

СЕТЬ ЦЕНТРОВ ЦИФРОВОГО
ОБРАЗОВАНИЯ ДЕТЕЙ «ИТ-КУБ»

ОБРАЗОВАНИЕ
НАЦИОНАЛЬНЫЕ
ПРОЕКТЫ
РОССИИ

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОРЕНБУРГСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ «АКБУЛАКСКИЙ
ПОЛИТЕХНИЧЕСКИЙ ТЕХНИКУМ»
Центр цифрового образования «ИТ-куб»



УТВЕРЖДАЮ

Директор ГАПОУ «АПТ»

Е.В.Симакова

« 27 » 08 2024 г.

Дополнительная общеобразовательная (общеразвивающая) программа
технической направленности

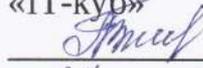
«Мобильная разработка»
(Углубленный уровень)
возраст обучающихся 12-17 лет

Срок реализации 1 год

Объем программы : 72 часа

СОГЛАСОВАНО:

Руководитель центра цифрового образования
«ИТ-куб»

 Г.В.Жукова

« 27 » 08 2024 г.

Автор-составитель:

Г.В.Жукова,

педагог

дополнительного

образования

Акбулак 2024г.

СОДЕРЖАНИЕ

1. Основные понятия и термины	2
2. Пояснительная записка.....	5
3. Цель и задачи программы.....	6
4. Нормативная база.....	6
5. Описание материально-технической базы центра цифрового образования детей «ИТ-Куб».....	8
6. Учебный план	10
7. Планируемые результаты	15
8. Методические рекомендации по проведению уроков.....	16
8.1. Форма аттестации	17
8.2. Методическое обеспечение.....	19
9. Перечень доступных источников информации	20

1. Основные понятия и термины

Активности – это специальные классы, представляющие и контролирующую работу одного экрана приложения Андроид.

Блок АИ – это визуальный программный блок, похожий на пазл. Обычно работает в комбинации с другими блоками.

Класс- это базовая структура единицы языка Java. Представлена в виде файла с расширением .java.

Компонента АИ – это визуальная компонента, размещаемая на экране приложения. Может быть видимой, как кнопка или изображение. Иногда бывает невидимой, например компонента, представляющая собой функционал камеры или сенсора.

ЛКМ,ПКМ – это левая и правая кнопка мыши.

ОС Андроид – это операционная система Андроид.

Платформа Андроид- это библиотеки и компоненты для разработки Андроид- приложений.

Приложение для сканирования QR- кода- это специальное мобильное приложение для распознавания QR- кода.

Расположения – это особые компоненты АИ, представляющие собой контейнеры для других компонентов.

Сенсоры- это датчики мобильного устройства.

Список (массив) – это упорядоченная изменяемая последовательность элементов различного типа.

Текстура – это изображение, близкое по визуальным свойствам к реальным объектам.

Файл манифеста- это файл с базовыми настройками Андроид- приложения.

Эмулятор – это система программных средств, которая копирует функции мобильного устройства на базе платформы Андроид с целью максимально близкой имитации эмулятором поведения мобильного устройства. Это позволяет запускать Андроид – приложения при отсутствии мобильного устройства.

Язык программирования- это набор формальных правил, по которым пишут программы.

АИ – app Inventor.

АРК – это формат архивных исполняемых файлов- приложений для Android и ряда других операционных систем, основанных на Android. Каждое приложение Android скомпилировано и упаковано в один файл, который включает в себя код приложения, ресурсы, активности, файл манифест и пр.

Google Play – это магазин приложений Google, куда Андроид- разработчики могут выставлять свои приложения. Соответственно пользователи мобильных устройств на базе Андроид могут оттуда скачивать приложения и устанавливать их на свои устройства.

IDE- это интегрированная среда обработки.

IntelliJ IDEA – это интегрированная среда разработки компании JetBrains.

IoT – это Internet of Things (интернет Вещей).

Java – это объективно- ориентированный язык высокого уровня со строгой типизацией.

JVM- это Java virtual Machine, виртуальная машина Java, специальная среда для выполнения байт – кода.

QR- код – это двухмерный штрихкод.

2. Пояснительная записка

Программа дополнительного образования по тематическому направлению «Мобильная разработка» имеет техническую направленность.

Целью программы является формирование у учащихся технической грамотности средствами приобщения обучающихся к разработке программ под современную платформу Android.

Новизна программы. Курс носит междисциплинарный характер и позволяет решить задачи развития у учащихся научно-исследовательских, проектных, технико-технологических и гуманитарных компетенций. В ходе освоения программы учащиеся разовьют навыки исследовательской, проектной деятельности. Также стоит отметить, что большое количество времени уделяется творческим заданиям, выполнение которых благоприятно скажется на развитии творческого потенциала учащихся.

Актуальность программы. Программа строится на концепции подготовки учащихся к профессии программиста – профессии будущего. Выросла потребность общества в технически грамотных специалистах и полностью отвечающих социальному заказу по подготовке квалифицированных кадров в области программирования. Знания, умения и практические навыки решения актуальных задач, полученные на занятиях, готовят учащихся 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.

