

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

Принята на заседании
научно-методического совета
ГАНОУ СО «Дворец молодёжи»
Протокол № 5 от 25.06.2026 г.

Утверждена директором
ГАНОУ СО «Дворец молодёжи»
А.Н. Слизько
Приказ № 860-д от 25.06.2026 г.

**Дополнительная общеобразовательная общеразвивающая программа
технической направленности**

«Мобильная разработка»

Стартовый, базовый, продвинутый уровни

Возраст обучающихся: 13 – 17 лет

Объем общеразвивающей программы: 432 часа

Срок реализации: 3 года

СОГЛАСОВАНО:

Начальник центра цифрового образования
«IT-куб»

А.А. Лаптева

Начальник центра цифрового образования
детей «IT-куб. Арамилъ»

В.А. Сырникова

Начальник центра цифрового образования
детей «IT-куб. Верхняя Пышма»

Е.Г. Евставьева

Начальник центра цифрового образования
детей «IT-куб. Солнечный»

О.А. Чуенко

Авторы-составители:

Самолов А.А.,

Габдуллина С.И.,

Сафин К.Р.,

Шмелёв А.А.,

педагоги дополнительного
образования,

Фефелова М.В.,

методист.

I. Комплекс основных характеристик программы

1. Пояснительная записка

На сегодняшний день разработка программного обеспечения является наиболее востребованным направлением в любых сферах применения. Кроме того, большое развитие мобильных платформ даёт более широкий выбор направлений разработки.

В современном мире существует несколько основных платформ и языков программирования, с помощью которых создают мобильные приложения. При выборе языка программирования принято ориентироваться на задачи, которые данное приложение будет выполнять. Среди наиболее популярных и часто используемых языков программирования выделяют следующие: Java, Kotlin, C#, Swift и другие.

Программа «Мобильная разработка» предназначена для изучения основных этапов разработки мобильных приложений и является унифицированной, что позволяет использовать при обучении разные языки программирования в зависимости от знаний и навыков преподавателя. В результате освоения программы обучающиеся приобретут теоретические знания и практические навыки программирования, а также смогут освоить основные принципы дизайна пользовательского интерфейса, смогут создавать прототипы и разрабатывать базовые приложения.

Направленность общеразвивающей программы

Программа «Мобильная разработка» имеет техническую направленность, ориентирована на развитие навыков программирования и проектирования программ для мобильных устройств.

Программа разработана с учётом требований следующих нормативных правовых актов и государственных программных документов:

- Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- Федеральный закон Российской Федерации от 14.07.2022 № 295-ФЗ «О внесении изменений в Федеральный закон «Об образовании в Российской Федерации»;

- Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребёнка в Российской Федерации» (в редакции от 20.02.2026 г.);
- Концепция развития дополнительного образования детей до 2030 года, утвержденная распоряжением Правительства Российской Федерации от 31 марта 2022 г. № 678-р;
- Указ Президента Российской Федерации от 21.07.2020 № 474 «О национальных целях развития Российской Федерации на период до 2030 года»;
- Указ Президента Российской Федерации от 09.11.2022 № 809 «Об утверждении Основ государственной политики по сохранению и укреплению традиционных российских духовно-нравственных ценностей»;
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 г. № 28 «Об утверждении СанПиН 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Постановление Правительства Российской Федерации от 11.10.2023 № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- Приказ Министерства труда и социальной защиты Российской Федерации от 22.09.2021 г. № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;
- Приказ Министерства просвещения Российской Федерации от 27.07.2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Приказ Министерства просвещения Российской Федерации от 03.09.2019 г. № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;

– Приказ Министерства науки и высшего образования РФ и Министерства просвещения РФ от 05 августа 2020 г. № 882/391 «Об утверждении Порядка организации и осуществления образовательной деятельности при сетевой форме реализации образовательных программ»;

– Письмо Минобрнауки России № 09-3242 от 18.11.2015 «О направлении информации» (вместе с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»);

– Письмо Минобрнауки России от 28.08.2015 № АК-2563/05 «О методических рекомендациях (вместе с «Методическими рекомендациями по организации образовательной деятельности с использованием сетевых форм реализации образовательных программ»);

– Письмо Министерства просвещения Российской Федерации от 07.05.2020 № ВБ-976/04 «Рекомендации по реализации внеурочной деятельности, программы воспитания и социализации и дополнительных общеобразовательных программ с применением дистанционных образовательных технологий»;

– Приказ Министерства образования и молодёжной политики Свердловской области от 30.03.2018 № 162-Д «Об утверждении Концепции развития образования на территории Свердловской области на период до 2035 года»;

– Приказ Министерства образования и молодёжной политики Свердловской области от 29.06.2023 № 785-Д «Об утверждении Требований к условиям и порядку оказания государственной услуги в социальной сфере «Реализация дополнительных образовательных программ в соответствии с социальным сертификатом»;

– Положение о дополнительных общеобразовательных общеразвивающих программах ГАНОУ СО «Дворец молодежи», утвержденное приказом от 14.05.2020 № 269-д;

– Положение о сетевой форме реализации образовательных программ в ГАНОУ СО «Дворец молодёжи», утвержденное приказом от 08.11.2021 г. № 947-д.

Актуальность общеразвивающей программы

В настоящее время широкое распространение получили мобильные устройства: планшеты, смартфоны, и др. Количество мобильных устройств значительно превысило количество настольных компьютеров и ноутбуков, их возможности уже приближаются к возможностям современных компьютеров по быстродействию и объему памяти. Значительное число новых информационных систем и программных продуктов разрабатывается с учётом возможности работы на мобильных устройствах. Необходимо готовить новое поколение специалистов, владеющих навыками работы в данной области. Программа демонстрирует основные направления в мобильной разработке, а также позволяет осветить углубленные моменты с практической стороны.

Компетенции, которые освоят обучающиеся в рамках программы, сформируют начальные знания и навыки в сфере разработки мобильных приложений, и станут преимуществом при выходе на рынок труда в будущем.

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

Отличительная особенность общеразвивающей программы

Отличительная особенность дополнительной общеобразовательной общеразвивающей программы «Мобильная разработка» заключается в ее комплексном подходе, который сочетает теоретические знания и практические навыки. Обучение направлено на приобретение практических навыков разработки мобильных приложений. Обучающиеся получают возможность создавать реальные проекты, решать конкретные задачи

и разрабатывать приложения для мобильных устройств, также познакомятся с последними тенденциями и технологиями в области мобильной разработки.

Также в результате изучения парадигмы объектно-ориентированного подхода к программированию происходит формирование базовых знаний и умений для работы с большинством популярных языков и необходимых при освоении других направлений информационных технологий.

Помимо технических навыков, обучающие развивают такие качества, как умение работать в команде, коммуникативные способности и создание проектов. Участие в реальных проектах, позволяет применять полученные знания на практике и развивать навыки работы в команде. Эти аспекты делают программу по мобильной разработке не только образовательной, но и практической, готовя обучающихся к реальным вызовам в индустрии мобильной разработки.

Программа подходит для обучения на разных языках программирования, выбор языка будет зависеть от знаний и навыков преподавателя.

В дальнейшем обучающиеся могут продолжить обучение по другим направлениям в Центрах цифрового образования детей «IT-куб».

Адресат общеразвивающей программы

Дополнительная общеразвивающая программа «Мобильная разработка» предназначена для обучающихся в возрасте 13 – 17 лет. Количество обучающихся в группе 10 – 14 человек. Состав групп постоянный.

Место проведения занятий:

- ЦЦО «IT-куб», г. Екатеринбург, ул. Красных Командиров, 11а;
- ЦЦОД «IT-куб». Солнечный», г. Екатеринбург, ул. Чемпионов, 11;
- ЦЦОД «IT-куб». Арамиль», г. Арамиль, ул. Щорса, 55;
- ЦЦОД «IT-куб. Верхняя Пышма», г. Верхняя Пышма, Успенский проспект, 2г.

Возрастные особенности группы

Содержание программы учитывает возрастные и психологические особенности детей в возрасте 13 – 17 лет, которые определяют выбор форм проведения занятий с обучающимися. Подростки данной возрастной группы характеризуются такими процессами, как изменение структуры личности

и бурного физического развития. Происходят качественные изменения и в познавательной деятельности, и в личности, и в межличностных отношениях. У каждого эти изменения происходят в разное время. В этом возрасте начинается переход от мышления, основанного на оперировании конкретными представлениями, к теоретическому мышлению, от непосредственной памяти к логической. В 13 лет ведущий тип деятельности – референтно значимый, к нему относятся: проектная деятельность (встреча замысла и результата как авторское действие подростка), проявление себя в общественно значимых ролях (выход в настоящую взрослую действительность). В 15 – 17 лет ведущей деятельностью является – учебно-профессиональная деятельность.

13 – 14 лет – подростковый период. Характерная особенность – личное самосознание, сознательное проявление индивидуальности. Ведущая потребность – самоутверждение. В подростковый период стабилизируются интересы. Основное новообразование – становление взрослости как стремление к жизни в обществе взрослых. К основным ориентирам взросления относятся:

К основным ориентирам взросления относятся:

- социально-моральные – наличие собственных взглядов, оценок, стремление их отстаивать;
- интеллектуально-деятельностные – освоение элементов самообразования, желание разобраться в интересующих подростка областях;
- культурологические – потребность отразить взрослость во внешнем облике, манерах поведения.

15 – 17 лет – юношеский возраст. Завершение физического и психического созревания. Социальная готовность к общественно полезному производительному труду и гражданской ответственности. В отличие от подросткового возраста, где проявление индивидуальности осуществляется благодаря самоидентификации – «кто я», в юношеском возрасте индивидуальность выражается через самопроявление – «как я влияю». Основная задача педагога дополнительного образования в работе с детьми в возрасте 15 – 17 лет сводится к решению противоречия между готовностью

их к полноценной социальной жизни и недопущением отставания от жизни содержания и организации их образовательной деятельности.

