

ОГЛАВЛЕНИЕ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.	3
УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН.....	11
СОДЕРЖАНИЕ ПРОГРАММЫ	14
МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ПРОГРАММЫ	25
СПИСОК ЭЛЕКТРОННЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ.....	28

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

В настоящее время процесс информатизации проявляется во всех сферах человеческой деятельности. Использование современных информационных технологий является необходимым условием успешного развития как отдельных отраслей, так государства в целом. Создание, внедрение, эксплуатация, а также совершенствование информационных технологий немыслимо без участия квалифицированных и увлеченных специалистов. Стремительный рост информационных технологий ставит новые задачи перед образованием и наукой, изучение классических дисциплин недостаточно для решения таких задач. В связи с этим актуальной задачей является подготовка специалистов сферы информационных технологий в соответствии с профессиональными требованиями динамично развивающихся отраслей. При этом требуется постоянная актуализация знаний, приобретения новых компетенций, формирование нового типа мышления. В этом смысле важнейшую роль играет процесс изучения основ информационных технологий еще в школьном возрасте.

Направленность программы

Программа имеет инженерно-техническую направленность, в связи с этим рассматриваются следующие аспекты изучения:

1. Технологический. Содержание программы рассматривается как средство формирования образовательного потенциала, позволяющего развивать наиболее передовые на сегодняшний день технологии — информационные, интегрирующие в себе науку, технологию, инженерное дело.

2. Общеразвивающий. Обучение по данной программе создает благоприятные условия для интеллектуального и духовного воспитания личности ребенка, социально-культурного и профессионального самоопределения, развития познавательной активности и творческой самореализации учащихся.

3. Общеобразовательный. Содержание программы рассматривается

как средство развития основных познавательных процессов, умения анализировать, выявлять сущности и отношения, описывать планы действий и делать логические выводы, опираясь на такие дисциплины, как механика, теория управления, программирование, теория информации.

Актуальность программы

Актуальность данной программы состоит в том, что она составлена с учётом современных потребностей рынка в специалистах в области информационных технологий, а именно в сфере мобильной разработки.

Данная программа дает возможность детям творчески мыслить, находить самостоятельные индивидуальные решения, а полученные умения и навыки применять в жизни. Развитие творческих способностей помогает также в профессиональной ориентации подростков.

Современное информационное общество требует постоянного обновления и расширения профессиональных компетенций. Необходимо улавливать самые перспективные тенденции развития мировой конъюнктуры, шагать в ногу со временем. В процессе реализации данной программы формируются и развиваются умения и навыки в области информационных технологий, новые компетенции, которые необходимы всем для успешности в будущем.

Новизна программы

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, а также использует новые формы диагностики и подведения итогов реализации программы, выполняемые в формате защиты проектов и участия во Всероссийском конкурсе мобильных приложений.

Введение в дополнительное образование общеобразовательной и общеразвивающей программы «Мобильная разработка» с использованием таких методов, как поиск проблем и их практическое решение, анализ и

обобщение опыта, подготовка инженерно-технических проектов и их защита, элементы соревнований, неизбежно изменит картину восприятия учащимися технических дисциплин, переводя их из разряда умозрительных в разряд прикладных.

Цели программы:

- привлечение обучающихся к изобретательской деятельности в инженерно-техническом направлении;
- повышение интереса обучающихся к информационным технологиям, программированию;
- профориентация школьников;
- подготовка к последующему профессиональному изучению программирования в высшей школе;
- реализация творческих идей обучающихся в области программирования в виде проектов высокого уровня сложности.

Задачи:

Образовательные:

- дать представление о значении информационных технологий в развитии общества и в изменении характера труда человека;
- познакомить с основными понятиями информатики непосредственно в процессе создания информационного продукта;
- обучить методам программирования на языке Java, применяемых в современной вычислительной технике, и работе в интегрированных средах разработки;
- сформировать навыки проектирования мобильных приложений, создания программ и их отладки на мобильных устройствах;
- формировать и развивать навыки публичного выступления.

