

Республика Бурятия Мухоршибирский район

МБОУ «Мухоршибирская СОШ №1»

Центр цифрового образования детей IT-CUBE.МУХОРШИБИРЬ

Утверждаю:

Директор МБОУ «Мухоршибирская СОШ №1»

О.А. Мункуева

01.09.2023 г.

Рабочая программа

Дополнительной общеобразовательной общеразвивающей программы

по направлению

«Мобильная разработка» 1 год обучения

Возраст учащихся: 14-18 лет

Срок реализации: 2023-2024 уч.год

Разработчик(и) : педагог ДО
Федоров Андрей Васильевич

с. Мухоршибирь, 2023 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

В современном мире мы можем наблюдать феномен внедрения информационных технологий в жизнь общества, семьи. Возможности и удобства, которые предоставляются в этой области, создают объективные предпосылки для использования информационных технологий в сфере образования, работы и отдыха.

Программирование способствует развитию мышления, логики, коммуникативных способностей, развивает навыки взаимодействия, самостоятельности при принятии решений, раскрывает творческий потенциал. Направление « разработка мобильных приложений» предполагает знакомство с основными понятиями программирования, решение большого количества творческих задач.

Разработка приложений на AppInventor под Android набирает популярность среди разработчиков.

Направленность программы:

Дополнительная общеобразовательная (общеразвивающая) программа «Разработка мобильных приложений» (далее - программа), является технической направленностью и предназначена для использования в системе дополнительного образования детей. Программа содержит профориентационную работу с учащимися к профессии разработчик мобильных приложений.

Актуальность программы:

С каждым годом растёт потребность в мобильных приложениях, так например количество покупок с мобильных телефонов на данный момент превышает 50% от общего числа онлайн покупок, а мобильные игры занимают свыше половины общего рынка гейминга. Таким образом, сфера мобильной разработки является актуальной и стабильно растёт, поэтому специалисты нужны постоянно. Причём выбор направлений для специалистов очень широк: от серьёзных проектов до разработки мобильных игр и приложений.

Особенность программы «Мобильная разработка» - в изучении основ языка программирования AppInventor и структуры приложения под ОС Android. Она строится в доступной и понятной для учащихся среде, т. е. программирование ведётся в текстово-графическом режиме, что позволяет сразу задавать необходимый функционал для элементной базы приложения.

Актуальность программы обусловлена необходимостью вернуть интерес детей и подростков к научно-техническому творчеству, так как в России наблюдается острая нехватка инженерных кадров.

Педагогическая целесообразность:

Программа развивает логическое и алгоритмическое мышление, активизирует интерес к техническому творчеству.

Также учащиеся научатся основам программирования, получат навыки разработки приложений различной направленности.

ЦЕЛЬ ПРОГРАММЫ

Целью данной программы является научить обучающихся разрабатывать мобильные приложения на базе интегрированных сред разработки, получить практический опыт в данной сфере; развивать алгоритмический и объектный стиль мышления; формировать мотивацию к получению образования в ИТ-сфере посредством организации практической деятельности.

Задачи

Обучающие (предметные):

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

- Сформировать навыки работы в интегрированной среде разработки AppInventor.
- Изучить основы программирования Android приложений.
- Ознакомить с основами разработки серверной части мобильных приложений.
- Развивающие (метапредметные):
- Развить умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности.
- Развить умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая.
- Развить умение критически оценивать правильность решения учебно-исследовательской задачи.
- Развить умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями. • Сформировать владение основами самоконтроля, способность к принятию решений.
- Развить умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ.
- Сформировать компетентность в области использования информационно-коммуникационных технологий (ИКТ-компетенция).
- Развить умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.
- Воспитательные (личностные):
- Сформировать ответственное отношение к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам.
- Сформировать способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию.
- Развить опыт участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам.
- Сформировать коммуникативную компетенцию в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня
- Сформировать целостное мировоззрение, соответствующее современному уровню развития информационных технологий.
- Сформировать осознанное позитивное отношение к другому человеку, его мнению, результату его деятельности.
- Сформировать ценность здорового и безопасного образа жизни.
- Обеспечить усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Адресат программы Программа рассчитана на учащихся 14 лет и старше.