Также следует отметить, что подростки в возрасте 15 – 17 лет характеризуются такими психическими процессами, как стремление углублённо понять себя, разобраться в своих чувствах, настроениях, мнениях, отношениях. Это порождает у подростка стремление к самоутверждению, самовыражению (проявления себя в тех качествах, которые он считает наиболее ценными)

и самовоспитанию. Эти процессы позволяют положить начало созданию начального профессионального самоопределения обучающихся.

Таким образом, возрастная периодизация определяет:

- возрастную особенность разработки общеобразовательных программ дополнительного образования детей;
- основные нормы условий полноты психофизиологического развития детей;
- базовые положения педагогической деятельности при реализации программы.

Режим занятий, периодичность и продолжительность занятий

Общее количество часов в неделю – 4 часа, продолжительность одного академического часа – 45 минут. Перерыв между учебными занятиями – 10 минут. Занятия проводятся 2 раза в неделю по 2 академических часа.

Срок освоения общеразвивающей программы

Срок освоения общеразвивающей программы определяется содержанием программы и составляет 3 года.

Формы обучения

Очная, а также возможна реализация программы очно с применением электронного обучения и дистанционных образовательных технологий (Закон № 273 – ФЗ, гл. 2, ст. 17, п. 2.).

Возможна реализация дополнительной общеобразовательной общеразвивающей программы в сетевой форме. ГАНОУ СО «Дворец молодёжи»

является базовой организацией, организация-участник определяется на основании заключенного договора о сетевой форме реализации программ.

Объём общеразвивающей программы

Общее количество учебных часов, запланированных на весь период обучения, необходимых для освоения программы 432 часа (144 часов на каждый модуль).

Уровневость общеразвивающей программы

По уровню освоения программа общеразвивающая, разноуровневая (стартовый, базовый, продвинутый уровни). Содержание программы имеет модульную структуру и организовано по принципу дифференциации по уровням сложности. Программа состоит из 3 модулей. Первый модуль является стартовым, второй модуль имеет базовый уровень сложности, третий модуль продвинутого уровня.

Первый модуль (стартовый уровень) рассчитан для обучающихся, проявляющих интерес к аналитической и исследовательской деятельности, информационным технологиям, приобретению навыков программирования. Зачисление на первый модуль производится без предварительного отбора (свободный набор). К концу первого года обучения обучающиеся приобретут навыки поиска, анализа и использования информации, получают навыки программирования в среде разработки и осваивают основы мобильной разработки. По окончании обучения на стартовом уровне проводится проектная работа, где обучающийся показывает свой навык в разработке мобильных приложений, работы с базами данных и веб-интерфейсами.

Второй модуль (базовый уровень) рассчитан на обучающихся, проявляющих интерес к информационным технологиям, понимающим концепции объектно-ориентированного программирования, умеющим писать на одном из языков программирования: Java, Kotlin, Swift, C#.

Зачисление обучающихся на базовый или продвинутый уровни обучения производится по результатам успешной сдачи итоговой аттестации (защиты итогового проекта) за каждый модуль обучения. Зачисление обучающихся,

ранее не занимавшихся по данной программе, происходит по результатам входного контроля (тестирования).

Каждый модуль является независимым курсом и может быть реализован отдельно от других. Ребёнок может быть принят на любой модуль обучения, соответствующий его возрасту, при наличии соответствующих знаний, а также вакантных мест в учебной группе. Однако для формирования стабильных знаний, умений и навыков, достижения высокого образовательного результата рекомендуется начинать обучение с первого модуля.

2. Цель и задачи общеразвивающей программы

Цель программы: создание условий для развития творческих и технических способностей обучающихся посредством изучения основ мобильной разработки.

Задачи программы:

Обучающие:

- познакомить с историей развития информационных технологий в России;
- способствовать формированию представления о современных и популярных платформах программирования мобильных приложений;
- обучить основам языка программирования для создания мобильных приложений;
- обучить объектно-ориентированному подходу в проектировании и разработке программного обеспечения;
- дать представление об основных компонентах архитектуры приложения для мобильных устройств;
- способствовать приобретению навыков программирования технических устройств;
- обучить современным нотациям и шаблонам программирования.

Развивающие:

- способствовать развитию умения искать, извлекать и отбирать нужную информацию из открытых источников;
- способствовать развитию умения самостоятельно ставить задачи, излагать мысли в четкой логической последовательности и отстаивать свою точку зрения;
- создавать условия для развития навыка прогнозирования результата и его достижения, при необходимости вносить коррективы в первоначальный замысел;
- способствовать формированию и развитию навыка исследовательской и проектной деятельности;

– познакомить с правилами индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Воспитательные:

– содействовать формированию культуры командной работы и уважительного взаимодействия;

– способствовать воспитанию упорства в достижении результата;

– способствовать формированию организованности и ответственного отношения к труду;

– способствовать воспитанию ценностного отношения к своему здоровью;

– способствовать воспитанию российской гражданской идентичности: патриотизма, любви и уважения к Отечеству, чувства гордости за свою Родину.

2.1. Цель и задачи Модуля 1

(Стартовый уровень)

Цель модуля: освоение обучающимися основ программирования посредством разработки приложений для мобильных устройств.

Задачи программы:

Обучающие:

– изучить основы работы персонального компьютера (ПК) и его устройств, включая аппаратное и программное обеспечение;

– способствовать приобретению навыков использования ПК и интегрированной среды разработки для программирования устройства;

– сформировать знания о современных и популярных платформах разработки мобильных приложений;

– обучить работе с запросами, сервером и базами данных;

– способствовать приобретению навыков чтения кода программы, поиска ошибок в нём;

– обучить объектно-ориентированному подходу в проектировании и разработке программного обеспечения;

– познакомить с основными элементами интерфейса мобильных приложений.

Развивающие:

– способствовать развитию навыков поиска, извлечения и отбора необходимой информации из открытых источников;

– сформировать представление об основах проектной деятельности, этапах создания проекта;

– познакомить с правилами индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Воспитательные:

– сформировать навыки коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной и проектной деятельности;

– сформировать целеустремлённость, организованность, ответственное отношение к труду и уважительное отношение к окружающим;

– способствовать воспитанию уважительного, позитивного отношения к окружающим, их мнению и деятельности.

2.2 Цель и задачи Модуля 2

(Базовый уровень)

Цель модуля: углубленное изучение и овладение практическими навыками разработки мобильных приложений с реализацией сетевого взаимодействия по клиент-серверной модели.

Задачи программы:

Обучающие:

– расширить знания о современных и популярных платформах;

– обучить созданию архитектуры клиент-серверного взаимодействия с использованием программного интерфейса приложений;

– сформировать знания о системах контроля версий (git);

– обучить работе с запросами, сервером и базами данных;

- освоить виды алгоритмов, алгоритмических конструкций и приемов;
- обучить языку структурированных запросов и языку разметки XML.

Развивающие:

- способствовать формированию алгоритмического мышления;
- способствовать формированию и развитию навыка исследовательской и проектной деятельности;
- способствовать развитию умения самостоятельно ставить и формулировать новые задачи, развивать мотивы своей познавательной деятельности.

Воспитательные:

- способствовать воспитанию упорства в достижении результата;
- содействовать формированию культуры командной работы и уважительного взаимодействия;
- способствовать приобретению практического опыта участия в социально значимой деятельности, повысить самооценку за счёт успешно завершённых проектов.

2.3 Цель и задачи Модуля 3

(Продвинутый уровень)

Цель модуля: углублённое освоение принципов проектирования архитектуры приложений с использованием актуальных технологических платформ и методологических подходов.

Задачи программы:

Обучающие:

- углубить знания о современных архитектурных паттернах (Clean Architecture, MVI) и принципах проектирования программного обеспечения;
- сформировать навыки использования библиотек внедрения зависимостей (Hilt, Coin) для создания масштабируемых и тестируемых приложений;

- обучить продвинутым методам работы с сетью (загрузка файлов, кэширование, обработка ошибок, повторные попытки);
- познакомить с инструментами профилирования и оптимизации производительности мобильных приложений;
- сформировать навыки командной работы с использованием систем контроля версий в рамках реального проекта;
- обучить методам тестирования (модульное, интеграционное, UI-тестирование) и отладки приложений.

Развивающие:

- способствовать развитию умения искать, извлекать и отбирать нужную информацию из открытых источников;
- способствовать формированию навыка самостоятельно формулировать цели и задачи проекта, доводить его до полной реализации;
- создавать условия для развития навыка прогнозирования результата и его достижения, при необходимости вносить коррективы в первоначальный замысел.

Воспитательные:

- способствовать воспитанию упорства в достижении результата;
- способствовать формированию организованности и ответственного отношения к труду;
- содействовать формированию культуры командной работы и уважительного взаимодействия.