Воспитательные:

- воспитать мотивацию учащихся к изобретательству, созданию собственных программных реализаций;

- привить стремление к получению качественного законченного результата в проектной деятельности;
- привить информационную культуру: ответственное отношение к информации с учетом правовых и этических аспектов её распространения, избирательного отношения к полученной информации;
- формировать правильное восприятие системы ценностей, принципов, правил информационного общества;
- формировать потребность в самостоятельном приобретении и применении знаний, потребность к постоянному саморазвитию;
- воспитывать социально-значимые качества личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность.

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;
- формировать творческий подход к поставленной задаче;
- развивать навыки инженерного мышления, умения работать как по предложенным инструкциям, так и находить свои собственные пути решения поставленных задач;
- развивать навыки эффективной деятельности в проекте;
- развивать стрессоустойчивость;
- развивать способности к самоанализу, самопознанию;
- формировать навыки рефлексивной деятельности.

Отличительные особенности программы

Программа предназначена для учащихся, проявляющих повышенный интерес к программированию.

Ключевым элементом обучения является проектная деятельность.

Проектная деятельность ориентирована на использование знаний, умений и навыков, полученных в ходе обучения, для постановки и решения практических задач, которые носят прикладной характер. Она позволяет учащимся участвовать в создании конкретного результата и научиться работать в условиях ограниченного времени, под руководством заказчика, презентовать проект, работать в команде, а также обрести навыки профессиональной коммуникации с контрагентами.

Категория обучающихся

Программа предназначена для детей, проявляющих интерес к программированию, стремящимся к саморазвитию, профессиональному самоопределению, имеющим начальные представления о языках программирования.

Возраст обучающихся: 14 — 17 лет.

Наполняемость группы: 12 человек.

Состав группы: разновозрастной.

Срок реализации программы: 1 год.

Структура программы:

Программа состоит из пяти модулей.

Форма реализации программы — очная с использованием электронного обучения.

Под электронным образованием понимается реализация образовательных программ с использованием информационно - образовательных ресурсов, информационно-коммуникационных технологий, технических средств, а также информационно-телекоммуникационных сетей, обеспечивающих передачу информационно-образовательных ресурсов и взаимодействие участников образовательного пространства.

Формы организации деятельности обучающихся

При изучении тем программа предусматривает использование фронтальной, индивидуальной и групповой формы учебной работы обучающихся.

Методы обучения

Основным методом обучения является метод проектов.

По способу организации занятий — словесные, наглядные, практические.

Типы занятий: теоретические, практические, комбинированные, контрольные.

Режим занятий

Четыре часа один раз в неделю или два часа по два раза в неделю.

Ожидаемые результаты

Основным результатом обучения является достижение высокой информационно-коммуникационной компетентности учащегося в области программирования мобильных приложений на языке Java.

В результате освоения программы обучающийся должен приобрести следующие знания, умения и навыки:

знать:

- ✓ правила работы с компьютером и технику безопасности;
- ✓ 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;
- ✓ знание техники ведения проектной деятельности и принципов тайм-менеджмента.

уметь:

- ✓ искать информацию с применением правил поиска в компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по

различным темам;

✓ следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;

✓ использовать приобретенные знания и умения в практической деятельности и повседневной жизни для:

– создания простейших моделей объектов и процессов в виде изображений и чертежей,

– создания информационных объектов, в том числе для оформления результатов учебной работы;

– организации индивидуального информационного пространства, создания личных коллекций информационных объектов;

– передачи информации по телекоммуникационным каналам в учебной и личной переписке, использования информационных ресурсов общества с соблюдением соответствующих правовых и этических норм;

✓ эффективно использовать интегрированную среду разработки;

✓ проектировать мобильные приложения, создавать программы и выполнять их отладку на мобильных устройствах;

✓ писать код программы на языке Java;

✓ формировать цели, ставить задачи для её достижения в ходе решения проблемных ситуаций;

✓ презентовать себя, свой продукт;

✓ мыслить творчески, придумывать и воплощать в жизнь свои идеи.