Педагогическая целесообразность. Данная программа педагогически целесообразна, т.к. ее реализация органично вписывается в единое образовательное пространство данной образовательной организации. Программа соответствует новым стандартам обучения, которые обладают отличительной особенностью, способствующей личностному росту учащихся, его социализации и адаптации в обществе.

Отличительные особенности программы. Заключаются в том, что она является практико-ориентированной. Освоенный детьми теоретический материал закрепляется в виде практических заданий, решения поставленных задач, выполнения проектов. На практических занятиях учащиеся решают актуальные прикладные задачи. Таким образом, обеспечено простое запоминание сложнейших терминов и понятий, которые в изобилии встречаются в обучении.

Адресат программы. Возраст детей, участвующих в реализации данной общеобразовательной программы: от 12 до 17 лет. Принимаются все желающие. Наполняемость в группах до 12 человек.

Режим занятий. Занятия проводятся 1 раз в неделю по 2 учебных часа. Продолжительность занятия - 45 минут. После 45 минут занятий организовывается перерыв длительностью 10 минут для проветривания помещения и отдыха учащихся.

Сроки реализации программы 1 год, занятия проводятся 1 раз в неделю по 2 учебных часа, недельная нагрузка.

Форма обучения: очная.

Программа предоставляет учащимся возможность освоения учебного содержания занятий с учетом их уровней общего развития, способностей, мотивации. В рамках программы предполагается реализация параллельных процессов освоения содержания программы на разных уровнях доступности и степени сложности, с опорой на диагностику стартовых возможностей каждого из участников.

3. Цели и задачи программы

Целью программы является формирование технической грамотности средствами приобщения обучающихся к разработке программ под современную платформу Android.

Для достижения поставленной цели планируется выполнение следующих задач

Образовательные:

- расширение знаний о современных популярных платформах;
- обучение языку программирования Kotlin, языку разметки XML;
- обучению объектно-ориентированному подходу в проектировании и разработке программного обеспечения;
- знакомство с архитектурой приложений под Android - обучение программированию технических устройств.

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.

Дополнительная общеобразовательная (общеразвивающая) программа разработана на основе педагогического опыта автора-составителя программы, и нормативно-правовой документации.

1. Нормативная база

1. Конституция Российской Федерации (принята всенародным голосованием 12.12.1993 с изменениями, одобренными в ходе общероссийского голосования 01.07.2020).
2. Федеральный закон от 29.12.2012 № 273-ФЗ (ред. от 31.07.2020) «Об образовании в Российской Федерации» (с изм. и доп., вступ. в силу с 01.09.2020).
3. Паспорт национального проекта «Образование» (утверждён президиумом Совета при Президенте РФ по стратегическому развитию и национальным проектам, протокол от 24.12.2018 № 16).
4. Государственная программа Российской Федерации «Развитие образования» (утверждена постановлением Правительства РФ от 26.12.2017 № 1642 (ред. от 22.02.2021) «Об утверждении государственной программы Российской Федерации «Развитие образования»).
5. Стратегия развития воспитания в Российской Федерации на период до 2025 года (утверждена распоряжением Правительства РФ от 29.05.2015 № 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»).
6. Профессиональный стандарт «Педагог (педагогическая деятельность в дошкольном, начальном общем, основном общем, среднем общем образовании), (воспитатель, учитель)» (ред. от 16.06.2019) (приказ Министерства труда и социальной защиты РФ от 18 октября 2013 г. № 544н, с изменениями, внесёнными приказом Министерства труда и соцзащиты РФ от 25 декабря 2014 г. № 1115н и от 5 августа 2016 г. № 422н).
7. Профессиональный стандарт «Педагог дополнительного образования детей и взрослых» (приказ Министерства труда и социальной защиты РФ от 5 мая 2018 г. № 298н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»).
8. Федеральный государственный образовательный стандарт основного общего образования (утверждён приказом Министерства образования и науки Российской Федерации от 17 декабря 2010 г. № 1897) (ред. 21.12.2020)
9. Федеральный государственный образовательный стандарт среднего общего образования (утверждён приказом Министерства образования и науки Российской Федерации от 17 мая 2012 г. № 413) (ред. 11.12.2020).
10. Методические рекомендации по созданию и функционированию детских технопарков «Кванториум» на базе общеобразовательных организаций

(утверждены распоряжением Министерства просвещения Российской Федерации от 12 января 2021 г. № Р-4).

11. Методические рекомендации по созданию и функционированию центров цифрового образования «ИТ-куб» (утверждены распоряжением Министерства просвещения Российской Федерации от 12 января 2021 г. № Р-5).
12. Методические рекомендации по созданию и функционированию в общеобразовательных организациях, расположенных в сельской местности и малых городах, центров образования естественно-научной и технологической направленностей («Точка роста»)(утверждены распоряжением Министерства просвещения Российской Федерации от 12 января 2021 г. № Р6).