3. Содержание общеразвивающей программы

3.1. Учебный (тематический) план

Модуль 1 (стартовый уровень)

Таблица 1

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
Раздел 1. Введение в разработку и основы программирования		40	18	22	
1.1	История развития информационных технологий в России. Лекция на тему «Что значит быть честным». Инструктаж по ТБ. Применение мобильной разработки. Знакомство со средой программирования	2	2	0	Опрос, входное тестирование
1.2	Переменные и типы данных. Ввод и вывод информации	4	2	2	Опрос, практическая работа
1.3	Условные операторы. Логические выражения	4	2	2	Опрос, практическая работа, устный опрос
1.4	Структуры данных. Циклы	4	2	2	Опрос, практическая работа
1.5	Функции: объявление, параметры, возвращаемые значения. Лямбда выражения	6	2	4	Опрос, практическая работа
1.6	Классы и объекты. Введение в принципы ООП	12	5	7	Опрос, практическая работа
1.7	Enum классы и модели данных	4	2	2	Опрос, практическая работа
1.8	Дополнительные возможности ООП	2	1	1	Опрос, практическая работа
1.9	Решение кейс-задания	2	0	2	Практическая работа
Раздел 2. Основы мобильной разработки		42	14	28	
2.1	Подходы к построению интерфейса, базовые компоненты	2	1	1	Опрос, практическая работа
2.2	Базовые визуальные компоненты. Контейнеры	6	2	4	Опрос, практическая работа
2.3	Кнопки и обработка нажатий. Введение в состояния	6	2	4	Опрос, практическая работа
2.4	Стилизация и модификаторы внешнего вида	4	1	3	Опрос, практическая работа
2.5	Использование готовых компонентов интерфейса (карточки, поля ввода, диалоги)	4	2	2	Опрос, практическая работа
2.6	Углубление в состояния	4	1	3	Опрос, практическая работа

2.7	Работа со списками. Отображение динамических данных	4	1	3	Опрос, практическая работа.
2.8	Навигация между экранами. Передача данных	6	2	4	Опрос, практическая работа
2.9	Промежуточный контроль. Введение в проектную деятельность	2	1	1	Промежуточный контроль
2.9	Создание мини-проекта	4	1	3	Практическая работа
Раздел 3. Архитектурные компоненты. Хранение данных		38	14	24	
3.1	Архитектура пользовательского интерфейса	2	2	0	Опрос
3.2	Разделение логики и интерфейса. Понятие модели представления	4	1	3	Опрос, практическая работа
3.3	Введение в асинхронность	6	2	4	Опрос, практическая работа
3.4	Сохранение простых настроек приложения	4	1	3	Опрос, практическая работа
3.5	Локальные базы данных: создание таблиц, запись и чтение данных	8	3	5	Опрос, практическая работа
3.6	Работа с медиа	4	1	3	Опрос, практическая работа
3.7	Подготовка концепта проекта	4	2	2	Опрос, практическая работа
3.8	Разбор библиотек для работы над проектами	6	2	4	Опрос, практическая работа
Раздел 4. Проектная деятельность		24	2	22	
4.1	Дизайн приложений. Дизайн-система	4	2	2	Опрос. Практическая работа
4.2	Итоговая аттестация	2	0	2	Итоговое тестирование
4.3	Работа над проектами	16	0	16	Практическая работа
4.4	Подготовка презентаций проектов. Защита проекта	2	0	2	Защита проектов
Итого:		144	48	96	

Содержание учебного (тематического) плана

Раздел 1. Введение в разработку и основы программирования

Тема 1.1. История развития информационных технологий в России.

Лекция на тему «Что значит быть честным». Инструктаж по ТБ. Применение мобильной разработки. Знакомство со средой программирования

Теория: лекция этапы становления информационных технологий в России и мире. Вклад Российской Федерации в сферу информационных технологий. Отечественные разработки. Антикоррупционное просвещение: беседа «Что значит быть честным». Проведение инструктажа по технике безопасности при работе за компьютером. Разработка мобильных приложений: что такое и с чего начать. Сферы применения мобильных разработок.

Практика: выполнение входного тестирования. Знакомство со средой разработки.

Тема 1.2. Переменные и типы данных. Ввод и вывод информации

Теория: понятие переменной. Примитивные типы данных: Операции присваивания, арифметические операторы. Ввод данных с клавиатуры (в консоли) и вывод на экран.

Практика: написание простых программ для вычисления арифметических выражений, обмена значениями переменных. Вывод результатов в консоль и на экран устройства.

Тема 1.3. Условные операторы. Логические выражения

Теория: логический тип Boolean. Операции отношения, логические операторы. Условные конструкции и ветвления. Понятие блока кода, область видимости переменных. Тернарный оператор.

Практика: решение задач на проверку условий (чётность числа, сравнение, определение времени года, простой калькулятор с выбором действия).

Тема 1.4. Структуры данных. Циклы

Теория: структуры хранения данных, работа с данными. Итерационные конструкции. Безусловные операторы перехода break с меткой, continue.

Практика: работа со структурами данных, вывод данных. Разбор примеров нахождения максимума и минимума, поиска на числовых массивах. Решение задач.

Тема 1.5. Функции: объявление, параметры, возвращаемые значения.
Лямбда-выражения

Теория: понятие функции, цели использования. Объявление функции, параметры, возвращаемое значение. Локальные и глобальные переменные. Модификаторы доступа. Функции высшего порядка, лямбда-выражения как способ передачи поведения.

Практика: написание функций для повторяющихся операций (например, вычисление площади фигур, приветствие пользователя). Использование лямбда-выражений для обработки коллекций (фильтрация, преобразование).

Тема 1.6. Классы и объекты. Введение в принципы ООП

Теория: объектно-ориентированный подход: класс как шаблон, объект как экземпляр класса. основные парадигмы ООП: инкапсуляция, наследование, полиморфизм. Понятие.

Практика: создание класса. Создание нескольких объектов, вывод информации о них. Написание программ с применением скрытия полей, предоставление методов доступа. Создание класса-наследника.

Тема 1.7. Енит классы и модели данных

Теория: перечисления для ограниченного набора значений. Дата классы.

Практика: создание для ролей пользователя. Создание data class для модели товара или заметки. Использование в простых приложениях.

Тема 1.8. Дополнительные возможности ООП

Теория: абстрактные классы и интерфейсы. Отличие интерфейса от абстрактного класса. Множественная реализация интерфейсов. Переопределение методов.

Практика: полиморфизм. Объекты компаньоны.

Тема 1.9. Решение кейс-задания

Практика: решение кейс задания.

Раздел 2. Основы мобильной разработки

Тема 2.1. Подходы к построению интерфейса, базовые компоненты

Теория: декларативный подход к построению интерфейса. Основные компоненты создания интерфейса.

Практика: создание экранов из базовых компонентов.

Тема 2.2. Базовые визуальные компоненты. Контейнеры

Теория: контейнеры для размещения элементов. Их свойства. Компоновка интерфейса.

Практика: компоновка базовых компонентов в контейнерах. Создание простого профиля пользователя.

Тема 2.3. Кнопки и обработка нажатий. Введение в состояния

Теория: обработка нажатий, передача состояний объектов. Связь состояний с пользовательским интерфейсом.

Практика: реализация счётчика нажатий с отображением текущего значения. Формирования простого списка элементов.

Тема 2.4. Стилизация и модификаторы внешнего вида

Теория: модификаторы для настройки внешнего вида. Стиливые файлы.

Практика: оформление интерфейса с применением стилей и модификаторов.

Тема 2.5. Использование готовых компонентов интерфейса (карточки, поля ввода, диалоги)

Теория: изучение готовых компонентов. Стили дизайн-системы. Их параметры и настройка.

Практика: создание простого приложения с подтверждением ввода и однотипных карт. Например: добавление товаров в магазин приложения или контактов.

Тема 2.6. Углубление в состояния

Теория: однонаправленный поток данных. Изучение различных состояний и их предназначения

Практика: реализация простых настроек приложения.

Тема 2.7. Работа со списками. Отображение динамических данных

Теория: эффективная переработка элементов. Ленивая загрузка.

Практика: применение ленивой загрузки. Возможность добавить новую задачу (через диалог или отдельный экран).

Тема 2.8. Навигация между экранами. Передача данных

Теория: знакомство с библиотекой навигации. Понятия стека и путей.

Практика: создание нескольких экранов с навигацией между ними. Передача данных между экранами.

Тема 2.9. Промежуточный контроль. Введение в проектную деятельность

Теория: понятие проектной деятельности, этапы работы над проектом.

Практика: выполнение промежуточного тестирования. Обсуждение возможных тем для итогового проекта.

Тема 2.10. Создание мини-проекта

Теория: понятие проектной деятельности, этапы работы над проектом.

Практика: разработка небольшого приложения с применением изученного. Например, приложение с заметками или список покупок. Работа в парах или индивидуально. Защита мини-проекта.

Раздел 3. Архитектурные компоненты. Хранение данных

Тема 3.1. Архитектура пользовательского интерфейса

Теория: принципы построения архитектуры мобильных приложений. Многослойная архитектура. Взаимодействие слоёв.

Практика: анализ структуры существующего мини-проекта, выделение слоёв.

Тема 3.2. Разделение логики и интерфейса. Понятие модели представления

Теория: ViewModel как компонент, отвечающий за хранение состояния и обработку событий. Жизненный цикл и правила работы.

Практика: создание проекта с использованием модели представления.

Тема 3.3. Введение в асинхронность

Теория: потоки выполнения, проблема долгих операций (сеть, база данных). Асинхронные функции, корутины.

Практика: имитация долгой загрузки (delay) с отображением прогресс-бара. Запуск асинхронного кода в модели представления.

Тема 3.4. Сохранение простых настроек приложения

Теория: DataStore. Сохранение простых ключей и значений.

Практика: сохранение темы оформления (светлая/тёмная) или счётчика запусков приложения. Чтение и запись настроек через DataStore.

Тема 3.5. Локальные базы данных: создание таблиц, запись и чтение данных

Теория: введение в SQL и реляционные базы данных. Библиотеки для работы с БД между таблицами

Практика: создание базы данных для хранения задач. Реализация репозитория для работы с БД. Интеграция с моделью представления.

Тема 3.6. Работа с медиа

Теория: загрузка изображений из интернета, кэширование. Работа с локальными медиафайлами: выбор из галереи, съёмка фото с использованием намерений.

Практика: добавление в приложение возможности выбрать фото из галереи и отобразить его. Загрузка изображения по URL.

Тема 3.7. Подготовка концепта проекта

Теория: выбор идеи для итогового проекта. Составление технического задания. Определение функциональных требований и минимального набора функций.

Практика: каждый учащийся или команда формулирует идею приложения, описывает основные экраны, пользовательские сценарии.

Тема 3.8. Разбор библиотек для работы над проектами

Теория: обзор популярных библиотек и инструментов, необходимых для реализации проектов. Углубленная работа с базами данных, базовая работа с REST API, обзорное внедрение зависимостей. Выбор подходящих библиотек.

Практика: разбор наиболее распространенных для проектов библиотек. Составление списков необходимых библиотек для проектов. Подключение выбранных библиотек в проект.

