по данной дополнительной общеобразовательной программе смогут помочь ребятам выявить свои интересы и склонности, связанные с разработкой мобильных приложений, программированием. В ходе освоения программы, учащиеся получают универсальные знания алгоритмов создания программ и применении этих знаний для программирования конкретных приложений под ОС Android.

Новизна программы состоит в том, что она учитывает новые технологические уклады, которые требуют новый способ мышления и тесного взаимодействия при постоянном повышении уровня междисциплинарности проектов. Данная программа формирует профессиональные компетенции, которые позволят обучающимся в будущем успешно конкурировать на рынке рабочей силы в области мобильной разработки.

Отличительной особенностью дополнительной общеразвивающей программы «Мобильная разработка» в том, что она является разноуровневой, что предполагает наличие системы средств и приёмов, с помощью которых достигается интегрирующая дидактическая цель в совокупности всех разделов конкретной учебной дисциплины. Раздел – структурная единица образовательной программы, имеющая определённую логическую завершённость по отношению к результатам обучения. Каждый раздел состоит из тем, направленных на формирование определённых компетенций (hard и soft). Результатом каждого раздела является «продукт» (групповой, индивидуальный), демонстрирующий сформированность компетенций.

«Стартовый уровень» рассчитан на детей в возрасте с 12 лет, проявляющих интерес к аналитической и исследовательской деятельности, IT-технологиям, приобретению навыков программирования.

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

программы.

«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.

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.

Адресат программы. Программа предназначена для обучающихся, не имеющих предварительной подготовки по основным темам и разделам программы, адресована обучающимся 12-16 лет и имеющим минимальный необходимый уровень входных компетенций: уверенный пользователь ПК, проявляют интерес к программированию. Наполняемость групп стартового и базового уровня обучения не более 12 человек.

Обучающиеся, имеющие начальные знания по основным разделам программы, могут быть зачислены сразу на базовый уровень обучения, минуя стартовый.

Система набора осуществляется на основании результатов собеседования, просмотра работ, наличия базовых знаний в определённой области и т.д. Набор учащихся в объединение осуществляется на добровольной основе. Зачисление в группы производится на основании заполнения родителями (законными представителями) заявления о зачислении в учебное объединение.

Срок реализации программы и объем учебных часов. Программа реализуется на базе ИТ-Куба в учебном кабинете с необходимым оборудованием, техническим и ресурсным обеспечением в соответствии с перечнем, указанным в методических рекомендациях по созданию и функционированию центров цифрового образования «ИТ-Куб» в рамках федерального проекта «Цифровая образовательная среда» национального проекта «Образование». Предполагает два уровня обучения – стартовый и базовый, каждый соответствует одному году обучения и рассчитана на 252 часов, из них 108 часов – стартовый уровень обучения, 144 – базовый уровень обучения.

Формы организации образовательного процесса.

- **индивидуальная работа** (самостоятельное выполнение заданий);
- **групповая**, работа с наглядными пособиями по подгруппам;
- **массовая**, обсуждение результата, исправление ошибок.

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 лично ориентированные технологии, технологии сотрудничества.

Режим занятий. Стартовый уровень обучения рассчитан на 108 часов, 3 часа 1 раз в неделю. Базовый уровень обучения рассчитан на 144 часа, 2 раза в неделю по 2 часа. Продолжительность учебного занятия 45 минут. Между учебными занятиями предполагается перерыв 10 минут. Учебные занятия строятся с учетом психолого-педагогических и физических возрастных особенностей детей.

Цель и задачи программы

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

Задачи:

образовательные

- дать представление об основных понятиях информатики непосредственно в процессе создания информационного продукта;
- обучить методам программирования на языке Java, применяемым в современной вычислительной технике, и работе в интегрированных средах разработки;
- научить проектировать мобильное приложение и переносить его в мобильное устройство;
- научить создавать Android проекты, графический интерфейс;

развивающие

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

воспитательные

- содействовать профессиональному самоопределению обучающихся;
- воспитывать мотивацию учащихся к изобретательству, созданию собственных программных реализаций;
- воспитывать навыки самоорганизации, самостоятельной и командной работы.

УЧЕБНЫЙ ПЛАН ПРОГРАММЫ

Уровень обучения	Объем учебных часов	Всего учебных недель	Режим работы
стартовый	108	36	3 часа в неделю: 1 раз в неделю по 3 часа
базовый	144	36	4 часа в неделю: 2 раза в неделю по 2 часа

Цель и задачи программы стартового уровня обучения

Цель: развитие у обучающихся алгоритмического и объектного стиля мышления; формирование мотивации к получению образования в ИТ-сфере посредством организации практической деятельности.