обладать навыками:

✓ исследовательской, проектной и социальной деятельности, строить логическое доказательство;

✓ использования, создания и преобразования различных символьных записей, схем и моделей для решения познавательных и учебных задач в различных предметных областях, исследовательской и проектной деятельности;

✓ проектирования, разработки, документирования и представления собственных проектов в составе команды;

✓ самообразования - периодической оценкой своих успехов и собственной работы самими обучающимися.

Способы определения результативности

Педагогическое наблюдение, педагогический анализ результатов решения задач с использованием автоматизированной системы контроля знаний, результаты участия в интеллектуальных конкурсах всероссийского уровня.

Виды контроля: входной, промежуточный, итоговый.

Формы подведения итогов реализации программы

По окончании обучения проводится итоговая аттестация в форме публичной защиты проектов. Документальной формой подтверждения итогов промежуточной аттестации является документ об образовании установленного Центром «Поиск» образца и сертификат от компании-партнера Samsung.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№	Наименование кейса, темы	Количество часов		
		Теория	Практика	Всего
Модуль 1. Основы программирования на языке Java.		6	12	18
1	Тема 1.1. Знакомство со средой программирования на языке Java. Типы данных и операции.	1	1	2
2	Тема 1.2. Логические выражения.	1	1	2
3	Тема 1.3. Условный оператор.	1	1	2
4	Тема 1.4. Циклы while, do while	1	1	2
5	Тема 1.5. Цикл for. Массивы.	1	1	2
6	Тема 1.6. Методы (процедуры и функции).	1	1	2
7	Тема 1.7. Многомерные массивы.		2	2
8	Промежуточный тест по модулю 1.		4	4
Модуль 2. Введение в объектно-ориентированное программирование.		10	16	26
9	Тема 2.1. Классы и объекты	2	2	4
10	Тема 2.2. Конструкторы и их перегрузка. Статические поля и методы.	2	2	4
11	Тема 2.3. Приемы тестирования и отладки на примерах со строками.	1	1	2
12	Тема 2.4. Знакомство с Android-разработкой.	2	2	4
13	Тема 2.5. Интерфейс Android-приложения.	1	3	4
14	Тема 2.6. Наследование и полиморфизм в Java.	2		2
15	Тема 2.7. Намерения.		2	2
16	Промежуточный тест по модулю 2.		4	4
Модуль 3. Основы программирования Android-приложений.		8	12	22
17	Тема 3.1. Объектно-ориентированное проектирование	2		2
18	Тема 3.2. Ввод, вывод и исключение.	1	1	2
19	Тема 3.3. Внутренние и анонимные классы.	1	1	2
20	Тема 3.4. Параллелизм и синхронизация, потоки.	2		2
21	Тема 3.5. Фрагменты в Android.	1	1	2
22	Тема 3.6. Двумерная графика в Android-приложениях.		2	4
23	Тема 3.7. Разработка игровых приложений SurfaceView.	1	3	4
24	Промежуточный тест по модулю 3.		4	4

	Модуль 4. Алгоритмы и структуры данных на языке Java.	14	16	30
25	Тема 4.1. Массивы. Класс ArrayList.	1	1	2
26	Тема 4.2. Связанные списки. Очереди, стеки, деки.	2	2	4
27	Тема 4.3. Списки в Android. Адаптеры.	1	1	2
28	Тема 4.4. СУБД. Реляционная модель.	2		2
29	Тема 4.5. СУБД SQLite. Основы языка SQL.	2	2	4
30	Тема 4.6. Рекурсия.	1	1	2
31	Тема 4.7. Деревья.	1	1	2
32	Тема 4.8. Алгоритмы сортировок.	2	2	4
33	Тема 4.9. Множества. Хеширование.	1	1	2
34	Тема 4.10. Ассоциативные массивы.	1	1	2
35	Промежуточный тест по модулю 4.		4	4
	Модуль 5. Основы разработки серверной части мобильных приложений.	18	30	48
36	Тема 5.1. IP – сети.	2	2	4
37	Тема 5.2. Веб-сервер, HTTP-запросы и ответы.	2	2	4
38	Тема 5.3. Клиент-серверная архитектура мобильных приложений.	2	2	4
39	Тема 5.4. Облачные платформы. REST-взаимодействие.	2	2	4
40	Тема 5.5. Серверные СУБД.	2	2	4
41	Промежуточный тест по модулю 5.		4	4
42	Работа над итоговым проектом.	4	8	12
43	Подготовка к защите проектов.		4	4
44	Защита итоговых проектов.		4	4
45	Рефлексия.	4		4
	Итого	56	86	144

СОДЕРЖАНИЕ КУРСА «МОБИЛЬНАЯ РАЗРАБОТКА».