Раздел 4. Проектная деятельность

Тема 4.1. Дизайн приложений. Дизайн-система

Теория: основы дизайна мобильных приложений. Понятия «пользовательский опыт» и «пользовательский интерфейс» дизайна. Основные компоненты и элементы дизайн-системы. Цветовая палитра, типографика, анимация и тени. Документация по Дизайн-системе.

Практика: проектирование экранов с учетом принципов дизайн-системы. Применение стилей и цветовых схем. Импорт готовых ресурсов из библиотек дизайн-системы.

Тема 4.2. Итоговая аттестация

Практика: выполнение итогового тестирования, охватывающего все темы изучения первого модуля.

Тема 4.3. Работа над проектами

Практика: техническая проработка проектов.

Тема 4.4. Подготовка презентаций проектов. Защита проекта

Практика: презентация и защита готового проекта мобильного приложения.

3.2. Учебный (тематический) план

Модуль 2 (базовый уровень)

Таблица 2

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
Раздел 1. Повторение и углублённые конструкции языка программирования		32	12	20	
1.1	Инструктаж по ТБ. Повторение основ языка и интерфейса. Входное тестирование	4	1	3	Опрос, входное тестирование
1.2	Классы данных и перечисления	4	1	3	Опрос, практическая работа
1.3	Обработка ошибок и исключений	4	2	2	Опрос, практическая работа, устный опрос
1.4	Обобщённые типы (generics). Их применение в коллекциях	4	2	2	Опрос, практическая работа
1.5	Многopotочность/корутины	6	3	3	Опрос, практическая работа
1.6	Повторение принципов разработки приложений. Кейс-задание «Список задач»	6	2	4	Опрос, практическая работа
1.7	Знакомство с системой контроля версий. Командная работа над проектом	4	1	3	Опрос, практическая работа, тестирование
Раздел 2. Сетевое взаимодействие и работа с удалёнными данными		42	14	28	
2.1	Основы клиент-серверного взаимодействия. Понятие API, HTTP, JSON. Инструменты тестирования запросов	8	3	5	Опрос, практическая работа
2.2	Библиотеки для выполнения сетевых запросов. Получение данных от сервера	6	2	4	Опрос, практическая работа
2.3	Rest-взаимодействие	8	3	5	Опрос, практическая работа
2.4	Серверные СУБД	6	2	4	Опрос, практическая работа
2.5	Стек для работы над проектом. Выбор предварительной темы проекта. Составление плана работы над проектами	4	2	2	Опрос, практическая работа
2.6	Решение учебного кейс-задания	6	0	6	Практическая работа
2.7	Представление результатов. Рефлексия. Промежуточная аттестация	4	2	2	Защита кейса. Промежуточный контроль

Раздел 3. Подготовка к проектной деятельности. Изучение дополнительных модулей		30	11	19	
3.1	Анимация в интерфейсе. Создание плавных переходов	6	2	4	Опрос, практическая работа
3.2	Работа с камерой и галереей устройства	6	2	4	Опрос, практическая работа, устный опрос
3.3	Геолокация и картографические сервисы. Отображение местоположения	8	3	5	Опрос, практическая работа
3.4	Проектная деятельность, выбор темы. Архитектурные шаблоны проектирования	6	2	4	Опрос, практическая работа.
3.5	Сборка проектов, углубленная работа с системой контроля версий	4	2	2	Опрос, практическая работа.
Раздел 4. Проектная деятельность		40	3	37	
4.1	Паттерны дизайна приложений	4	2	2	Опрос. Практическая работа
4.2	Работа над проектами	22	0	22	Практическая работа
4.3	Подготовка презентаций проектов. Защита проекта	4	0	4	Защита проектов
4.4	Итоговая аттестация	4	0	4	Итоговая работа.
4.5	Итоговое занятие. Доработка проектов	6	1	5	Опрос. Практическая работа
Итого:		144	40	104	

Содержание учебного (тематического) плана

Раздел 1. Повторение и углублённые конструкции языка программирования

Тема 1.1. Инструктаж по ТБ. Повторение основ языка и интерфейса.

Входное тестирование

Теория: повторение правил техники безопасности при работе за компьютером. Основные конструкции языка.

Практика: выполнение входного тестирования. Решение задач для повторения материала.

Тема 1.2. Классы данных и перечисления

Теория: понятие исключения. Иерархия исключений. Проверяемые и непроверяемые исключения. Создание собственных исключений.

Практика: использование классов в приложении. Решение задач.

Тема 1.3. Обработка ошибок и исключений

Теория: парадигма ООП наследование, инкапсуляция, полиморфизм. Абстрактные методы и классы, интерфейсы.

Практика: добавление обработки ошибок в существующее приложение: перехват исключений при вводе некорректных данных. Создание своего исключения для проверки возраста пользователя.

Тема 1.4. Обобщённые типы (generics). Их применение в коллекциях

Теория: параметризованные типы. Обобщённые классы, интерфейсы и функции.

Практика: создание обобщённого класса-обёртки для хранения данных любого типа. Написание обобщённой функции поиска элемента в списке.

Тема 1.5. Многопоточность и корутины

Теория: понятие потока, проблемы многопоточности. Корутины как легковесные потоки. Основные строительные блоки. Диспетчеры.

Практика: отображение прогресса загрузки, обработка результатов.

Тема 1.6. Повторение принципов разработки приложений.

Кейс-задание «Список задач»

Теория: планирование структуры приложения для кейс-задания.
Повторение архитектур.

Практика: разработка полноценного приложения «Список задач». Защита выполненного задания.

Тема 1.7. Знакомство с git. Командная работа над проектом

Теория: применение средства контроля версий, основные команды. Ветвление и слияние веток. Конфликты и их разрешение.

Практика: настройка репозитория на веб-платформе. Совместная работа над проектом с использованием команд системы контроля версий. Решение конфликта слияния веток.

Раздел 2. Сетевое взаимодействие и работа с удалёнными данными

Тема 2.1. Основы клиент-серверного взаимодействия. Понятие API, HTTP, JSON. Инструменты тестирования запросов

Теория: архитектура «клиент-сервер». Протокол HTTP. Заголовки, коды ответов. RESTful API. Инструменты для отладки запросов.

Практика: изучение документации открытого API. Выполнение запросов через Postman, анализ ответов. Получение списка постов, создание нового поста.

Тема 2.2. Библиотеки для выполнения сетевых запросов. Получение данных от сервера

Теория: настройка, создание интерфейса API, добавление конвертера. Выполнение GET-запроса. Обработка ответа в корутине.

Практика: подключение библиотек. Создание data классов для ответа сервера.

Тема 2.3. Rest-взаимодействие

Теория: стили клиент-серверного взаимодействия. Облачные сервисы для хостинга серверной части приложения.

Практика: реализация мини проекта с Rest-взаимодействием. Отображение состояний загрузки и ошибок.

Тема 2.4. Серверные СУБД

Теория: понятие серверной базы данных. Реляционные СУБД. Соединение с БД через API (краткий обзор, без глубокого погружения). Без серверные решения.

Практика: сохранение и получение данных из без серверных решений в приложении (альтернативно – работа с локальной БД как аналогом серверной).

Тема 2.5. Стек для работы над проектом. Выбор предварительной темы проекта. Составление плана работы над проектами

Теория: понятие стека. Распространенные стеки в мобильной разработке. Анализ стеков компаний на «рынке».

Практика: подбор технологического стека для конкретного проекта.

Тема 2.6. Решение учебного кейс-задания

Практика: разработка приложения с использованием сетевого взаимодействия (например, приложение для просмотра погоды, новостей или галереи изображений).

Тема 2.7. Представление результатов. Рефлексия. Промежуточная аттестация

Теория: анализ выполненной работы, обсуждение возникших трудностей и способов их решения. Критерии оценки проектов.

Практика: презентация разработанного приложения перед группой. Проведение промежуточной аттестации. Обмен опытом, рефлексия.

Раздел 3. Подготовка к проектной деятельности. Изучение дополнительных модулей

Тема 3.1. Анимация в интерфейсе. Создание плавных переходов

Теория: виды анимаций: анимация изменения состояния, анимация появления/исчезновения, анимация жестов. Понятие спринговой анимации.

Практика: работа с анимацией. Анимация появления нового элемента в списке. Создание анимированного переключателя тем.

Тема 3.2. Работа с камерой и галереей устройства

Теория: запрос разрешений. Использование намерений для вызова камеры и выбора изображения из галереи. Знакомство с библиотеками.

Практика: реализация функции выбора фото из галереи и отображения его в приложении. Добавление возможности сделать фото и сохранить его в локальное хранилище.

Тема 3.3. Геолокация и картографические сервисы. Отображение местоположения

Теория: получение текущего местоположения, запрос разрешений. Интеграция картами, добавление маркеров, управление камерой.

Практика: создание приложения, отображающего текущее местоположение на карте с возможностью поставить метку и сохранить координаты. Работа с картой и геоданными.

Тема 3.4. Проектная деятельность, выбор темы. Архитектурные шаблоны проектирования

Теория: понятие архитектурного шаблона. Обзор MVVM, MVP, MVI. Углубление в MVVM: роль репозиториев, источники данных (локальные и удалённые). Использование Clean Architecture (обзорно). Принципы SOLID.

Практика: выбор окончательной темы для группового проекта. Проектирование архитектуры будущего приложения с использованием MVVM и репозиториев.

Тема 3.5. Сборка проектов, углублённая работа с системой контроля версий

Теория: ветвление и слияние в системе контроля версий. Разрешение конфликтов слияния. Работа с тегами.

Практика: создание веток для разных функций, слияние их в основную ветку. Разрешение конфликтов при совместной работе. Настройка .gitignore. Подготовка репозитория к командной разработке.