1. Описание материально-технической базы центра цифрового образования детей «ИТ-Куб»

Для организации учебного процесса в рамках реализации дополнительной общеобразовательной программ по тематическому направлению «Мобильная разработка» согласно распоряжению «Об утверждении методических рекомендаций по созданию и функционированию центров цифрового образования «ИТ-Куб» от 12.02.2021 № ТВ-1984/04 рекомендуется следующее оборудование лаборатории:

Рабочее место преподавателя и ученика:

- Планшет (13 штук);
- Наушники (12 штук);
- Ноутбук тип 2 (12 штук);
- МФУ;
- Ноутбук тип 1 (рабочее место педагога)
- Напольная мобильная стойка для интерактивных досок или универсальное настенное крепление Моноблочное интерактивное устройство.

1. Учебный план Тематическое планирование

№ п. п	Тема	Содержание	Целевая установка урока	Кол-во часов	Основные виды деятельности обучающихся на уроке / внеурочном занятии	Использованное оборудование
1.	Модуль 1. Основы программирования.	Вводное занятие. Инструктаж по ТБ. Введение в программу. Среда разработки. Здравствуй мир! (Kotlin). Примитивные типы данных. Арифметика. Операции отношения и логические операции. Условные конструкции. Блоки. Итеративные конструкции. Методы (функции). Видимость переменных. Многомерные и неровные массивы	Ознакомить со средой разработки. Здравствуй мир!(Kotlin), примитивными типами данных, арифметикой, операциями отношений и логическими операциями, условными и интерактивными конструкциями, блоки, методами (функции), видимостью переменных, многомерными и неровными массивами	12	Наблюдение за работой учителя, самостоятельная работа в среде App Inventor, ответы на вопросы, участие в дискуссии. Выполнение лабораторных работ.	Компьютер, проектор, интерактивная доска.
2.	Модуль 2. Объектно-ориентированное программирование	Классы и объекты. Классы: конструкторы, статические методы. Строки. Начальные приёмы тестирования и отладки. Знакомство с Android разработкой. Интерфейс пользователя. Наследование, инкапсуляция и полиморфизм в Kotlin. Context и Intent. Параметризованные типы	Ознакомить с классами и объектами, классы: конструкторы, статические методы. Строками Начальными приёмами тестирования и отладки. Знакомство с Android разработкой. Интерфейс пользователя. Наследование, инкапсуляция и	12	Наблюдение за работой учителя, самостоятельная работа в среде App Inventor, ответы на вопросы, участие в дискуссии. Выполнение лабораторных работ.	Компьютер, проектор, интерактивная доска.

			<p>полиморфизм в Kotlin. Context и Intent. Параметризованные типы</p>			
3.	<p>Модуль 3. Основы программирования Android-приложений</p>	<p>Практикум ООП проектирования. Ввод-вывод в Kotlin. Обработка исключений. Внутренние классы в обработке событий. Параллелизм и синхронизация. Сервисы в Android. Фрагменты (Fragments). Сенсоры. Двумерная графика в Android- приложениях. Разработка игровых приложений. Разработка 3D игр с использованием фреймворка libGDX</p>	<p>Ознакомить с практикумом ООП проектирования, вводом-выводом в Kotlin, обработкой исключений, внутренними классами в обработке событий, параллелизмом и синхронизацией, сервисами в Android, фрагментами (Fragments), сенсорами, двумерной графикой в Android- приложениях. Разработка игровых приложений. Разработка 3D игр с использованием фреймворка libGDX</p>	14	<p>Наблюдение за работой учителя, самостоятельная работа в среде App Inventor, ответы на вопросы, участие в дискуссии. Выполнение лабораторных работ.</p>	<p>Компьютер, проектор, интерактивная доска.</p>
4.	<p>Модуль 4. Алгоритмы и структуры данных</p>	<p>Массивы. Алгоритм двоичного поиска. Списки. Адаптеры в Android. Деревья. Рекурсия. Алгоритмы сортировки. Хэш-таблица и функция хэширования. Ассоциативные массивы. Реляционная модель данных. СУБД. Локальные СУБД. Введение в SQL.</p>	<p>Знакомство с массивами, алгоритмом двоичного поиска, списками, адаптерами в Android, деревьями, рекурсией, алгоритмами сортировки, хэш-таблицами и функциями хэширования, ассоциативными массивами, Реляционной моделью данных, СУБД, Локальными СУБД, введением в SQL.</p>	18	<p>Наблюдение за работой учителя, самостоятельная работа в среде App Inventor, ответы на вопросы, участие в дискуссии. Выполнение лабораторных работ.</p>	<p>Компьютер, проектор, интерактивная доска.</p>
5.	<p>Модуль 5. Алгоритмы и</p>	<p>IP-сети. Web-сервер. HTTP-запросы и ответы.</p>	<p>Ознакомить с IP-сетями, Web-сервером, HTTP-</p>	12	<p>Наблюдение за работой учителя,</p>	<p>Компьютер, проектор,</p>