Образовательные:

- Ознакомить с основами разработки серверной части мобильных приложений.
- Ознакомить с принципами и методами функционального программирования.
- Ознакомить с принципами и методами объектно-ориентированного программирования.
- Сформировать и развить навыки алгоритмического и логического мышления, грамотной разработки программ.

Развивающие:

- Развивать алгоритмическое и логическое мышление.
- Развивать умение постановки задачи, выделения основных объектов, математическое модели задачи.
- Развивать умение поиска необходимой учебной информации.
- Формировать мотивацию к изучению программирования.

Воспитательные:

- Воспитывать умение работать индивидуально и в группе для решения поставленной задачи.
- Воспитывать трудолюбие, упорство, желание добиваться поставленной цели.
- Воспитывать информационную культуру.

Учебный план программы стартового уровня обучения

№	Название раздела, темы	Общее количество часов			Формы контроля
		Всего	Теория	Практика	
1	Раздел 1. Логика	9	2	7	
1.1	«Событие». Понятие, группа	3	1	2	Беседа. Опрос
1.2	Бинарная логика. Бинарные операторы	3	1	2	Беседа. Опрос Практическая работа
1.3	Решение задач	3	-	3	Самостоятельная работа
2	Раздел 2. Математика. Система исчислений	15	3	12	
2.1	Арифметика в информатике. DEC. СИ десятичная	3	1	2	Беседа. Опрос

2.2	VIN. СИ двоичная. Bit. Byte	6	1	5	Беседа. Опрос Практическая работа
2.3	HEX. СИ шестнадцатеричная	3	1	2	Беседа. Опрос Практическая работа
2.4	Решение задач	3	-	3	Самостоятельная работа
3	Раздел 3. Java. IntelliJ IDEA IDE	84	17	67	
3.1	Техника безопасности. Среда разработки. «Hello, world!»	3	1	2	Беседа. Опрос
3.2	System.in/out. Тип String	3	1	2	Беседа. Опрос Практическая работа
4	Раздел 4 «Примитивные типы»	6	1	5	Беседа. Опрос Практическая работа
4.1	Представление целых чисел: прямой код, код со сдвигом, дополнительный код. byte, short, int, long	6	1	5	Беседа. Опрос Практическая работа
4.2	Представление вещественных чисел. float, double	3	1	2	Беседа. Опрос Практическая работа
4.3	Представление символов, таблицы кодировок. Char	3	1	2	Беседа. Опрос Практическая работа
4.4	Арифметика числовая и символьная	3	1	2	Беседа. Опрос Практическая работа
4.5	Операции отношения. Логические операции	6	1	5	Беседа. Опрос Практическая работа
4.6	Условные конструкции. Тип Boolean	3	1	2	Беседа. Опрос Практическая работа
4.7	Итеративные конструкции. Массивы	6	1	5	Беседа. Опрос Практическая работа
4.8	Функции. Синтаксис. Виды. Рекурсия	3	1	2	Беседа. Опрос Практическая

					работа
4.9	Решение задач	3	1	2	Самостоятельная работа
5	Раздел 5 «Конвертер BIN<->DEC<->HEX»	9	1	8	Беседа. Опрос Практическая работа
5.1	Постановка проблемы	3	1	2	Беседа. Опрос Практическая работа
5.2	Идеи и формирование образа проекта	6	1	5	Беседа. Опрос Практическая работа
5.3	Макетирование проекта и деление на функциональные подзадачи	9	1	8	Беседа. Опрос Практическая работа
5.4	Прототипирование и тестирование. Подготовка презентации	6	1	5	Беседа. Опрос Практическая работа
5.5	Защита	3	-	3	Защита проекта. /Итоговое тестирование по стартовому уровню
	Общее количество часов	108	22	86	

Содержание программы стартового уровня обучения

Раздел 1. Логика

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 В В том случае, если А При А будет В Из А следует В В случае А произойдет В В, так как А В, потому что А А – достаточное условие для В В – необходимое условие для А Множество. Дать определение, показать группировку элементов во множество.

Тема 1.2. Бинарная логика. Бинарные операторы

Теория: Знакомство с операторами и их применением, логическими операциями. Операции с множествами (пересечение, дополнение, разность).

Пересечение множеств: $A \cap B := \{x/x \in A \wedge x \in B\} \vee \exists$

Объединение множеств: $A \cup B := A \cap B = \{x/x \notin A \wedge x \in B\}$

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. 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, D. 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 D не нарушил, то А нарушил, а С не нарушал. 4. Если D 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?