Раздел 4. Проектная деятельность

Тема 4.1. Паттерны дизайна приложений

Теория: понятие паттернов проектирования. Обзор наиболее часто используемых паттернов в мобильной разработке

Практика: анализ выбранного проекта: какие паттерны можно применить? Проектирование экранов с учётом дизайн-системы, создание собственной темы, использование продвинутых компонентов.

Тема 4.2. Работа над проектами

Теория: запрос разрешений. Использование намерений для вызова камеры и выбора изображения из галереи. Знакомство с библиотеками.

Практика: разработка группового проекта в соответствии с выбранной темой и ТЗ. Работа ведётся итерационно (спринтами по 2 недели).

Тема 4.3. Подготовка презентаций проектов. Защита проекта

Практика: подготовка к презентации. Защита проекта.

Тема 4.4. Итоговая аттестация

Практика: выполнение итоговой работы (тестирование или практическое задание), охватывающей все темы второго года обучения. Оценка знаний и навыков, полученных за год.

Тема 4.5. Итоговое занятие. Доработка проектов

Теория: рефлексия. Перспективы дальнейшего развития проекта.

Практика: внесение финальных правок в проекты по результатам защиты и замечаниям.

3.3. Учебный (тематический) план

Модуль 3 (продвинутый уровень)

Таблица 3

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
Раздел 1. Повторение и принципы проектирования приложений		64	20	44	
1.1	Инструктаж по ТБ. Обзор современных трендов в мобильной разработке	2	1	1	Опрос, входное тестирование
1.2	Повторение основ отрисовки приложений	8	2	6	Опрос, практическая работа
1.3	Повторение работы с сетевым взаимодействием и удаленными данными	16	4	12	Опрос, практическая работа, устный опрос
1.4	Архитектурные паттерны. Применение на практике	12	6	6	Опрос, практическая работа
1.5	Работа со сторонними библиотеками. Библиотеки внедрения зависимостей	12	4	8	Опрос, практическая работа
1.6	Разработка виджетов	8	2	6	Опрос, практическая работа
1.7	Решение кейс-задания. Промежуточный контроль	6	1	5	Опрос, практическая работа. Промежуточное тестирование
Раздел 2. Технологический стек и углубленные темы		18	8	10	
2.1	Продвинутая работа с сетью: загрузка файлов, обработка ошибок, кэширование	10	4	6	Опрос, практическая работа
2.2	Производительность и инструменты профилирования	4	2	2	Опрос, практическая работа
2.3	Оптимизация производительности	4	2	2	Опрос, практическая работа
Раздел 3. Проектная деятельность		62	13	49	
3.1	Прототипирование проектов	6	2	4	Опрос, практическая работа
3.2	Роли в команде. Выбор стека для разработки.	4	2	2	Опрос, практическая работа, устный опрос
3.3	Формирование ТЗ, определение мотивации пользователей	8	3	5	Опрос, практическая работа
3.4	Разработка минимально жизнеспособного продукта	16	2	14	Опрос, практическая работа
3.5	Тестирование и отладка	4	2	2	Опрос, практическая работа

3.6	Доработка проекта. Подготовки к защите проекта	16	2	14	Опрос, практическая работа
3.7	Защита проектов	4	0	4	Практическая работа
3.8	Итоговое занятие. Доработка проекта. Итоговая аттестация	4	0	4	Итоговое тестирование, опрос, практическая работа
Итого:		144	41	103	

Содержание учебного (тематического) плана

Раздел 1. Повторение и принципы проектирования приложений

Тема 1.1. Инструктаж по ТБ. Обзор современных трендов в мобильной разработке

Теория: повторение правил техники безопасности при работе за компьютером. Актуальные направления развития мобильной индустрии.

Практика: входное тестирование. Обсуждение интересов учащихся, выбор направления для углублённого изучения.

Тема 1.2. Повторение основ отрисовки приложений

Теория: повторение принципов декларативного UI. Компоновка элементов, модификаторы.

Практика: разработка небольшого приложения с использованием всех основных компонентов.

Тема 1.3. Повторение работы с сетевым взаимодействием и удалёнными данными

Теория: повторение основ клиент-серверного взаимодействия. Работа с различными API. Обработка ответов и ошибок. Кэширование данных.

Практика: создание приложения для работы с открытым API

Тема 1.4. Архитектурные паттерны. Применение на практике

Теория: углублённое изучение архитектурных паттернов: MVVM, MVI, Clean Architecture.

Практика: рефакторинг существующего приложения с применением Clean Architecture. Выделение доменного слоя, создание репозитория. Обсуждение полученной структуры.

Тема 1.5. Работа со сторонними библиотеками. Библиотеки внедрения зависимостей

Теория: понятие внедрения зависимостей. Проблемы ручного внедрения зависимостей. Обзор библиотек.

Практика: рефакторинг: замена ручного создания зависимостей на Hilt. Добавление модулей для сетевого слоя, базы данных.

Тема 1.6. Разработка виджетов

Теория: понятие виджетов. Возможности и ограничения. Жизненный цикл виджета. Создание конфигурационного экрана. Обновление виджета.

Практика: разработка простого виджета для отображения списка ближайших задач или курса валют. Интеграция виджета с приложением, обновление данных.

Тема 1.7. Решение кейс-задания. Промежуточный контроль

Теория: обобщение пройденного материала. Разбор типичных ошибок при проектировании архитектуры. Критерии оценки кейс-задания.

Практика: решение кейс-задания с использованием Clean Architecture, Hilt, сетевого взаимодействия. Проведение промежуточного тестирования.

Раздел 2. Технологический стек и углублённые темы

Тема 2.1. Продвинутая работа с сетью: загрузка файлов, обработка ошибок, кэширование

Теория: загрузка файлов на сервер. Перехват запросов, логирования, добавления заголовков.

Практика: обзор возможности загрузки изображений на сервер, реализация кэширования списков для офлайн-режима, обработка ошибок сети с повторными запросами.

Тема 2.2. Производительность и инструменты профилирования

Теория: ключевые метрики производительности: время запуска, плавность, потребление памяти, размер APK.

Практика: профилирование текущего проекта, выявление проблем. Оптимизация списков, устранение утечек памяти, анализ сетевых запросов.

Тема 2.3. Оптимизация производительности

Теория: методы оптимизации: ленивая загрузка. Минимализация перекомпиляции, стабильность типов.

Практика: примеры оптимизации производительности. Уменьшение перекомпиляций.

Раздел 3. Проектная деятельность

Тема 3.1. Прототипирование проектов

Теория: важность прототипирования. Инструменты. Создание пользовательских историй, информационная структура.

Практика: Прототипирование проекта в проектных группах. Защита прототипов.

Тема 3.2. Роли в команде. Выбор стека для разработки

Теория: распределение ролей в команде: тимлид, разработчик, дизайнер, тестировщик, аналитик. Обязанности каждой роли. Выбор технологического стека под проект.

Практика: команды определяют роли (с учётом пожеланий и навыков). Составляют итоговый стек технологий, обосновывают выбор. Создают репозиторий на веб-сервисе, настраивают проекты.

Тема 3.3. Формирование ТЗ, определение мотивации пользователей

Теория: методы генерации идей. Составление технического задания: цели, задачи, функциональные требования, нефункциональные требования, целевая аудитория

Практика: команды окончательно утверждают тему проекта. Написание развёрнутого технического задания, описание портрета пользователя, составление дорожной карты разработки.

Тема 3.4. Разработка минимально жизнеспособного продукта

Теория: выявление ключевого функционала, который обязательно должен быть реализован в первую очередь. Итеративная разработка, спринты.

Практика: команды приступают к реализации MVP. Проведение работы в соответствии с планом, используются система управления версиями.

Тема 3.5. Тестирование и отладка

Теория: виды тестирования: модульное (unit), интеграционное, UI-тестирование. Инструменты. Написание простых тестов. Отладка приложений, использование отладчика, логирование.

Практика: написание нескольких модульных тестов. Проведение ручного тестирования приложения, составление отчёта об ошибках. Исправление найденных багов.

Тема 3.6. Доработка проекта. Подготовка к защите проекта

Теория: отладка приложения: анимации, обработка граничных случаев, улучшение UX. Подготовка к публикации: создание иконок, скриншотов, описания. Структура презентации проекта.

Практика: финальная доработка функционала, добавление анимации, улучшение интерфейса. Подготовка презентации (слайды, демо-видео). Сборка подписанного APK/AAB.

Тема 3.7. Защита проектов

Практика: защита проекта.

Тема 3.8. Итоговое занятие. Доработка проекта.

Итоговая аттестация

Практика: внесение финальных правок в проекты по результатам защиты и замечаниям. Перспективы дальнейшего развития проекта. Выполнение итоговой работы (тестирование или практическое задание), охватывающей все темы обучения. Оценка знаний и навыков, полученных за период обучения.

4. Планируемые результаты

Предметные результаты:

- знание истории развития информационных технологий в России;
- знание современных и популярных платформ программирования мобильных приложений;
- знание основ языка программирования для создания мобильных приложений;
- умение использовать объектно-ориентированный подход в проектировании и разработке программного обеспечения;
- знание основных компонентов архитектуры приложения для мобильных устройств;
- применение навыков программирования технических устройств;
- знание современных нотаций и шаблонов программирования.

Метапредметные результаты:

- проявление умения искать, извлекать и отбирать нужную информацию из открытых источников;
- проявление умения самостоятельно ставить задачи, излагать мысли в четкой логической последовательности и отстаивать свою точку зрения;
- использование навыка предвидеть результат и достигать его, умение вносить коррективы в первоначальный замысел;
- проявление навыка исследовательской и проектной деятельности;
- знание правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Личностные результаты:

- проявление элементов этики групповой работы, отношений делового сотрудничества, взаимоуважения;
- проявление упорства в достижении результата;
- проявление организованности и ответственного отношения к труду;
- проявление ценностного отношения к своему здоровью;

- проявление российской гражданской идентичности: патриотизма, любви и уважения к Отечеству, чувства гордости за свою Родину.