	структуры данных	Клиент-серверная архитектура мобильных приложений. Облачные платформы. REST взаимодействие. Серверные СУБД. Material Design.	запросами и ответами, клиент-серверной архитектурой мобильных приложений, облачными платформами, REST взаимодействием, серверными СУБД. Material Design.		самостоятельная работа в среде App Inventor, ответы на вопросы, участие в дискуссии. Выполнение лабораторных работ.	интерактивная доска.
6.	Тестирование	Создание приложений	Проверка полученных навыков по теме «Работа с компонентами интерфейса и программными блоками в среде AI»	2	Наблюдение за работой учителя, самостоятельная работа в среде App Inventor, ответы на вопросы, участие в дискуссии. Выполнение лабораторных работ.	Компьютер, проектор, интерактивная доска.
10.	Индивидуальное задание	Разработка индивидуального или группового проекта. Защита индивидуального или групповых проектов.	Защита проекта	2	Самостоятельная индивидуальная или групповая проектная деятельность	Компьютер, проектор, интерактивная доска.
Итого:				72		

Содержание учебного плана

Структура программы:

Структура программы основа на модульном принципе.

Модуль 1. Введение в программирование.

Учащиеся познакомятся с основами программирования на языке Kotlin и средой разработки Android studio, напишут свои первые программы.

Модуль 2. Введение в объектно-ориентированное программирование. Учащиеся познакомятся с объектно-ориентированным программированием, парадигмами объектно-ориентированного программирования.

Модуль 3. Основы программирования Android приложений.

Учащиеся познакомятся с двух-мерной графикой и попрактикуются в разработке игровых приложений.

Модуль 4. Алгоритмы и структуры данных.

Учащиеся познакомятся с понятием массив, список, база данных.

Модуль 5. Основы разработки серверной части мобильных приложений.

Учащиеся разработают графические приложения.

Содержание занятий

Модуль1 Введение в программирование.

1. Здравствуй мир (Hello, World)!

Теория: Порядок создания, компиляции, сборки и запуска в IDE приложения Kotlin. Порядок инсталляции IDE в домашних условиях.

Практика: Знакомство с системой обучения в IT-Cube.

Первичное знакомство с интерфейсом Android Studio.

2. Типы данных и операции

Теория: переменные, примитивные типы данных, арифметические операторы, выражения и присваивания. Представление данных в памяти: представление числовых (в прямом коде) и символьных данных. Представление отрицательных и вещественных чисел в памяти. Представление чисел в обратном и дополнительном коде. Поразрядные операции.

Практика: Отработка способов представления данных.

3. Логические выражения.

Теория: Тип Boolean. Операции отношения, логические и тернарные операции
Практика: Отработка способов представления логических выражений.

4. Условные конструкции.

Теория: Понятие блока, область действия блоков. Условные конструкции: if- else, when

Практика: Отработка способов ветвления

5. Итеративные конструкции.

Теория: Итеративные конструкции while, do-while. Вложенные циклы. Одномерные массивы, цикл for each в Kotlin. Разбор примеров нахождения максимума и минимума, поиска на числовых массивах

Практика: Отработка применения данных итеративных.

6. Методы. Указатели.

Теория: Изучение понятия функций на примере методов Kotlin. Передача параметров, возвращение результата. Видимость переменных. Указатели на примере языка C. понятие, использование в качестве аргументов функций. Методы передачи аргументов в функцию. Динамическое выделение памяти

Практика: Отработка механизма методов. Отработка применения указателей

7. Многомерные массивы.

Теория: Многомерные массивы. Неровные массивы
Практика: Отработка применения многомерных массивов

8. Практикум

Практика: Закрепление изученных тем.

9. Контрольное тестирование по модулю. Практика: Проведение тестирования.

Модуль 2. Введение в объектно-ориентированное программирование.

1. Понятие класса и объекта.

Теория: Понятие класса и объекта. Цели и задачи ОО-подхода к проектированию и разработке ПО. Объект, сообщение, класс, экземпляр объекта, метод. Общее понятие о парадигмах ООП: абстракция, инкапсуляция, наследование, полиморфизм.

Практика: Описание протокола класса. Обзор классов, соответствующих примитивным типам

2. Классы: конструкторы, статические методы
Теория: Конструкторы и деструкторы. Статические методы. Открытые и закрытые поля. Доступ к полям объекта. Инициализация переменных, массивов, полей классов в конструкторе. Конструктор по умолчанию. Перегрузка методов на примере конструкторов
Практика: Работа с экземплярами класса
3. Строки. Начальные приёмы тестирования и отладки
Теория: Начальные приемы тестирования и отладки, сценарии тестирования на примерах со строками
Практика: Отработка указанных приемов
4. Знакомство с Android разработкой
Теория: ОС Android. Среда разработки. Принципиальная архитектура Android- приложения
Практика: Создание первого Android приложения в IDE.
Изучение жизненного цикла Activity
5. Интерфейс пользователя
Теория: Построение простейшего интерфейса пользователя. Язык разметки XML. Описание ресурсов Android с помощью XML. Разметки (Layouts) и их применение. Представления (Views)
Практика: Создание приложения с простейшим интерфейсом и обработкой событий
6. Наследование, инкапсуляция и полиморфизм
Теория: Производные классы и наследование. Защищенные части классов и правила доступа для классов и объектов в Kotlin. Сравнение иерархии классов и контейнеризации классов. Полиморфные методы и позднее связывание в Kotlin. Абстрактные методы и классы, интерфейсы.
Практика: Создание класса, хранящего информацию о человеке (возраст, имя). Реализация иерархии классов «Геометрические фигуры».
7. Context и Intent. Параметризованные типы
Теория: Понятия контекста (Context) и намерения (Intent) в Android. (Generic) как пример статического полиморфизма в Kotlin
Практика: Намерения (Intents) в Android. Использование Generic