Модуль 1. Основы программирования на языке Java.

Уровень предъявления материала обеспечивает учащимся знакомство с основами программирования через создание простых мобильных приложений на языке программирования Java.

Учащиеся должны знать:

- базовые и сложные конструкции языка программирования Java;
- основные методы сортировки массивов (метод простого выбора, метод простого обмена, метод простой вставки);
- алгоритмы обработки символьных переменных;
- понятие подпрограммы, функции пользователя;
- способы организации процедур (подпрограммы) и функций, принципы модульной технологии программирования;
- алгоритмы обработки массивов (поиск элемента с заданными свойствами, удаление элементов, вставка элементов, замена элементов, обмен местами групп элементов).

Учащиеся должны уметь:

- устанавливать необходимый инструментарий для разработки приложений;
- составлять программный код с использованием разных типов данных и операций над ними;
- запускать созданные программы на мобильном устройстве;
- применять условные конструкции if/else, switch/case;
- решать задачи с использованием разных видов циклов;
- составлять программы обработки массивов;
- использовать функции и процедуры;
- искать информацию в свободных источниках и структурировать ее;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного кейса:

- лекционная,

- групповая работа,
- групповые консультации,
- индивидуальная работа,
- тестирование.

Тема 1.1. Знакомство со средой программирования на языке Java. Типы данных и операции.

Теория. Понятия «алгоритм», «программа», «компилятор», «отладчик», «программирование», «язык программирования».

Практика. Установка и знакомство с интерфейсом сред разработки Eclipse, Android Studio, IntelliJ IDEA. Составление первой простой программы.

Форма подведения итогов: выполнение теста по теме.

Тема 1.2. Логические выражения.

Теория. Операторы сравнения. Приоритет операций. Логические выражения. Тип boolean. Тернарная операция.

Практика. Принадлежность точки закрашенной области. Составление логических выражений.

Форма подведения итогов: выполнение теста по теме.

Тема 1.3. Условный оператор.

Теория. Конструкция if/else. Условные выражения. Вложенность условных операторов.

Практика. Создание простой игры-погони с использованием переменных.

Форма подведения итогов: выполнение теста по теме.

Тема 1.4. Циклы while, do while

Теория. Конструкция while, do while. Паттерны использования циклов. Оператор break.

Практика. Пример решения задачи с помощью цикла. Android-практикум: рисование узоров на Canvas.

Форма подведения итогов: выполнение теста по теме.

Тема 1.5. Цикл for. Массивы.

Теория. Конструкция for. Оператор break с меткой. Оператор continue.

Одномерные массивы. Цикл for each.

Практика. Примеры программ обработки массивов. Создание простейшей анимации. Графики функций. Модель солнечной системы. Маятники.

Форма подведения итогов: выполнение теста по теме.

Тема 1.6. Методы (процедуры и функции).

Теория. Определение метода. Параметры метода. Возвращаемое значение. Области видимости переменной.

Практика. Применение функций.

Форма подведения итогов: выполнение теста по теме.

Тема 1.7. Многомерные массивы.

Теория. Создание многомерных массивов и обращение к их элементам. Двумерный массив. Трехмерный массив. «Неровные» массивы.

Практика. Клетчатое поле в Android.

Форма подведения итогов: промежуточное тестирование по модулю.

Модуль 2. Введение в объектно-ориентированное программирование.

Уровень предъявления материала обеспечивает учащимся возможность создавать программы для мобильных устройств на языке программирования Java.

Учащиеся должны знать:

- базовые и сложные конструкции языка программирования Java;
- способы организации процедур (подпрограммы) и функций, принципы модульной технологии программирования;
- определения и описания классов и объектов;
- понятия «инкапсуляция», «наследование», «полиморфизм»;
- конструкторы и способы их перегрузки;
- статические поля и методы.