Раздел 2. Математика. Система исчислений

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 смешенная) 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 17 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. Решение задач

Практика: Задание 2.4.1. Дано $A=A716$, $B=2518$. 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 100q фруктовых деревьев: из них 33q яблони, 22q груши, 16q слив и 17q вишен. Найдите основание q системы счисления, в которой посчитаны деревья. Задание 2.4.5. Найдите основание x системы счисления, если известно, что $2002x = 13010$.

Раздел 3. Java. IntelliJ IDEA IDE

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. Тип 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.

Раздел 4. Примитивные типы.

Тема 4.4 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 4.5 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 4.6 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 4.7 Арифметика числовая и символьная

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

Практика: Разбор примеров по изучаемой теме.

Тема 4.8 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 операндов: 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 4.9 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 4.10 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 заикливанию. 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 4.11 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 3.12 Решение задач

Практика: интеллектуальная игра.

Раздел 5. Кейс "Конвертер BIN<->DEC<->HEX"

Тема 5.1 Постановка проблемы

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

Практика: 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 отстаивать свою точку зрения.

Тема 5.2 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 отстаивать свою точку зрения.

Тема 5.3 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 вычленением 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, объемнопространственное мышление, интернет-сёрфинг, планирование рабочего процесса. Soft Skills – командная работа, коммуникативность, аналитическое мышление, фокусировка, декомпозиция идей, логическое мышление, внимание, скрупулезность.

Тема 5.4 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 5.5 Защита.

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

Цель и задачи программы базового уровня обучения

Цель: сформировать навыки программирования на языке Java и разработки мобильных приложений под операционную систему Android для реализации собственных проектов, вызванных конкретной потребностью каждого обучающегося.

Задачи:

Обучающие задачи:

- изучение основных понятий об алгоритмах;
- изучение алгоритма разработки проекта мобильного приложения;
- изучение современных и часто используемых платформ для разработки мобильных приложений;
- формирование навыков объектно-ориентированного проектирования при разработке приложения;
- формирование навыков программирования на языке Java.

Развивающие:

- способствовать развитию творческих способностей учащихся, познавательных интересов, развитию индивидуальности и самореализации;
- расширять технологические навыки при подготовке различных информационных материалов;
- развивать познавательные способности ребенка, память, внимание, пространственное мышление, аккуратность и изобретательность;
- развивать навыки инженерного мышления, умения работать по инструкциям и находить свои собственные пути решения поставленных задач;
- формировать навыки рефлексивной деятельности.

Воспитательные:

- воспитать стремление учащихся к созданию собственных программных продуктов;
- привить стремление к получению качественного законченного результата в проектной деятельности;
- привить информационную культуру: ответственное отношение к информации с учетом правовых и этических норм её распространения;
- формировать правильное восприятие системы ценностей, принципов, правил информационного общества.

Учебный план программы базового уровня обучения

№	Название раздела, темы	Общее количество часов			Формы контроля
		Всего	Теория	Практика	
1	Раздел 1. Java. Arduino IDE	74	12	62	
1.1	Техника безопасности. Среда разработки. «Hello, world!»	4	2	2	Беседа. Опрос
1.2	Знакомство с ATmega328. Регистры.	12	2	10	Беседа. Опрос Практическая

	Примитивные типы. Арифметика. uintX_t, X = 8.16.32.64				работа
1.3	Операции отношения. Логические операции. Работа со светодиодами	14	2	12	Беседа. Опрос Практическая работа
1.4	Условные конструкции. Работа с кнопками	14	2	12	Беседа. Опрос Практическая работа
1.5	Итеративные конструкции. Массивы. Led-матрица	14	2	12	Беседа. Опрос Практическая работа
1.6	Метод конечных автоматов	12	2	10	Беседа. Опрос Практическая работа
1.7	Решение задач	4	-	4	Самостоятельная работа
2	Раздел 2. «Сейф»	70	8	62	
2.1	Постановка проблемы	4	2	4	Беседа. Опрос
2.2	Идеи и формирование образа проекта	12	2	12	Беседа. Опрос Практическая работа
2.3	Макетирование проекта и деление на функциональные подзадачи	22	2	22	Беседа. Опрос Практическая работа
2.4	Прототипирование и тестирование. Подготовка презентации	24	2	24	Беседа. Опрос Практическая работа
2.5	Защита/Итоговое занятие по разделу	8	-	8	Защита проекта. /Итоговое тестирование по стартовому уровню
	Общее количество часов	144	20	122	

Содержание программы базового уровня обучения

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 операндов: 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: 30 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 заикливанию. 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. Решение задач

Практика: интеллектуальная игра.