Практикум

Практика: закрепление изученного
материала

Постановка задачи на индивидуальный
проект

Практика: Формирование постановки задачи на
индивидуальный проект (срок публикации в системе обучения
темы индивидуального проекта – окончание 2 модуля)

Контрольное тестирование по модулю

Практика: Проведение
тестирования

Модуль 3. Основы программирования Android приложений.

1. Практикум ООП проектирования.

Теория: Разбор кейсов проектирования архитектуры классов приложения. Диаграммы UML

Практика: Разбор задания минипроекта. Получение диаграммы классов мини- проекта

2. Ввод-вывод в Kotlin. Обработка исключений

Теория: Библиотечные классы ввода-вывода. Обработка исключений и классы исключений. Стандартные исключения Kotlin. Работа с файлами в Android

Практика: Работа с классом File, как пример необходимости обработки исключений.

3. Внутренние и анонимные классы

Теория: Внутренние и анонимные классы на примерах обработчиков событий пользовательского интерфейса

Практика: Разбор примеров использования Listener. Работа над мини-проектом

4. Параллелизм и синхронизация. Сервисы в Android.

Теория: Процессы и потоки в Android. Классы AsyncTask и Thread. Реализация логики потоков. Синхронизация потоков. Жизненный цикл сервисов и управление им. Класс IntentService

Практика: Разбор примера использования AsyncTask. Работа над мини- проектом. Разбор примеров

5. Фрагменты (Fragments). Сенсоры

Теория: Создание и управление фрагментами. Класс

Fragment и его методы. Взаимодействие фрагментов и активностей. Типы сенсоров и обработка событий Практика:

Разбор примера изменения интерфейса с фрагментами с учетом данных гироскопа

6. Двумерная графика

Теория: Двумерная графика в Android приложениях. Класс Canvas
Практика: Реализация простейшего приложения на Canvas

7. Разработка игровых приложений. Разработка 3D игр с использованием фреймворка libGDX

Теория: Этапы проектирования и реализации. Профессии в мире индустрии игр. Понятие игрового движка. Реализация графики на основе SurfaceView. Материал <https://habr.com/ru/post/276139/>

Практика: Разбор игры с анимацией на SurfaceView. Простейший пример с анимацией, HUD, обработкой касаний

Практикум

Практика: Закрепление изученного материала

Защита прототипа индивидуального проекта

Практика: Защита прототипа индивидуального проекта

Контрольное тестирование Практика: Проведение тестирования

Модуль 4. Алгоритмы и структуры данных.

1. Массив

Теория: Массив, как базовая структура данных. Алгоритм двоичного поиска: идея, применения, реализация. Классы Arrays (массивы), ArrayList (неограниченный массив)

Практика: Изучение класса Arrays: заполнение, копирование, сравнение, печать, методы поиска и др. общие методы. Изучение класса ArrayList. Создание итераторов, навигация

2. Список

Теория: Список, как базовая структура данных: стеки, очереди, односвязные и двусвязные списки

Практика: Изучение класса LinkedList, реализующему связанные списки

3. Адаптеры в Андроид

Теория: Назначение и применение. Стандартные адаптеры

Практика: Использование готовых адаптеров ArrayAdapter и SimpleAdapter для реализации ListView

Практика: Разбор задания на минипроект по БД в сформированном проекте. Проектирование схемы БД минипроекта

4. Дерево

Теория: Дерево, как базовая структура данных.

Сбалансированные деревья. Двоичные деревья

Практика: Практическое занятие по библиотечному классу TreeSet

5. Рекурсия

Теория: Линейная и ветвящаяся рекурсия. Стек вызовов

Практика: Пример использования при обходе дерева директорий

6. Алгоритмы сортировки

Теория: пузырьковая, вставкой и быстрая в сравнении трудоемкости. Компараторы. Практика: Компаратор, интерфейс Comparable. Практическое занятие по использованию методов класса Arrays, реализующих сортировку

7. Хэш-таблица и функция хэширования

Теория: Поддержка хэширования в Kotlin, метод hashCode. Семейства контейнеров Collections и Map.