Численность обучающихся

Группы 1 года обучения формируются из расчета - до 12 человек.

Условия набора

Принимаются учащиеся, обладающие начальным уровнем компьютерной грамотности. Возраст учащихся внутри одной группы может не совпадать.

Форма обучения

Программа реализуется в очной форме, в том числе с возможностью использования дистанционных образовательных технологий и электронного обучения.

Режим занятий

Занятия проводятся 2 раза в неделю по 2 урока. Продолжительность занятия - 40 минут. После 40 минут занятий организовывается перерыв длительностью 10 минут для проветривания помещения и отдыха учащихся.

Срок реализации программы: Программа рассчитана на 1 год, общее количество часов - 144 часа.

Особенности организации образовательного процесса

Научность. Этот принцип предопределяет сообщение обучаемым только достоверных, проверенных практикой сведений, при отборе которых учитываются новейшие достижения науки и техники.

Доступность. Предусматривает соответствие объема и глубины учебного материала уровню общего развития учащихся в данный период, благодаря чему, знания и навыки могут быть сознательно и прочно усвоены.

Связь теории с практикой. Обязывает вести обучение так, чтобы обучаемые могли сознательно применять приобретенные ими знания на практике.

Воспитательный характер обучения. Процесс обучения является воспитывающим, ученик не только приобретает знания и нарабатывает навыки, но и развивает свои способности, умственные и моральные качества.

Сознательность и активность обучения. В процессе обучения все действия, которые отрабатывает ученик, должны быть обоснованы. Нужно учить, обучаемых, критически осмысливать, и оценивать факты, делая выводы, разрешать все сомнения с тем, чтобы процесс усвоения и наработки необходимых навыков происходили сознательно, с полной убежденностью в правильности обучения. Активность в обучении предполагает самостоятельность, которая достигается хорошей теоретической и практической подготовкой и работой педагога.

Наглядность. Наглядная демонстрация существующих и проектируемых программных продуктов, а также связанных с этим материалов. Для наглядности применяются, как уже, существующие методические материалы, так и материалы своего изготовления.

Систематичность и последовательность. Учебный материал дается по определенной системе и в логической последовательности с целью лучшего его освоения. Как правило этот принцип предусматривает изучение предмета от простого к сложному, от частного к общему.

Прочность закрепления знаний, умений и навыков. Качество обучения зависит от того, насколько прочно закрепляются знания, умения и навыки учащихся. Не прочные знания и навыки обычно являются причинами неуверенности и ошибок. Поэтому закрепление умений и навыков должно достигаться неоднократным целенаправленным повторением и тренировкой.

Индивидуальный подход в обучении. В процессе обучения педагог исходит из индивидуальных особенностей детей (уравновешенный, неуравновешенный, с хорошей памятью или не очень, с устойчивым вниманием или рассеянный, с хорошей или замедленной реакцией, и т.д.) и опираясь на сильные стороны ребенка, доводит его подготовленность до уровня общих требований.

Формы проведения образовательного процесса

- вводные занятия;
- регулярные групповые занятия;
- индивидуальные занятия;
- открытые занятия;
- конференции, соревнования, конкурсы, выставки;
- беседы (тематические, а также по технике безопасности);

Таблица 1

Методы	Формы	Приемы
Исследование готовых знаний	Поиск материалов, систематизация знаний, лекций	Работа с методической и периодической литературой.
Метод Объяснительно иллюстративный	Лекции, беседы, рассказы, демонстрации	Беседа: «Применение компьютеров в жизни человека»
Метод репродуктивный	Воспроизведение приемов действий, применение знаний на практике	Практическая работа по разным направлениям
Метод творческих проектов	Поисковая и творческая деятельность	Самостоятельная разработка модели
Метод проверки знаний и умений	Игры, выставки по разделам	Викторина по пройденным темам

Формы подведения итогов

Форма итогового контроля - экспертная оценка педагогом результативности каждого учащегося по итогам освоения всех тем программы. Презентация и защита собственного проекта. По итогам заполняется информационная карта "Итоговая оценка результативности образовательного процесса":

Таблица 2

№	Фамилия, имя	1	2	3	4	5	6			Итог

Оценка производится по 5-балльной шкале: "5" - отлично, "4" - хорошо, "3" - посредственно, "2" - плохо.