Учащиеся должны уметь:

- запускать созданные программы на мобильном устройстве;
- тестировать и осуществлять отладку программ;

- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного кейса:

- лекционная,
- групповая работа,
- групповые консультации,
- индивидуальная работа,
- тестирование.

Тема 2.1. Классы и объекты.

Теория. Классы и объекты. Инкапсуляция. Наследование. Полиморфизм.

Описание класса.

Практика. Обзор классов-оболочек примитивных типов.

Форма подведения итогов: выполнение теста по теме.

Тема 2.2. Конструкторы и их перегрузка. Статические поля и методы.

Теория. Конструкторы. Перегрузка методов. Ключевое слово `this`.

Спецификаторы доступа. Статические компоненты класса.

Практика. Инициализация различных типов данных.

Форма подведения итогов: выполнение теста по теме.

Тема 2.3. Приемы тестирования и отладки на примерах со строками.

Теория. Строки. Отладочный вывод и логирование. Использование отладчика. Использование утверждений (assertions). Модульное тестирование.

Практика. Применение логирования.

Форма подведения итогов: выполнение теста по теме.

Тема 2.4. Знакомство с Android-разработкой.

Теория. Платформа Android. Структура проекта. Активности (Activity).

Практика. Создаем Android-проект. Запуск приложения.

Форма подведения итогов: выполнение теста по теме.

Тема 2.5. Интерфейс Android-приложения.

Теория. Язык разметки XML. XML-документ. Описание ресурсов Android с помощью XML. Строковые ресурсы.

Практика. Интерфейс пользователя. Разметка (layout). Компоненты (View).

Форма подведения итогов: выполнение теста по теме.

Тема 2.6. Наследование и полиморфизм в Java.

Теория. Понятие наследования. Защищенные члены класса. Ключевое слово super. Понятие полиморфизма. Абстрактные классы. Ключевое слово final. Понятие интерфейса.

Практика. Графическое описание структуры классов в UML. Иерархия наследования и преобразования типов.

Форма подведения итогов: выполнение теста по теме.

Тема 2.7. Намерения.

Теория. Контекст в Android. Намерения (Intent). Неявные намерения (Intenet).

Форма подведения итогов: промежуточное тестирование по теме модуля.

Модуль 3. Основы программирования Android-приложений.

В процессе работы по данному кейсу учащиеся овладеют навыками проектирования мобильных приложений, создания программ и отладки на мобильных устройствах.

Учащиеся должны знать:

- основные понятия объектно-ориентированного проектирования;
- назначение обработки исключений;
- внутренние и анонимные классы;
- типы сенсоров;
- определение параллелизма и синхронизации;
- основы графики в Android;
- общие способы создания потоков в Java.

Учащиеся должны уметь:

- генерировать идеи;
- обрабатывать исключения с помощью конструкции try-catch;
- работать с файлами, считывать и записывать в файл;

- управлять потоками;
- разрабатывать игровые приложения;
- разрабатывать 3D-игры с использованием фреймворка libGDX;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного кейса:

- лекционная,
- групповая (командная) работа,
- групповые консультации;
- защита проектов.

Тема 3.1. Объектно-ориентированное проектирование.

Теория. Операции с дробями. Шаблоны и принципы проектирования.

Практика. Текстовый квест. Электронный журнал.

Тема 3.2. Ввод, вывод и исключение.

Теория. Обработка исключений. Ключевые слова `throw` \ `throws`.

Обработка исключения с помощью конструкции `try-catch`. Ключевое слово `finally`. Основные методы класса `Exception`.

Практика. Работа с файлами. Чтение \ запись в файл.

Тема 3.3. Внутренние и анонимные классы.

Теория. Понятие внутреннего класса. Внутренние классы-члены.

Локальные внутренние классы. Анонимные классы.

Практика. Текстовый квест.

Тема 3.4. Параллелизм и синхронизация, потоки.