Практика: Практическое занятие по библиотечным классам HashSet и TreeSet

8. Ассоциативные массивы. Контент-провайдеры. Введение в криптографию и криптоанализ. Введение в шифрование

Теория: Класс Map, контейнеры HashMap, TreeMap. Хранение данных в Android Preferences. Контент-провайдеры в Android. Назначение и применение. Стандартные контент-провайдеры ContactsContract, MediaStore. Простейшие алгоритмы подстановочного шифрования. Понятие ключа шифрования. Симметричные и асимметричные ключи. Алгоритм DES. Алгоритм RSA. Примеры использования.

Практика: Практическое занятие по библиотечным классам, реализующим ассоциативные контейнеры. Практика использования стандартных контент-провайдеров на примере контактов и медиа-файлов. Практическое занятие на распознавание сообщения, зашифрованного подстановочным шифром. Практическое занятие на освоение ключей шифрования.

9. Реляционная модель данных

Теория: Реляционная модель данных и реляционные схемы. Представление данных в виде таблиц. Типы связей: один к одному, один ко многим, многие-ко- многим. Проектирование простейшей БД. Необходимость возникновения и история развития СУБД. Обзор и классификация современных СУБД

10. Локальная СУБД

Теория: Локальная СУБД на примере SQLite. Введение в SQL. Создание и наполнение таблиц: команды CREATE, INSERT. Команда SELECT для выборки данных, UPDATE для изменения, DELETE для удаления записей из таблицы. Дополнения к запросам SELECT. Ключевые слова ORDER BY, DISTINCT. Агрегация в SELECT в запросах: COUNT, SUM, AVG, MAX, MIN

Практика: Знакомство с SQLite. Разбор заготовки Android приложения.

Реализация изученных команд SQL в сформированном.

Практикум

Практика: Закрепление изученного материала.

Предзащита проектов

Практика: Предзащита индивидуального проекта

Контрольное тестирование

Практика: проведение тестирования.

Модуль 5. Основы разработки серверной части мобильных приложений.

1. IP-сети

Теория: Адресация в IP-сетях. IPv4. Автоматизация назначения IP-адресов (DHCP). Доменные имена (DNS), URL-ссылки. Несколько IP адресов для одного сайта. Популярные сетевые команды ping, tracert, ipconfig. Сервисы работы с IP-адресами.

Практика: Работа с сетевыми командами и сервисами

2. Web сервер. HTTP запросы и ответы

Теория: Протокол HTTP. Понятие web-сервера. Запросы клиента POST и GET, коды ответов сервера, заголовки запросов и ответов. Разбор запросов и ответов сервера.

Практика: Отправка запроса на север с помощью формы из браузера.

Обработка методов на сервере средствами Kotlin, PHP.

3. Клиент-серверная архитектура мобильных приложений

Теория: Клиент-серверная архитектура мобильных приложений: структура, схема взаимодействия сервера и клиента мобильного приложения. Формат JSON и XML. Сериализация. Библиотека Retrofit.

Практика: Отправка запросов из Android приложения. Реализация сервера средствами Kotlin, PHP.

4. Облачные платформы. REST взаимодействие. Серверные СУБД. Material Design

Теория: Облачные сервисы для хостинга серверной части приложений. Реализация серверной части средствами Kotlin, PHP. Стилль взаимодействия REST. Хранение данных на сервере в виде БД. Индексы. Знакомство с серверной СУБД на примере PostgreSQL/MySQL. Реализация back-end части web сервера средствами Kotlin, PHP. Фреймворки Spring (опционально), Дизайн программного обеспечения и приложений Material Design

Практика: Реализация Android приложения REST. Реализация БД в СУБД

PostgreSQL на <https://www.heroku.com/> и Android клиента к ней. Разбор примеров.

Контрольное тестирование по модулю

1. Практикум. Работа над индивидуальным проектом

Практика: Закрепление изученного материала. Работа над индивидуальным проектом

Консультации по индивидуальному проекту. Практика: Репетиция защит, ответы на вопросы. Защита индивидуального проекта.

Защита ИП, подведение итогов.

7. Планируемые результаты

Предметные: Получение знаний:

- формировать общее представление о создании мобильных приложений на базе платформы Андроид;
- формировать представления о структуре и функционировании среды App Inventor;
- формировать умения и навыки построения различных видов алгоритмов с помощью блоков в среде АИ;
- формировать умение использовать компоненты, блоки и их комбинации в среде АИ для создания мобильных приложений;
- формировать умения создавать типовые мобильные приложения на базе компонент среды АИ;
- формировать ключевые компетенции проектной и исследовательской деятельности.

Метапредметные:

- формирование умения ориентировки в системе знаний;
- формирование умения выбора наиболее эффективных способов решения задач на компьютере в зависимости от конкретных условий;
- формирование приёмов проектной деятельности, включая умения видеть проблему, формулировать тему и цель проекта, составлять план своей деятельности, осуществлять действия по реализации плана, соотносить результат своей деятельности с целью, классифицировать, наблюдать, проводить эксперименты, делать выводы и заключения, доказывать, защищать свои идеи, оценивать результаты своей работы;
- формирование умения распределения времени;
- формирование умений успешной самопрезентации.