Ожидаемый результат

Обучающие (предметные):

- Сформированы и развиты навыки алгоритмического и логического мышления, грамотной разработки программ.
- Изучены принципы и методы функционального программирования.
- Изучены принципы и методы объектно-ориентированного программирования.
- Сформированы навыки работы в интегрированной среде разработки AppInventor
- Изучены основы программирования Android приложений.
- Изучены основы разработки серверной части мобильных приложений.

Развивающие (метапредметные):

- Развито умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности.
- Развито умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая.
- Развито умение критически оценивать правильность решения учебно- исследовательской задачи.
- Развито умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями.
- Сформировано владение основами самоконтроля, способность к принятию решений.

- Развито умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ.
- Сформирована компетентность в области использования информационно-коммуникационных технологий (ИКТ-компетенция).
- Развито умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Воспитательные (личностные):

- Сформированы ответственное отношение к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам.
- Сформированы способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию.
- Развиты опыт участия в социально значимых проектах, повышен уровень самооценки благодаря реализованным проектам.
- Сформирована коммуникативная компетенция в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня
- Сформировано целостное мировоззрение, соответствующее современному уровню развития информационных технологий.
- Сформировано осознанное позитивное отношение к другому человеку, его мнению, результату его деятельности.
- Сформирована ценность здорового и безопасного образа жизни.
- Усвоены правила индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

СОДЕРЖАНИЕ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ

Структура программы:

Структура программы основана на модульном принципе. **Модуль 1. Введение в программирование.**

Учащиеся познакомятся с основами программирования на визуальном языке и средой разработки AppInventor, 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.

Учебный план

Таблица 3

№ п/п	Название раздела, темы	Кол-во часов			Форма аттестации/контроля
		всего	теория	практика	
Модуль 1. Основы программирования.		24	8	16	
1.1	Вводное занятие. Инструктаж по ТБ. Введение в программу. Среда разработки. Здравствуй мир! (AppInventor)	2	1	1	Знакомство. Опрос. Инструктаж по ТБ
1.2	Примитивные типы данных. Арифметика	2	1	1	Устныйопрос
1.3	Операции отношения и логические операции	2	1	1	Устныйопрос
1.4	Условные конструкции. Блоки	2	1	1	Устныйопрос
1.5	Итеративные конструкции	4	2	2	Устныйопрос
1.6	Методы (функции). Видимость переменных	2	1	1	Устныйопрос
1.7	Многомерные и неровные массивы	2	1	1	Устныйопрос
1.8	Практикум	6	0	6	Устныйопрос

1.9	Контрольное тестирование модулю по	2	0	2	Тест № 1
Модуль 2. Объектно-ориентированное программирование		26	11	15	
2.1	Классы и объекты	2	1	1	Устный опрос
2.2	Классы: конструкторы, статические методы	4	2	2	Устный опрос
2.3	Строки. Начальные приёмы тестирования и отладки	2	1	1	Устный опрос
2.4	Знакомство с Android разработкой	2	1	1	Устный опрос
2.5	Интерфейс пользователя.	4	2	2	Устный опрос
2.6	Наследование, инкапсуляция и полиморфизм в Kotlin	4	2	2	Устный опрос
2.7	Context и Intent. Параметризованные типы	2	1	1	Разработанные программы
2.8	Практикум	2	0	2	
2.9	Постановка задачи на индивидуальный проект	2	1	1	
2.10	Контрольное тестирование модулю по	2	0	2	Тест № 2
Модуль 3. Основы программирования Android-приложений		28	8	20	
3.1	Практикум ООП проектирования	6	0	6	Разработанные программы
3.2	Ввод-вывод в Kotlin. Обработка исключений	2	1	1	Устный опрос
3.3	Внутренние классы в обработке событий	4	2	2	Разработанные программы
3.4	Параллелизм и синхронизация. Сервисы в Android	2	1	1	Устный опрос
3.5	Фрагменты	2	1	1	Устный опрос