Теория. Общий способ создания потоков в Java. Реализация логики потока. Синхронизация потоков. Управление потоками. Блокировки.

Тема 3.5. Фрагменты в Android.

Теория. Фрагменты. Класс `Fragment` и его методы. Взаимодействия между фрагментами и активностями.

Практика. Создание фрагментов. Управление фрагментами.

Форма подведения итогов: выполнение теста по теме.

Тема 3.6. Двумерная графика в Android-приложениях.

Практика. Игра «Забавные птички»: игровое поле, создание класса Sprite для управления анимацией, добавление противника и контроль столкновений.

Форма подведения итогов: выполнение теста по теме.

Тема 3.7. Разработка игровых приложений SurfaceView.

Теория. Общие подходы для реализации игровых приложений. Понятие игрового движка и его использование при разработке игры. Класс SurfaceView.

Практика. Создание приложений с помощью SurfaceView.

Форма подведения итогов: промежуточное тестирование по теме модуля.

Модуль 4. Алгоритмы и структуры данных на языке Java.

В процессе работы по данному кейсу учащиеся овладеют навыками работы с массивами и списками, научатся работать с системой управления базами данных, познакомятся с основами криптографии и криптоанализа.

Учащиеся должны знать:

- структуры данных (массивы, списки, множества);
- алгоритмы поиска элементов по значению;
- понятия стеков, очереди, дека;
- понятие реляционной модели, классификацию СУБД;
- определение рекурсии (линейной, ветвящейся);
- определение и разновидности деревьев;
- различные виды шифров (подстановки, перестановки);
- методы шифрования (симметричный, на основе алгоритма DES, ассиметричный, алгоритм RSA)
- понятие криптоанализа.

Учащиеся должны уметь:

- генерировать идеи;
- работать с СУБД (SQLite на Android-устройстве);
- производить сортировку (пузырьком, вставками, быстрая);
- осуществлять синхронизацию ассоциативных массивов;

- использовать контент-провайдеры;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного кейса:

- лекционная,
- групповая (командная) работа,
- групповые консультации;
- защита проектов.

Тема 4.1. Массивы. Класс ArrayList.

Теория. Структуры данных. Сложность алгоритмов. Массив — базовая структура данных. Операции с массивами. Класс Arrays. Сравнение объектов. Компараторы. Расширяемый массив. Класс ArrayList.

Практика. Алгоритмы поиска элементов по значению.

Форма подведения итогов: выполнение теста по теме.

Тема 4.2. Связанные списки. Очереди, стеки, деки.

Теория. Связные списки. Класс LinkedList. Сравнение ArrayList и LinkedList. Коллекции. Интерфейс List. Стеки, очереди, деки.

Практика. Демонстрация работы очередей и стеков.

Форма подведения итогов: выполнение теста по теме.

Тема 4.3. Списки в Android. Адаптеры.

Теория. Список из ресурсов. ArrayAdapter. Собственная разметка.

Практика. Задание по Android-практикуму.

Тема 4.4. СУБД. Реляционная модель.

Теория. Реляционная модель. Реляционная БД из нескольких таблиц. Классификация СУБД.

Тема 4.5. СУБД SQLite. Основы языка SQL.

Теория. СУБД SQLite. Создание таблиц. Добавление записей в таблицу. Выборка данных. Изменение таблицы. Удаление записей. Агрегированные запросы.

Практика. Работа с базой данных SQLite на Android-устройстве.

Тема 4.6. Рекурсия.

Теория. Рекурсия в программировании и не только. Стек вызовов. Линейная рекурсия. Ветвящаяся рекурсия.

Практика. Разработка приложения, которое рисует рекурсивные фигуры на плоскости.

Форма подведения итогов: выполнение теста по теме.

Тема 4.7. Деревья.

Теория. Дерево. Разновидности деревьев. Понятие бинарного дерева. Понятие сбалансированного дерева. Библиотечный класс TreeSet.

Практика. Создание приложения с использованием деревьев.

Форма подведения итогов: выполнение теста по теме.

Тема 4.8. Алгоритмы сортировок.

Теория. Введение в сортировку данных. Сортировка пузырьком. Сортировка вставками. Быстрая сортировка.