Раздел 2 «Сейф»

Тема 4.1 Сборка электронного замка для сейфа. 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 отстаивать свою точку зрения.

Тема 4.2 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 отстаивать свою точку зрения.

Тема 4.3 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 вычленением 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, объемнопространственное мышление, интернет-сёрфинг, планирование рабочего процесса. Soft Skills – командная работа, коммуникативность, аналитическое мышление, фокусировка, декомпозиция идей, логическое мышление, внимание, скрупулезность.

Тема 4.4 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 4.5 Итоговое занятие по разделу

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

Планируемые результаты

Предметные результаты

В результате освоения программы учащиеся

будут знать:

- технику безопасности при нахождении в IT-Кубе, работе со специальным оборудованием при выполнении практико-ориентированных заданий;
- правила безопасной работы на компьютере;
- назначение и функции используемых информационных технологий;
- особенности работы с интегрированной средой разработки;
- базовые и сложные конструкции, способы организации процедур и функций в языке программирования Java;
- принципы разработки мобильных приложений;
- особенности различных мобильных платформ;
- этапы разработки проектов; правила презентации и продвижения проектного продукта;

будут уметь:

- организовывать рабочее место;
- соблюдать технику безопасности, технологически правильно обращаться с оборудованием IT-Куба и инструментами при выполнении практико-ориентированных работ, следовать требованиям гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;
- подключать библиотеки;
- создавать графический интерфейс и загружать нужные изображения в программу;
- создавать обработчики для описания различных событий;
- проектировать пользовательский интерфейс;
- переносить приложение в мобильное устройство;
- работать с файлами;
- создавать базу данных и строить к ней простейший запрос;
- эффективно использовать интегрированную среду разработки;
- проектировать мобильные приложения, создавать программы и выполнять их отладку на мобильных устройствах;
- писать код программы на языке Java;
- использовать ПК и IDE-разработки для программирования устройства;
- 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);

- развитие 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.

**КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ.
КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК**

№	Год обучения	Дата начала занятий	Дата окончания занятий	Количество учебных недель	Количество учебных дней	Количество учебных часов	Режим занятий	Сроки проведения промежуточной/ итоговой аттестации
1	Первый год (стартовый уровень)	15.09.2023	31.05.2024	36	1	3	1 занятие по 3 часа в неделю	Промежуточная аттестация – декабрь 2023 Итоговая аттестация май 2024
2	Первый год (базовый уровень)	15.09.2023	31.05.2024	36	2	4	2 занятия по 2 часа в неделю с перерывом на перемену 10 минут	Промежуточная аттестация – декабрь 2023 Итоговая аттестация май 2024

УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Материально-техническое обеспечение

Условия реализации программы: учебный кабинет, оснащенный оборудованием (стандарт).

Перечень необходимого оборудования и расходных материалов (количество единиц оборудования и материалов указано из расчета на 12 человек):

– компьютеры и ноутбуки, на которых установлено соответствующее программное обеспечение: на каждого учащегося и преподавателя - 13 шт. или 1 шт. на малую группу (должны быть подключены к единой Wi-Fi сети с доступом в интернет);

– презентационное оборудование – 2 шт.;

– маркерная доска – 1 шт.;

– Планшет- 12 шт

– МФУ (принтер, сканер, копир) тип 2 – 1 шт (Рабочее место педагога).

Кадровое обеспечение

Программа реализуется педагогом дополнительного образования, имеющим опыт работы в учреждении дополнительного образования технической направленности, образование – не ниже средне-профессионального, профильное или педагогическое.

Формы контроля и подведения итогов реализации программы

На занятиях используются: входной и текущий контроль, промежуточная и итоговая аттестация.

Входной контроль осуществляется через наблюдение за деятельностью учащихся, предполагает собеседование с учащимися, в ходе которого определяется наличие у них минимального необходимого уровня входных компетенций: уверенный пользователь ПК, знание языка Java на стартовом уровне.

Текущий контроль осуществляется посредством наблюдения за деятельностью учащихся на каждом занятии и фиксации их умений во время работы над практическими заданиями/работами по разделам, тестами. Отмечается активность участия учащихся в мероприятиях, степень самостоятельности при работе над практическими заданиями, самостоятельный поиск и разработка интересных тем для доклада (или мини-проекта) по направлению «Мобильные разработки».