4.1.Планируемые результаты

Модуль 1 (стартовый уровень)

Предметные результаты:

- знание основ работы персонального компьютера (ПК), его устройств, включая аппаратное и программное обеспечение;
- применение навыков использования ПК и интегрированной среды разработки для программирования устройства;
- знание современных и популярных платформ разработки мобильных приложений;
- умение применять на практике навык работы с запросами, сервером и базой данных;
- проявление навыка чтения кода программы, поиска ошибок в нём;
- проявление объектно-ориентированного подхода в проектировании и разработке программного обеспечения;
- знание основных элементов интерфейса мобильных приложений.

Метапредметные результаты:

- проявление навыка поиска, извлечения и отбора необходимой информации из открытых источников;
- знание основ проектной деятельности, этапов создания проекта;
- знание правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Личностные результаты:

- уважительное отношение при учебном сотрудничестве и совместной деятельности со сверстниками в процессе проектной и учебной деятельности;
- проявление целеустремлённости, организованности, ответственного отношения к труду и уважительного отношения к окружающим;
- проявление уважительного, позитивного отношения к окружающим, их мнению и деятельности.

4.2.Планируемые результаты

Модуль 2 (базовый уровень)

Предметные результаты:

- знание современных и популярных платформ;
- умение создавать архитектуру клиент-серверного взаимодействия с использованием программного интерфейса приложений;
- знание система контроля версий;
- умение работать с запросами, сервером и базами данных;
- уметь работать с основными видами алгоритмов, алгоритмических конструкций и приемов;
- знание языка структурированных запросов и языка разметки XML.

Метапредметные результаты:

- проявление алгоритмического мышления;
- проявление навыка исследовательской и проектной деятельности;
- проявление умения самостоятельно ставить и формулировать новые задачи, развивать мотивы своей познавательной деятельности.

Личностные результаты:

- проявление ответственного отношения к учению, целеустремленности и организованности;
- проявление навыков культуры командной работы и уважительного взаимодействия;
- участие в социально значимой деятельности, защита проектов.

4.3.Планируемые результаты

Модуль 3 (продвинутый уровень)

Предметные результаты:

- знание современных архитектурных паттернах (Clean Architecture, MVI) и принципов проектирования программного обеспечения;
- использование библиотек внедрения зависимостей (Hilt, Coin) при создании масштабируемых и тестируемых приложений;

- знание и практическое применение продвинутых сетевых методик, охватывающих загрузку файлов, кэширование данных, обработка сбоев и настройку повторных попыток;
- знание и практическое применение инструментов профилирования и оптимизации производительности мобильных приложений;
- проявление навыков командной работы с использованием систем контроля версий в рамках реального проекта;
- знание методов тестирования и отладки приложений.

Метапредметные результаты:

- проявление умения искать, извлекать и отбирать нужную информацию из открытых источников;
- проявление навыка самостоятельно формулировать цели и задачи проекта, доводить его до полной реализации;
- использование навыка предвидеть результат и достигать его, умение вносить коррективы в первоначальный замысел.

Личностные результаты:

- проявление упорства в достижении результата;
- проявление организованности и ответственного отношения к труду;
- проявление уважительного, позитивного отношения к окружающим, их мнению и деятельности.

II. Воспитание

1. Цель, задачи и целевые ориентиры воспитания

Воспитательная работа призвана обеспечить гармоничное сочетание образования с развитием личности, поддержать интерес к инновациям, стимулируя социальную активность. С целью содействия всестороннему развитию обучающихся, включая формирование их информационно-технологических компетенций, этического отношения к технологиям, а также укрепление морально-нравственных и гражданских ценностей, патриотического осознания обучающимися вклада российских изобретателей, инженеров в технологическое развитие человечества, образовательное учреждение осуществляет организацию различных воспитательных мероприятий.

В начале учебного года педагог-организатор проводит серию мероприятий в учебных группах на знакомство и сплочение коллектива. Показателем успешной организации воспитательной деятельности выступает вовлеченность обучающихся в проводимые мероприятия. Вовлеченность обучающихся – это уровень активного участия и эмоциональной включенности в процесс обучения.

Цель воспитания: формирование у обучающихся духовно-нравственных ценностей, гражданской идентичности, трудолюбия, ответственности, уважения к культурному наследию России и Свердловской области, а также развитие социально-эмоционального интеллекта и навыков коллективного взаимодействия.

Воспитательные задачи:

- содействовать присвоению обучающимися традиционных российских духовно-нравственных ценностей (жизнь, достоинство, права и свободы человека, патриотизм, гражданственность, служение Отечеству, созидательный труд, крепкая семья и др.);
- формировать у обучающихся активную жизненную позицию, готовность к участию в общественно значимой деятельности, волонтерстве, добровольчестве;

- развивать коммуникативные навыки, культуру поведения, умение работать в команде, эмпатию и уважение к другим людям;
- воспитывать ответственное отношение к своему здоровью, безопасному поведению в природной и городской среде.

Целевые ориентиры воспитания:

Программа воспитания направлена на гармоничное развитие личности в условиях цифровой среды. На стартовом уровне происходит формирование первоначальных представлений о нормах поведения в коллективе, бережном отношении к результатам труда, уважении к старшим. Базовый уровень направлен на развитие устойчивых навыков культурного поведения, осознанное участие в коллективных делах, проявление инициативы и ответственности. На продвинутом уровне отмечается сформированность гражданской позиции, готовность к социальному проектированию, наставничеству, передаче опыта младшим.

Программа воспитания обеспечивает освоение базовых навыков в информационных технологиях, всестороннее развитие личности – от цифровой этики до экологической ответственности.

2. Формы и методы воспитания

Формы и методы воспитания:

- беседы, дискуссии, этические диалоги;
- викторины, интеллектуальные игры, квесты;
- коллективные творческие дела (праздники, акции, фестивали, выставки);
- экскурсии.

3. Условия воспитания и анализ результатов

Условия воспитания:

- создание воспитывающей среды в учебной группе и структурном подразделении;
- поддержка детских инициатив, самоуправления;

- взаимодействие с семьёй, общественными организациями, учреждениями культуры и спорта;
- использование ресурсов региональной системы дополнительного образования, в том числе навигатора дополнительного образования детей Свердловской области.

Анализ результатов воспитания осуществляется через педагогическое наблюдение, анкетирование, анализ участия в мероприятиях, портфолио достижений обучающегося.

Оценивая результаты, педагог – организатор проводит наблюдение за обучающимися, отслеживает динамику развития отношения обучающихся к учебной деятельности, окружающей действительности и взаимодействию между собой.

4. Календарный план воспитательной работы (пример на 2026-2027 учебный год)

Таблица 4

№ п/п	Название события, мероприятия	Сроки	Форма проведения	Практический результат и информационный продукт, иллюстрирующий успешное достижение цели события
1.	«Детям о коррупции» - беседа с обучающимися	Сентябрь	Беседа, презентация	Формирование антикоррупционного мировоззрения у обучающихся. Фото и видеоматериалы.
2.	Игры на знакомство и командообразование в учебных группах.	Сентябрь-октябрь	Игра, тренинг	Объединение обучающихся в слаженный коллектив, развитие навыков работы в команде. Фото и видеоматериалы. Серия постов в официальном сообществе центра в социальной сети «ВКонтакте».
3.	Мастер-класс «Что такое проект: как найти тему и что может стать результатом»	Декабрь	Мастер-класс	Воспитание проектного мышления обучающихся, подготовка к проектной деятельности на учебных занятиях. Фото и видеоматериалы. Серия постов в официальном сообществе центра в социальной сети «ВКонтакте».

4.	Дни науки в IT- куб: интеллектуальная игра об открытиях и изобретениях	Февраль	Игра	Общеинтеллектуальное и гражданское воспитание, знакомство с достижениями русской науки. Фото и видеоматериалы. Серия постов в официальном сообществе центра в социальной сети «ВКонтакте».
5.	Комплекс мероприятий к Дню защитника Отечества	Февраль	Игры, мастер- классы, викторины	Презентация в учебной группе. Участие в объявленных конкурсах.
7.	Комплекс мероприятий ко Дню космонавтики	Апрель	Игры, мастер- классы, викторины	Презентация в учебной группе. Участие в объявленных конкурсах.
9.	Комплекс мероприятий ко Дню Победы в IT- куб	Май	Игры, мастер- классы, викторины	Общеинтеллектуальное и гражданское воспитание, знакомство с главными событиями русской истории. Фото и видеоматериалы. Серия постов в официальном сообществе центра в социальной сети «ВКонтакте».
10.	Информирование и привлечение обучающихся к участию в конкурсных мероприятиях разного уровня	В течение года	Конкурсы, соревнования, хакатоны, олимпиады	Сформировать стремление к познанию окружающего мира, к проектной деятельности в области IT-знаний. Фото и видеоматериалы. Серия постов в официальном сообществе центра в социальной сети «ВКонтакте».
11.	Информирование и привлечение обучающихся к участию в экскурсиях на промышленные предприятия района и города	В течение года	Экскурсии	Профориентация, знакомство с IT- предприятиями города, района. Фото и видеоматериалы. Серия постов в официальном сообществе центра в социальной сети «ВКонтакте».
12.	Информирование и привлечение обучающихся к участию в мероприятиях IT- куб	В течение года	Конкурсы, соревнования, хакатоны, олимпиады	Сформировать стремление к познанию окружающего мира, к проектной деятельности в области IT-знаний. Фото и видеоматериалы. Серия постов в официальном сообществе центра в социальной сети «ВКонтакте».

**III. Комплекс организационно-педагогических условий реализации
общеразвивающей программы**

1. Календарный учебный график на 2026–2027 учебный год

Таблица 5

№ п/п	Основные характеристики образовательного процесса	
1.	Количество учебных недель	36
2.	Количество учебных дней	72
3.	Количество часов в неделю	4
4.	Количество часов на период обучения	432
4.1	Количество часов на учебный год по каждому из модулей	144
5.	Недель в I полугодии	16
6.	Недель во II полугодии	20
7.	Начало занятий	07 сентября
8.	Выходные дни	31 декабря – 10 января
9.	Окончание учебного года	31 мая

2. Условия реализации общеразвивающей программы

Материально-техническое обеспечение

Требования к помещению:

- помещение для занятий, отвечающие требованиям СП 2.4.3648-20 для учреждений дополнительного образования;
- качественное освещение;
- столы, стулья по количеству обучающихся и рабочим местом для педагога.