	(Fragments). Сенсоры				
3.6	Двумерная графика в Android-приложениях	2	1	1	Устныйопрос
3.7	Разработка игровых приложений. Разработка 3D игр с использованием фреймворка libGDX	4	1	3	Разработанные программы
3.8	Практикум	2	0	2	Разработанные программы
3.9	Защита прототипа индивидуального проекта	2	1	1	
3.10	Контрольное тестирование по модулю	2	0	2	Тест № 3
Модуль 4. Алгоритмы и структуры данных		36	13	23	
4.1	Массивы. Алгоритм двоичного поиска	4	2	2	Разработанные программы
4.2	Списки	4	2	2	Разработанные программы
4.3	Адаптеры в Android	2	1	1	Устныйопрос
4.4	Деревья	2	1	1	Устныйопрос
4.5	Рекурсия	2	1	1	Устныйопрос
4.6	Алгоритмы сортировки	2	1	1	Устныйопрос
4.7	Хэш-таблица и функция хэширования	2	1	1	Устныйопрос
4.8	Ассоциативные массивы	2	1	1	Устныйопрос
4.9	Реляционная модель данных. СУБД	2	1	1	Устныйопрос
4.10	Локальные СУБД. Введение в SQL	4	2	2	Разработанные программы
4.11	Практикум	4	0	4	Разработанные программы
4.12	Предзащита проектов	4	0	4	
4.13	Контрольное тестирование по модулю	2	0	2	Тест № 4
Модуль 5.		28	9	19	

Алгоритмы и структуры данных					
5.1	IP-сети	2	1	1	Устный опрос
5.2	Web-сервер. HTTP-запросы и ответы	4	2	2	Разработанные программы
5.3	Клиент-серверная архитектура мобильных приложений	4	2	2	Разработанные программы
5.4	Облачные платформы. REST взаимодействие. Серверные СУБД. Material Design	4	2	2	Разработанные программы
5.5	Практикум. Работа над индивидуальным проектом	6	2	4	Разработанные программы
5.6	Контрольное тестирование по модулю	2	0	2	Тест № 5
6.	Консультации по ИП	4	0	4	
7.	Итоговая защита	4	0	4	
Итого		144	49	95	

Содержание занятий

Модуль 1.

1. Здравствуй мир (Hello, World)!

Теори

я: 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.**
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

8. Практикум

Практика: закрепление изученного материала

9. Постановка задачи на индивидуальный проект

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

10. Контрольное тестирование по модулю

Практика: Проведение тестирования.

Модуль 3.

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

8. Практикум

Практика: Закрепление изученного материала

9. Защита прототипа индивидуального проекта

Практика: Защита прототипа индивидуального проекта 10.

Контрольное тестирование

Практика: Проведение тестирования

Модуль 4.

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.

11. Практикум

Практика: Закрепление изученного материала.

12. Предзащита проектов

Практика: Предзащита индивидуального проекта

13. Контрольное тестирование

Практика: проведение тестирования.

Модуль 5.

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, tracer, 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 сервер 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, PHP.

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

Теория: Облачные сервисы для хостинга серверной части приложений. Реализация серверной части средствами Kotlin, PHP. Стиль 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.

5. Контрольное тестирование по модулю

6. 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.

Календарный учебный график

	Дата начала обучения по программе	Дата окончания обучения по программе	Всего учебных недель	Количество учебных часов	Режим занятий
	01.09.2022	31.05.2023	36	144	академических часов

Рабочая программа группы 1 года обучения

№ пп	Дата проведения	Название темы	Общее кол-во часов
1	2	3	4
СЕНТЯБРЬ			
1		Вводное занятие. Инструктаж по ТБ. Введение в программу. Среда разработки. Здравствуй мир! (Kotlin). Прimitives типы данных. Арифметика	4
2		Операции отношения и логические операции. Условные конструкции. Блоки.	4
3		Итеративные конструкции. Методы (функции). Видимость переменных.	4
4		Многомерные и неровные массивы. Практикум	4
			16 часов
ОКТАБРЬ			
5		Практикум. Контрольное тестирование по модулю.	4
6		Классы и объекты. Классы: конструкторы.	4
7		Статические методы. Строки. Начальные приёмы тестирования и отладки.	4
8		Знакомство с Android разработкой. Интерфейс пользователя.	4
			16 часов
НОЯБРЬ			
9		Наследование, инкапсуляция и полиморфизм в Kotlin	4
10		Context и Intent. Параметризованные типы. Практикум.	4
11		Постановка задачи на индивидуальный проект. Контрольное тестирование по модулю.	4
12		Практикум ООП проектирования.	4
13		Практикум ООП проектирования. Ввод-вывод в Kotlin. Обработка исключений.	4
			20 часов