Промежуточная и итоговая аттестация предполагает разработку и реализацию проектов, представление и защиту индивидуальных и групповых проектов, публичное выступление с демонстрацией результатов работы, творческое портфолио, экспертную оценку наставника, участие в профильных конкурсах и мероприятиях.

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Раздел программы	Формы занятий	Приёмы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение занятий	Формы подведения итогов
Вводное занятие	Лекция, презентация, игра, инструктаж	Словесно-наглядный, проблемное изложение, поиск ответов на поставленные вопросы	Презентация, инструкции, подборка профильных мероприятий	Оборудование ИТ-Куба	Собеседование, наблюдение
Основы программирования	Лекция, демонстрация, самостоятельная работа, групповая, практическая работа, практикум	Словесно-наглядный, поисковый, практический, проблемный	Презентация, медиатека, тематические материалы, тестовые задания	Оборудование ИТ-Куба	Практические работы, тест
Основы программирования на Android	Лекция, групповая, индивидуальная, практическая работа	Словесно-наглядный, поисковый, практический, проблемный	Презентация, медиатека	Оборудование ИТ-Куба	Практические работы, самостоятельная работа, экспертная оценка наставника
Создание практического приложения	Лекция, групповая, индивидуальная, практическая работа, работа в парах, проекты	Словесно-наглядный, поисковый, практический, проблемный	Презентация, медиатека	Оборудование ИТ-Куба	Практические работы, проекты, экспертная оценка наставника

Функционирование приложений в системе Android	Лекция, групповая, индивидуальная, практическая Работа, проекты	Словесно-наглядный, поисковый, практический, проблемный	Презентация, медиатека	Оборудование ИТ-Куба	Практические работы, проекты, экспертная оценка наставника
Основные структуры данных	Проект, практическая работа	Словесно-наглядный, поисковый, практический, проблемный, презентация проектов	Презентация, банк проектов	Оборудование ИТ-Куба	Проекты, практические работы, экспертная оценка наставника
Аттестация. Защита проекта	Выступление, демонстрация, опрос	Поисковый, практический, проблемный	Презентация, портфолио, подборка Профильных мероприятий	Оборудование ИТ-Куба	Защита проекта, портфолио

Список литературы

1. Авдеева А. В., Богданова М. В. // Информационные технологии в образовательном процессе вуза и школы: материалы XIV Всероссийской научно-практической конференции. — 2020. — С. 5—10.
2. Аллан А. Программирование для мобильных устройств на iOS: Профессиональная разработка приложений для iPhone, iPad, and iPod Touch / А. Аллан. - СПб.: Питер, 2013. - 416 с.
3. Байбородова Л. В. Трансформация дидактических принципов в условиях цифровизации образования / Л. В. Байбородова, Н. В. Тамарская // Педагогика. — 2020. — № 7. — С. 22—30.
4. Веракса Н. Е. Проблема средств в цифровом обучении / Н. Е. Веракса, А. Н. Веракса // Педагогика. — 2020. — № 4. — С. 19—26.
5. Глейзер, Дж. Многопользовательские игры. Разработка сетевых приложений / Дж. Глейзер, С. Мадхав. - СПб.: Питер, 2019. - 152 с.
6. Дэрси, Л. Разработка приложений для Android-устройств. Т. 1: Базовые принципы / Л. Дэрси, Ш. Кондер. - М.: Лори, 2014. - 402 с.
7. Заяц, А.М. Проектирование и разработка WEB-приложений. Введение в frontend и backend разработку на JavaScript и node.js: Учебное пособие / А.М. Заяц, Н.П. Васильев. - СПб.: Лань, 2019. - 120 с.
8. Милехина О. В. Информационные системы: теоретические предпосылки к построению: учеб пособие / О. В. Милехина, Е. Я. Захарова, В. А. Титова. – 2-е изд-е. – Новосибирск: Изд-во НГТУ, 2014. - 283 с.
9. Соколова В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений: учебное пособие для вузов / В. В. Соколова. – Москва: Издательство Юрайт, 2020. – 175 с. – (Высшее образование).
10. Соколова В. В. Разработка мобильных приложений: Учебное пособие / Соколова В.В. – Томск: Изд-во Томского политех. университета, 2014. – 176 с.
11. Федотенко М.А. Разработка мобильных приложений. Первые шаги / М.А. Федотенко; под ред. В.В. Тарапаты. – Москва: Лаборатория знаний, 2019. – (Школа юного программиста).
12. Цыганенко В. Н. CALS/CASE-технологии проектирования информационных систем: конспект лекций. Изд-во ОмГТУ, 2007. - 86 с.