Оборудование:

- Wi-Fi для поддержания online доступа к системе обучения;
 - графический планшет на каждого обучающегося и преподавателя;
 - доска магнитно-маркерная настенная;
 - интерактивная панель/дисплей;
 - монитор;
 - мультимедийный проектор либо интерактивная доска для показа презентаций;
 - МФУ формата А4;
 - наушники;
 - ноутбуки на каждого обучающегося и преподавателя, поддерживающие технологию Bluetooth 4.0;
 - планшет;
 - принтер 3D;
 - сетевое устройство;
 - сетевой удлинитель 3м (6 розеток);
 - сканер 3D;
 - смартфон;
 - тележка для ноутбуков;
 - флипчарт магнитно-маркерный на треноге.
- ***Расходные материалы:***

- бумага А4;
- бумага для флипчарта;
- маркеры для доски;
- перманентные маркеры;
- шариковые ручки.

Информационное обеспечение:

- Postman;
- SQL-lite;
- Yandex Browser;
- ПК для педагога, объединенный с функцией сервера;
- программное обеспечение МойОфис, LibreOffice;
- программное обеспечение BlueStacks;
- операционная система Linux;
- программное обеспечение Visual Studio;
- программное обеспечение, объединенные в локальную сеть.

Кадровое обеспечение

Реализовывать программу могут педагоги дополнительного образования, имеющие высшее образование (бакалавриат, специалитет или магистратура), среднее профессиональное образование, владеющие педагогическими методами и приемами, обладающие профессиональными знаниями и компетенциями в организации и проведении образовательной деятельности.

Методическое обеспечение:

Методические пособия, разработанные преподавателем с учетом конкретных задач, варианты демонстрационных программ, материалы по терминологии ПО, инструкции по настройке оборудования, учебная и техническая литература. Используются педагогические технологии индивидуализации обучения и коллективной деятельности.

3. Формы аттестации и оценочные материалы

Система контроля знаний и умений обучающихся представляется в виде учёта индивидуальных результатов по итогам выполнения практических заданий и посредством наблюдения, отслеживания динамики развития обучающегося.

Система отслеживания результатов обучающихся выстроена следующим образом:

- входная диагностика;
- промежуточный контроль;
- итоговый контроль.

Входная диагностика определения уровня умений, навыков, развития обучающихся в области компьютерной грамотности проводится в начале обучения согласно предложенной форме и разрабатывается для каждого модуля программы отдельно (Приложение 1, Приложение 4, Приложение 7). Входная диагностика отвечает педагогическому запросу, позволяющему отследить уровень знаний обучающихся на начальном этапе, и проводится педагогом.

Текущий контроль усвоения учебного материала осуществляется на каждом занятии по средствам опросов, педагогического наблюдения. Проверка знаний и умений обучающихся в форме педагогического наблюдения осуществляется в процессе выполнения ими практических заданий.

Промежуточный контроль проводится в форме контрольного тестирования (Приложение 2, Приложение 5, Приложение 8). Максимальное количество баллов за выполнение промежуточного тестирования по каждому из модулей составляет 25.

Для подведения итогов по окончании обучения проводится итоговое тестирование, которое позволяет выявить уровень освоения учебного материала обучающимися. (Приложение 3, Приложение 6, Приложение 9). Максимальное количество баллов за выполнение итогового тестирования по каждому из модулей составляет 25.

В завершении курса обучающиеся создают индивидуальный или групповой итоговый проект и готовят защитное слово. Максимальное

количество баллов – 50. Оценка финального проекта обучающихся проводится по каждому модулю. Оцениваются как подготовленные обучающимися проекты, так и умение презентовать их. Для этого педагог заполняет предложенный бланк оценки итоговых проектов, выставяя баллы каждому обучающемуся (Приложение 10).

Итоговая аттестация обучающихся по каждому модулю осуществляется путём суммирования баллов за промежуточную аттестацию, итоговое тестирование и защиту итогового проекта. Сумма баллов результатов промежуточной аттестации и итогового проекта переводится в один из уровней освоения образовательной программы согласно Таблица 6:

Уровень освоения программы по окончании обучения

Таблица 6

Баллы, набранные обучающимися	Уровень освоения
1-39	Низкий
40-79	Средний
80-100	Высокий

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

4.1 Перечень диагностического материала для осуществления мониторинга личностных и метапредметных планируемых результатов

Оценивая личностные и метапредметные результаты, педагог проводит наблюдение за обучающимися, отслеживание динамики изменения их творческих, коммуникативных и иных способностей.

1. Мониторинг достижения обучающимися личностных результатов (Приложение 11, Приложение 13, Приложение 15);

2. Мониторинг достижения обучающимися метапредметных результатов (Приложение 12, Приложение 14, Приложение 16).

4. Методические материалы

Особенности организации образовательного процесса

Образовательный процесс осуществляется очно, возможна реализация очно с применением электронного обучения и дистанционных образовательных технологий.

Также возможна реализация дополнительной общеобразовательной общеразвивающей программы в сетевой форме. ГАНОУ СО «Дворец молодёжи» является базовой организацией, организация-участник определяется на основании заключенного договора о сетевой форме реализации программ.

Методы обучения:

- исследовательский;
- метод проектов;
- наглядный;
- объяснительно-иллюстративный;
- проблемный (постановка проблемы и решение её самостоятельно или группой).

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

Методы воспитания:

- мотивация;
- поощрение;
- создание ситуации успеха;
- стимулирование;
- убеждение и др.

Формы организации образовательного процесса:

- групповая;
- индивидуальная;

- индивидуально-групповая.

Формы организации учебного занятия:

- беседа;
- защита проектов;
- кейс;
- открытое занятие;
- практическое занятие;
- соревнование;
- тестирование.

Педагогические технологии:

- группового обучения;
- дистанционного обучения;
- дифференцированного обучения;
- здоровьесберегающие технологии;
- игровой деятельности;
- индивидуализации обучения;
- исследовательской деятельности;
- коллективного взаимообучения;
- коллективной творческой деятельности;
- коммуникативная технология обучения;
- проблемного обучения;
- проектной деятельности;
- развивающего обучения;
- решения изобретательских задач.

Дидактические материалы:

Методические пособия, разработанные преподавателем с учётом конкретных задач; таблицы, платы, рисунки, чертежи; фотографии, видеофильмы; конструкции деталей судомоделей.

Список литературы

Список литературы, использованной при написании программы:

1. Android. Полное руководство по разработке приложений от сообщества Stack Overflow / [сост. коллектив авторов]; пер. с англ. ООО «Интеджер». – М.: АСТ, 2024. – 640 с.
2. Гриффитс Д., Гриффитс Д. Head First. Программирование для Android на Kotlin / Д. Гриффитс, Д. Гриффитс. – СПб: Питер, 2023. – 912 с.
3. Дарвин Я.Ф. Android. Сборник рецептов. Задачи и решения для разработчиков приложений / Я.Ф. Дарвин. – М.: Диалектика, 2018. – 768 с.
4. Жемеров Д. Kotlin в действии / Д. Жемеров, С. Исакова. – М.: ДМК Пресс, 2018. – 363 с.
5. Коул Р. Блистательный Agile. Гибкое управление проектами с помощью Agile, Scrum и Kanban / Р. Коул, Э. Скотчер. – СПб.: Питер, 2019. – 304 с.
6. Пименов С. Язык программирования Kotlin / С. Пименов. – М.: Агенция Айпио, 2017. – 304 с.
7. Рындак В.Г., Москвина А.В., Пак Л.Г. Педагогика: теория и методика воспитательной работы : учебное пособие / В.Г. Рындак, А.В. Москвина, Л.Г. Пак. – М. : ИНФРА-М, 2023. – 334 с.
8. Федотенко М.А. Разработка мобильных приложений. Первые шаги / М.А. Федотенко; под ред. В.В. Тарапаты. — М.: Лаборатория знаний, 2020. – 336 с.
9. Шуваев Я.А. UX/UI дизайн для создания идеального продукта. Полный и исчерпывающий гид / Я.А. Шуваев. – М.: Бомбора, 2022. – 240 с.

Электронные ресурсы:

1. 20+ инструментов и ресурсов для новичков в Android-разработке. [Электронный ресурс]. URL: <https://blog.skillfactory.ru/instrumenti-i-resursi-dlya-android-razrabotki/> (дата обращения: 23.03.2026).
2. Android Developers – сборник документации и руководств по разработке мобильных приложений под Android. [Электронный ресурс]. URL: <https://developer.android.com/> (дата обращения: 19.03.2026).

3. Code Basics: обучение базовым аспектам языков программирования от образовательной платформы Hexlet. [Электронный ресурс]. URL: <https://ru.code-basics.com/> (дата обращения: 19.03.2026).

4. Степанов П.В. Воспитательная деятельность как система. [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/vospitatelnaya-deyatelnost-kak-sistema/viewer> (дата обращения: 14.05.2026).

Литература, рекомендованная обучающимся:

1. Аделекан И. Kotlin: программирование на примерах / И. Аделекан. – СПб.: БХВ-Петербург, 2020. – 432 с.

2. Болье Алан. Изучаем SQL: генерация, выборка и обработка данных / Алан Болье; перевод с английского и редакция И.В. Красикова. – 3-е изд. – Москва; Санкт-Петербург: Диалектика, 2021. – 400 с.

3. Дарвин Я.Ф. Android. Сборник рецептов. Задачи и решения для разработчиков приложений / Я.Ф. Дарвин. – М.: Диалектика, 2018. – 768 с.