Практика. Реализация сортировок в библиотечных классах Java.

Форма подведения итогов: выполнение теста по теме.

Тема 4.9. Множества. Хеширование.

Теория. Множества. Множество целых чисел от 0 до 100. Хеширование. Хеш-таблица.

Практика. Интерфейс Set. Классы HashSet и TreeSet.

Форма подведения итогов: выполнение теста по теме.

Тема 4.10. Ассоциативные массивы.

Теория. Ассоциативный массив как набор пар «ключ — значение». Интерфейс Map. Классы для Map. Контейнер HashMap. Контейнер TreeMap. Синхронизация ассоциативных массивов. Хранение данных в Android Preferences.

Практика. Разработка приложения с сохранением строк и целого числа.

Форма подведения итогов: выполнение теста по теме и промежуточного тестирования по теме модуля.

Модуль 5. Основы разработки серверной части мобильных приложений.

В процессе работы по данному кейсу учащиеся овладеют основными навыками клиент-серверной разработки приложений.

Учащиеся должны знать:

- основные понятия компьютерных сетей;
- принцип работы глобальной сети Интернет и локальных сетей;
- понятие эталонной сетевой модели OSI;
- принципы передачи данных – методы управления;
- структуру пакетов;
- принципы работы протокола TCP/IP и примеры прикладных сетевых протоколов;
- основы маршрутизации в сетях TCP/IP;
- определение IP адреса, его версии и формат;
- понятия доменных имен, URL-ссылок;
- структуру HTTP-запроса;
- понятие веб-сервера
- облачные технологии;
- клиент-серверные архитектуры.

Учащиеся должны уметь:

- применять популярные сетевые команды;
- устанавливать необходимое программное обеспечение для работы с сервером;
- писать HTTP-приложения под Android;
- осуществлять сериализацию с помощью JSON;
- использовать библиотеку Retrofit;
- реализовать серверную часть приложения, выполненную ранее на языке Java, при помощи скриптового языка программирования PHP;
- объективно оценивать результаты своей работы.

Формы занятий, используемые при изучении данного кейса:

- лекционная,
- групповая (командная) работа,

- групповые консультации;
- защита проектов.

Тема 5.1. IP – сети.

Теория. Интернет и протоколах TCP/IP. Адресация в IP-сетях. Версия интернет-протокола IPv4. Автоматизация процесса назначения IP-адресов. Доменные имена (DNS), URL-ссылки.

Практика. Сервисы работы с IP-адресами. Популярные сетевые команды.

Тема 5.2. Веб-сервер, HTTP-запросы и ответы.

Теория. HTTP-протокол. Структура HTTP-запроса. Ответы сервера. Веб-сервер.

Практика. Реализация сервера на PHP.

Форма подведения итогов: выполнение теста по теме.

Тема 5.3. Клиент-серверная архитектура мобильных приложений.

Теория. Архитектура клиент-сервер. Форматы JSON и XML. Сериализация. Библиотека Retrofit.

Практика. Отправка запросов из Android-приложений. Использование JSON и библиотеки Retrofit.

Тема 5.4. Облачные платформы. REST-взаимодействие.

Теория. Облачные технологии. Модели развертывания. Модели обслуживания. Платформа как услуга. REST-взаимодействие. REST-аутентификация и OAuth-авторизация.

Практика. Синхронные и асинхронные запросы. Возможности REST-взаимодействия на примере одного из API Яндекс.Предиктора.

Тема 5.5. Серверные СУБД.

Теория. Клиент-серверные архитектуры. Серверные СУБД. Настройка PostgreSQL и подключение к БД. Реализация back end части приложения на языке Java.

Практика. Реализация back end части приложения на языке PHP.

Форма подведения итогов: промежуточное тестирование по теме модуля.

**МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ
ПРОГРАММЫ «МОБИЛЬНАЯ РАЗРАБОТКА»**

Тема модуля	Форма занятий	Приёмы и методы организации образовательного процесса	Дидактический материал. Электронные источники	Техническое оснащение и расходный материал	Форма подведения итогов
Модуль 1. Основы программирования на языке Java	Комбинированная	Метод проектов. Объяснительно-иллюстративный.	https://myitschool.ru/opencourse/course/view.php?id=3&section=1 - "IT школа SAMSUNG" онлайн ресурс по мобильной разработке компании Samsung Electronics	<ul style="list-style-type: none"> ● Ноутбуки с мышкой и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7); Android Studio; Eclipse. ● Презентационное оборудование. 	Тестирование

<p>Модуль 2. Введение в объектно-ориентированное программирование</p>	<p>Комбинированная</p>	<p>Метод проектов. Объяснительно-иллюстративный.</p>	<p>https://myitschool.ru/opencourse/course/view.php?id=3&section=1 - "IT школа SAMSUNG" онлайн ресурс по мобильной разработке компании Samsung Electronics</p>	<ul style="list-style-type: none"> ● Ноутбуки с мышкой и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7); Android Studio; Eclipse. <ul style="list-style-type: none"> ● Презентационное оборудование. ● Мобильные устройства на базе ОС Android. 	<p>Тестирование</p>
<p>Модуль 3. Основы программирования Android-приложений.</p>	<p>Комбинированная</p>	<p>Метод проектов. Объяснительно-иллюстративный.</p>	<p>https://myitschool.ru/opencourse/course/view.php?id=3&section=1 - "IT школа SAMSUNG" онлайн ресурс по мобильной разработке компании Samsung Electronics</p>	<ul style="list-style-type: none"> ● Ноутбуки с мышкой и доступом к сети Интернет, на которых установлено 	<p>Тестирование</p>

				<p>следующие программное обеспечение: операционная система Windows (версия не ниже 7); Android Studio; Eclipse.</p> <ul style="list-style-type: none"> • Презентационное оборудование. • Мобильные устройства на базе ОС Android. 	
Модуль 4. Алгоритмы и структуры данных на языке Java.	Комбинированная	Метод проектов. Объяснительно-иллюстративный.	https://myitschool.ru/opencourse/course/view.php?id=3&section=1 - "IT школа SAMSUNG" онлайн ресурс по мобильной разработке компании Samsung Electronics	<ul style="list-style-type: none"> • Ноутбуки с мышкой и доступом к сети Интернет, на которых установлено следующее программное обеспечение: операционная система Windows 	Тестирование

				<p>(версия не ниже 7); Android Studio; Eclipse.</p> <ul style="list-style-type: none"> • Презентационное оборудование. • Мобильные устройства на базе ОС Android. 	
Модуль 5. Основы разработки серверной части мобильных приложений.	Комбинированная	Метод проектов. Объяснительно-иллюстративный.	<p>https://myitschool.ru/opencourse/course/view.php?id=3&section=1 - "IT школа SAMSUNG" онлайн ресурс по мобильной разработке компании Samsung Electronics</p> <p>http://www.intuit.ru/studies/courses/4455/712/lecture/21291?page=2 - Протокол HTTP/HTTPS [Электронный ресурс]/ НОУ Интуит</p>	<ul style="list-style-type: none"> • Ноутбуки с мышкой и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7); Android Studio; Eclipse. • Презентационное оборудование. 	Тестирование. Защита проектов

				<ul style="list-style-type: none">● Мобильные устройства на базе ОС Android.	
--	--	--	--	--	--

СПИСОК ЭЛЕКТРОННЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ

использованных при написании программы:

1. Войков В. «АЙТИ Квантум тулкит». - Базовая серия «Методический инструментарий тьютора», 2017.
2. <https://myitschool.ru/opencourse/course/view.php?id=3§ion=1> - "IT школа SAMSUNG" онлайн ресурс по мобильной разработке компании Samsung Electronics

рекомендованных обучающимся:

1. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 – Робин Никсон;
2. PHP. Быстрый старт – Каллум Хопкинс;
3. <https://stepik.org/> – ресурс для самообразования, образовательная платформа и конструктор онлайн-курсов;
4. <http://www.intuit.ru/studies/courses/4455/712/lecture/21291?page=2> - Протокол HTTP/HTTPS [Электронный ресурс]/ НОУ Интуит