Личностные:

- формирование умения самостоятельной деятельности;
- формирование умения работать в команде;
- формирование коммуникативных навыков;
- формирование навыков анализа и самоанализа;
- формирование эстетического отношения к языкам программирования, осознание их выразительных возможностей;
- формирование целеустремленности и усидчивости в процессе творческой, исследовательской работы и учебной деятельности.

Познавательные:

Сформировать умение работать с источниками информации; сформировать умение самостоятельно определять цели своего обучения.

Коммуникативные:

Сформировать умение организовать учебное сотрудничество и совместную деятельность с педагогом и сверстниками; сформировать умение работать индивидуально и в группе, уметь вступать в контакт со сверстниками.

Социальные:

Сформировать умение пользоваться приемами коллективного творчества; сформировать умение эстетического восприятия мира и доброе отношение к окружающим.

Развивающие:

Развить творческую активность; развить умение представлять результаты своей работы окружающим, аргументировать свою позицию; развить аналитическое, практическое и логическое мышление; развить самостоятельность и самоорганизацию; развить умение работать в команде; развить коммуникативные навыки; развить познавательную активность.

8. Методические рекомендации по проведению уроков

Трудно представить современный мир без мобильных устройств и разного рода гаджетов. То, что казалось ещё 20 лет назад фантастикой, сейчас распахнуло двери и стремительно врывается в наш мир, который даже по человеческим меркам ещё совсем недавно пользовался дисковыми телефонными аппаратами. А сейчас всё вокруг неумолимо и стремительно переходит к новому технологическому укладу. Согласно Элвину Тоффлеру, следующий мировой технологический и социальный уклад установит ценность человеческого ума и талантов как высший приоритет. При этом новом укладе мобильные устройства являются не только предтечей и воплощением будущего, и должны быть не только инструментом постижения мира, но и проводником, способствующим нашей трансформации. Посредством этих инструментов человечество должно преодолеть непростые ступени нового мира и застолбить своё место в грядущем новом мире.

Первые мобильные приложения появились еще в далёком 1993 году. А первый мобильный телефон появился за 20 лет до этого, в ещё более далёком 1973 году, когда 3 апреля два инженера-разработчика компаний Bell Labs и Motorola осуществили первый разговор.

На сегодняшний день мир мобильной разработки представлен двумя основными операционными системами и технологиями на их базе: Android и iOS. С большим отрывом превалирует Android.

Знания и умения в области разработки мобильных приложений в ближайшем будущем будут востребованы уже в рядах специальностей, перспективных профессиях. В сложившихся условиях возникает необходимость формирования единого методического инструментария, который соответствует следующему набору критериев:

- Модульное представление материала с возможностью строить как обзорные уроки, так и углубленно рассматривать некоторые темы.
- Возможность частичного использования материалов в упрощенном виде в рамках интегрированных уроков по разным предметам.
- Представление адаптированного материала к различным возрастным категориям.
- Примерные материалы, на базе которых возможно построение занятий.

База знаний в удобном для использования виде с возможностью оперативного изменения и дополнения с учетом развития современных информационных технологий.

В рамках подготовки к урокам важно помнить о том, что все соответствующие материалы должны соответствовать следующим дидактическим принципам:

- Активной вовлеченности;
- Доступности;
- Мотивации;
- Рефлексивности;
- Системности;
- Открытости содержания.

Под этим подразумевается, что в процессе изучения материала происходит обращение к личному опыту ребенка и развитие этого опыта на основе получения новых знаний или систематизации имеющихся. При этом подача материала должна учитывать возрастные характеристики участников занятия, их социальный статус и жизненный опыт, а также уровень полученных в процессе обучения знаний и иметь форму, которая будет стимулировать к использованию полученных знаний в повседневной жизни, подталкивать к самостоятельному поиску новой информации. В комплексе это дает ребенку возможность соотнести полученные знания и собственный опыт, корректировать модели собственного поведения.

Структурированная информация, представленная в форме простых правил и лаконичных формулировок, как основа новых знаний, дополняет и уточняет единую информационную картину, а также предполагает, что преподаватель имеет возможность свободного частичного или полного использования существующих материалов, а также их актуализации.

При подготовке к фактическому занятию преподаватель на основе методических рекомендаций и дидактических материалов создаёт собственное занятие, дополняя и расширяя его собственными методическими наработками.

8.1. Форма аттестации.

Педагогический мониторинг включает в себя: текущий контроль, промежуточную аттестацию, итоговую аттестацию.

Текущий контроль осуществляется регулярно в течение учебного года.

Контроль теоретических знаний осуществляется с помощью педагогического наблюдения, тестов, опросов, дидактических игр. В практической деятельности результативность оценивается качеством выполнения работ учащихся, где анализируются положительные и отрицательные стороны работ, корректируются недостатки. Формы контроля – фронтальная и индивидуальная беседа, выполнение практических заданий, участие в конкурсах и выставках технической направленности, защиты проектов и т.д.