ДЕКАБРЬ			
14		Внутренние классы в обработке событий.	4
15		Параллелизм и синхронизация. Сервисы в Android. Фрагменты (Fragments). Сенсоры	4
16		Двумерная графика в Android-приложениях. Разработка игровых приложений.	4
17		Разработка 3D игр с использованием фреймворка libGDX. Практикум.	4
			16 часов
ЯНВАРЬ			
18		Защита прототипа индивидуального проекта. Контрольное тестирование по модулю.	4
19		Массивы. Алгоритм двоичного поиска.	4
			8 часов
ФЕВРАЛЬ			
20		Списки	4
21		Адаптеры в Android. Деревья.	4
22		Рекурсия. Алгоритмы сортировки	4
23		Хэш-таблица и функция хэширования. Ассоциативные массивы.	4
			16 часов
МАРТ			
24		Реляционная модель данных. СУБД	4
25		Локальные СУБД. Введение в SQL.	4
26		Практикум.	4
27		Предзащита проектов.	4
28		Контрольное тестирование по модулю.	4
			20 часов
АПРЕЛЬ			
29		IP-сети	4
30		Web-сервер. HTTP-запросы и ответы	4
31		Клиент-серверная архитектура мобильных приложений	4

32		Облачные платформы. REST взаимодействие. Серверные СУБД. Material Design	4
			16 часов
МАЙ			
33		Практикум.	4
34		Работа над индивидуальным проектом. Контрольное тестирование.	4
35		Консультации по ИП.	4
36		Итоговая защита. Подведение итогов.	4
			16 часов
		ИТОГО	144 часа

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Средства обучения.

Демонстрационный материал

- Тематическая подборка презентационного материала по темам
- Примеры показательных результатов разработки мобильных приложений

Наглядные пособия:

- Видеоматериалы по тематике мобильной разработки
- Блок-схемы алгоритмов

Формы и методы организации учебно-воспитательного процесса

В рамках данной программы определены приоритетные формы и методы организации учебно-воспитательного процесса:

- классно-урочная система обучения с упором на практические занятия,
- элементы проектно-исследовательской деятельности,
- проведение экспериментов,
- соревновательные элементы.

В процессе обучения используется следующие оценочные материалы:

- карта самооценки учащегося - два раза в год (декабрь, май);
- карты «Оценка результативности образовательного процесса» - по итогам тем;
- карта «Оценка результативности выполнения собственного проекта - один раз в год;
- карта "Итоговая оценка результативности образовательного процесса" - по окончании обучения по программе - один раз в год(май).

Материально-техническое обеспечение

Для реализации данного курса требуется следующее оборудование:

- Проектор и экран для демонстрации учебного материала
- Доска
- Персональные компьютеры для обучающихся Требуемое программное обеспечение:
- Среды разработки Android Studio
- Пакет офисных приложений
Браузер Google Chrome, Mozilla Firefox или «Яндекс Браузер»

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. IT-куб. [Электронный ресурс] URL: <http://айтикуб.рф/> [Дата обращения: 1.09.2021]
2. IT-куб Хабаровск. Центр цифрового образования детей. [Электронный ресурс] URL: <https://itcube.kco27.ru/> [Дата обращения: 1.09.2021]
3. Информационные технологии в нашей жизни. [Электронный ресурс] URL: <http://iteach.vspu.ru/07-2019/19758/> [Дата обращения: 9.09.2021]
4. Хабр: Профессия: Мобильный разработчик. [Электронный ресурс] URL: <https://habr.com/ru/company/ruvds/blog/495422/> [Дата обращения: 9.09.2021]
5. Документы [Электронный ресурс] URL: <https://kotlinlang.org/docs/home.html> [Дата обращения: 1.09.2021]