Система контроля знаний и умений учащихся представляется в виде учёта результатов по итогам выполнения заданий отдельных кейсов и посредством наблюдения, отслеживания динамики развития учащегося.

Таблица 2 - Критерии оценивания учащихся.

№ п/п	Ф.И.О. учащегося	Сложность продукта (от 0 до 5)	Соответствие продукта поставленной задаче (от 0 до 5)	Презентация продукта. Степень владения специальными терминами (от 0 до 5)	Степень увлечённости продуктом и стремление к оригинальности (от 0 до 5)
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					

В конце учебного года учащиеся проходят защиту индивидуальных/групповых проектов. Индивидуальный/групповой проект оценивается формируемой комиссией. Состав комиссии не менее 3-х человек.

Оценочный лист результатов предварительной аттестации учащихся.

Срок проведения: декабрь, май.

Цель: оценка роста качества знаний и практического их применения за период обучения.

Форма проведения: практическое задание, контрольное занятие, отчетные мероприятия (соревнования, конкурсы и т.д.).

Содержание аттестации: сравнительный анализ качества выполненных работ начала и конца учебного года (выявление уровня знаний и применения их на практике). Форма оценки: уровень (высокий, средний, низкий).

Таблица 3 – Оценочный лист.

№ п/п	Параметры оценки	Критерии оценки		
		Низкий уровень	Средний уровень	Высокий уровень

1	Техническое исполнение	Не умение самостоятельно составить программу	Лёгкий уровень составленных программ, ошибки в построении алгоритмов	Использование сложных технологических приёмов (условные алгоритмы, переменные, списки, подпрограммы)
2	Творческое исполнение	Отсутствие творческого подхода	Творческий замысел воплощён частично	В работе воплощён творческий замысел.
3	Личностный рост	Не усидчивость, не умение работать самостоятельно	Слабая усидчивость, не полная самостоятельность	Самостоятельность в работе, дисциплинированность, аккуратность
4	Личностные достижения (участие в конкурсах)	Не участвовал	Участие без призового места	Работа заняла призовое место

8.2. Методическое обеспечение.

В образовательном процессе используются следующие методы:

- объяснительно-иллюстративный;
- метод проблемного изложения (постановка проблемы и решение её самостоятельно или группой);
- проектно-исследовательский;
- наглядный: демонстрация плакатов, схем, таблиц, диаграмм, использование технических средств, просмотр видеороликов;
- практический: практические задания, анализ и решение проблемных ситуаций и т. д.

Выбор методов обучения осуществляется исходя из анализа уровня готовности учащихся к освоению содержания модуля, степени сложности материала, типа учебного занятия. На выбор методов обучения значительно влияет персональный состав группы, индивидуальные особенности, возможности и запросы детей.

Формы обучения: фронтальная – предполагает работу педагога сразу со всеми учащимися в едином темпе и с общими задачами. Для реализации обучения используется компьютер педагога, интерактивный комплекс, посредством которых, учебный материал демонстрируется всей группе.

Занятия проводятся с применением следующих методических материалов: методические рекомендации, дидактический материал (игры; сценарии; задания, задачи, способствующие «включению» внимания, восприятия, мышления, воображения учащихся), учебно-планирующая документация (рабочие программы),

диагностический материал (кроссворды, анкеты, тестовые и кейсовые задания), наглядный материал, аудио и видео материал.

7. Перечень доступных источников информации

1. Язык Kawa (на англ.языке) [Электронный ресурс] URL: <https://www.gnu.org/software/kawa/index.html> (дата обращения: 19.03.2021).
2. Установка эмулятора (на англ.языке) [Электронный ресурс] URL: <http://appinventor.mit.edu/explore/ai2/setup-emulator> (дата обращения: 19.03.2021).
3. Установка эмулятора в ОС Windows (на англ.языке) [Электронный ресурс] URL: <http://appinventor.mit.edu/explore/ai2/windows> (дата обращения: 19.03.2021).
4. AITech - Using Procedures and Any component blocks (на англ.языке) [Электронный ресурс] URL: <https://appinventor.mit.edu/explore/blogs/karen/2016/07-0.html> (дата обращения: 19.03.2021).
5. Процедуры в АИ (на англ.языке) [Электронный ресурс] URL: <https://appinventor.mit.edu/explore/ai2/support/concepts/procedures> (дата обращения: 19.03.2021).
6. База данных TinyDB (на англ.языке) [Электронный ресурс] URL: <https://tinydb.readthedocs.io/en/latest/> (дата обращения: 19.03.2021).
7. Игра Пианино (на англ.языке) [Электронный ресурс] URL: https://drive.google.com/drive/folders/1f9D_bQPу-G17EmdPCpY3-KoKAfH1E7qE (дата обращения: 19.03.2021).
8. Игра «Найди золото» (на англ.языке) [Электронный ресурс] URL: https://drive.google.com/drive/folders/1xRSZGMLmtU7nJn22ToWCZIC92Z_bPaEF (дата обращения: 19.03.2021).
9. Инструкции по установке USB соединения (на англ.языке) [Электронный ресурс]