4. Дункан М. От Java к Kotlin: руководство по рефакторингу: перевод с английского / Дункан Макгрегор, Нэт Прайс. – СПб: БХВ-Петербург, 2023. – 442 с.

5. Объектно-ориентированные системы: реализация паттернов проектирования: учебное пособие / З.С. Сейдаметова, Ю.П. Москалева, З.Ш. Абдураманов, Г.С. Сейдаметов; Министерство образования, науки и молодежи Республики Крым, Крымский инженерно-педагогический университет имени Февзи Якубова. – Симферополь: ИП Хотеева Л.В., 2021. – 139 с.

Пример входной диагностики
Модуль 1 (стартовый уровень)
(максимальное количество баллов – 10)

1. За минимальную единицу измерения количества информации принимают. (1 балл)

- | | |
|------------|--------|
| а) байт | в) бит |
| б) пиксель | г) бот |

2. Что такое Android? (1 балл)

- | | |
|-----------------|----------------------------|
| а) телефон | в) веб-приложение |
| б) мобильная ОС | г) искусственный интеллект |

3. Получено сообщение, информационный объем которого равен 32 битам. Чему равен этот объем в байтах? (1 балл)

- | | |
|------|------|
| а) 5 | в) 3 |
| б) 2 | г) 4 |

4. Устройство, выполняющее арифметические и логические операции и управляющее другими устройствами компьютера, называется. (1 балл)

- | | |
|---------------|--------------|
| а) контроллер | в) монитор |
| б) клавиатура | г) процессор |

5. Файл – это (1 балл)

- | | |
|---|-------------------------------------|
| а) программа в ОП | в) единица измерения информации |
| б) программа или данные на диске, имеющие имя | г) текст, распечатанный на принтере |

6. Программы, управляющие оперативной памятью, процессором, внешними устройствами и обеспечивающие возможность работы других программ, называют (1 балл)

- | | |
|-------------|-----------------------------|
| а) утилиты | в) операционные системы |
| б) драйверы | г) системы программирования |

7. Какое из устройств предназначено для ввода информации? (1 балл)

- | | |
|--------------|---------------|
| а) процессор | в) клавиатура |
| б) принтер | г) монитор |

8. Текстовый редактор - программа, предназначенная для (1 балл)

- а) создания, редактирования и форматирования текстовой информации;
- б) работы с изображениями в процессе создания игровых программ;
- в) управление ресурсами ПК при создании документов;
- г) автоматического перевода с символьных языков в машинные коды.

9. Курсор – это (1 балл)

- а) устройство ввода текстовой информации;
- б) клавиша на клавиатуре;
- в) наименьший элемент отображения на экране;
- г) метка на экране монитора, указывающая позицию, в которой будет отображен текст, вводимый с клавиатуры.

10. Переменная – это (1 балл)

- а) ячейка для хранения и записи данных, имеющая имя;
- б) место именованное в коде;
- в) бесконечно и случайно изменяющееся значение;
- г) изменение в коде, приводящее к ошибке.

Пример промежуточного контроля
Модуль 1 (стартовый уровень)
(максимальное количество баллов – 25)

- 1. Напишите программу для вычисления суммы чисел а и b введенных пользователем (3 балла)**
 Вычислите $a+b$

Исходные данные
 а и b

Результат
 $a+b$

Пример

Исходные данные	Результат
1 5	6

- 2. Помогите локализаторам автоматизировать процесс – напишите программу, которая по количеству монстров выдаст соответствующее этому количеству слов. (5 баллов)**

Локализаторы стараются перевести игру, но исключить все числа, заменив их на слова. Они составили таблицу, в которой указали правила замены чисел, обозначающих количество монстров, на существительные и местоимения:

Количество	Обозначение на русском языке	Обозначение на языке аиндильяква
от 1 до 4	несколько	few
от 5 до 9	немного	several
от 10 до 19	отряд	pack
от 20 до 49	толпа	lots
от 50 до 99	орда	horde
от 100 до 249	множество	throng
от 250 до 499	сонмище	swarm
от 500 до 999	полчище	zounds
от 1000	легион	legion

Исходные данные

В единственной строке записано целое число n ($1 \leq n \leq 2000$) — количество монстров.

Результат

Выведите слово, соответствующее данному количеству монстров на языке аиндилъяква.

Примеры

Исходные данные	Результат
7	several
40	lots

3. Решите задачу (3 балла)

Необходимо обработать боевой космический корабль. Известно, что на обработку одного квадратного метра панели требуется 1 нанограмм сульфида. Всего необходимо обработать N прямоугольных панелей размером A на B метров. Вам необходимо как можно скорее подсчитать, сколько всего сульфида необходимо на обработку всех панелей «Энии». И не забудьте, что панели требуют обработки с обеих сторон.

Исходные данные

Единственная строка содержит целые числа N ($1 \leq N \leq 100$), A ($1 \leq A \leq 100$), B ($1 \leq B \leq 100$).

Результат

Выведите вес необходимого для обработки сульфида тория в нанограммах.

Пример

Исходные данные	Результат
5 2 3	60

4. Решите задачу (7 баллов)

n школьников делят k яблок поровну, неделящийся остаток остается в корзинке. Сколько яблок достанется каждому школьнику? Сколько яблок останется в корзинке?

Исходные данные

Программа получает на вход числа n и k и должна вывести искомое количество яблок (два числа).

Результат

Выведите два числа. Сколько яблок достанется каждому школьнику?
Сколько яблок останется в корзинке?

Примеры

Исходные данные	Результат
6	8
50	2

5. Напишите программу (7 баллов)

Дан список чисел. Выведите все элементы списка, которые больше предыдущего элемента.

Исходные данные

Программа получает на вход список одной строкой.

Результат

Элементы списка, больше предыдущего.

Примеры

Исходные данные	Результат
1 5 2 4 3	5
	4

Пример итогового контроля
Модуль 1 (стартовый уровень)
(максимальное количество баллов – 25)

1. Основные понятия объектно-ориентированного подхода в программировании (теория). Создание нового класса на основе уже существующего называют: (1 балл)

- | | |
|-----------------|---------------------------|
| а) наследование | в) декомпозиция программы |
| б) полиморфизм | г) инкапсуляция |

2. Какой тип результата вернет выражение `val x = true && false || true`? (1 балл)

- | | |
|-----------|------------|
| а) String | в) Boolean |
| б) Int | г) Double |

3. Чем отличается цикл `do...while` от цикла `while`? (1 балл)

- | | |
|---|--|
| а) они идентичны | в) ничего общего между ними нет |
| б) условие проверяется перед выполнением тела цикла в <code>do...while</code> | г) тело цикла выполняется хотя бы один раз в <code>do...while</code> |

4. Как называется встроенная структура данных, представляющая собой набор элементов, доступ к которым осуществляется по индексу? (1 балл)

- | | |
|---------|----------|
| а) Map | в) Set |
| б) List | г) Queue |

5. Когда возникает ошибка `NullPointerException`? (1 балл)

- а) при делении числа на ноль
- б) если ссылка ссылается на null и производится попытка вызвать метод объекта
- в) при выходе индекса массива за пределы допустимого диапазона
- г) если класс наследуется неправильно

6. Что такое намерение (Intent)? (1 балл)

- а) метод отправки сообщений между компонентами Android-приложений
- б) средство навигации между экранами приложения
- в) метаданные о поведении компонента
- г) контекст текущего окна приложения

7. Что произойдет, если возникнет исключение при выполнении блока кода, защищенного оператором `try-catch`? (1 балл)

- а) приложение автоматически завершится
- б) исключение перехватывается блоком `catch`, и программа продолжает работу
- в) программа переходит к обработке следующего оператора
- г) исключение игнорируется системой

8. Зачем нужен адаптер в Android-разработке? (2 балла)

- а) создает дополнительные элементы UI
- б) управляет сетевыми соединениями
- в) устанавливает связь между данными и элементами списка/грида
- г) сохраняет состояние приложения при закрытии

9. Что означает термин хеширование? (2 балла)

- а) шифрование паролей с использованием обратимых алгоритмов
- б) преобразование строки в фиксированную последовательность символов с целью безопасного хранения
- в) процесс оптимизации загрузки ресурсов
- г) кодировка файлов мультимедиа

10. Какой из следующих методов используется для обработки нажатия кнопки в *View.OnClickListener*? (2 балла)

- а) `onClick()`
- б) `onTouch()`
- в) `onPress()`
- г) `onSelect`

11. Напишите программу что будет выведено пользователю в результате работы этой программы: (4 балла)

```
fun main() {  
    val tom = Person()  
    tom.name = "Tom"  
    tom.age = 37  
  
    tom.sayHello()  
    tom.go("the shop")  
    println(tom.personToString())  
}  
  
class Person{  
    var name: String = "Undefined"  
    var age: Int = 18  
  
    fun sayHello(){  
        println("Hello, my name is $name")  
    }  
  
    fun go(location: String){  
        println("$name goes to $location")  
    }  
  
    fun personToString() : String{  
        return "Name: $name Age: $age"  
    }  
}
```

12. Калькулятор площадей фигур. Создайте приложение-калькулятор, вычисляющее площадь круга и прямоугольника. Реализуйте форму ввода значений радиуса и сторон фигуры, вывод результатов вычислений и обработку возможных исключительных ситуаций (например, ввод отрицательных чисел). (8 баллов)

Входная диагностика
Модуль 2 (базовый уровень)
(максимальное количество баллов – 20)

1. Какой класс является основным строительным блоком для компонентов пользовательского интерфейса (UI), определяет прямоугольную область экрана и отвечает за прорисовку и обработку событий? (1 балл)

- | | |
|---------|----------------|
| а) GUI | в) UIComponent |
| б) View | г) Widget |

2. Какой слушатель используется для отслеживания события касания экрана устройства? (1 балл)

- | | |
|--------------------|--------------------|
| а) OnPressListener | в) OnClickListener |
| б) onTouchListener | г) OnInputListener |

3. Основные понятия объектно-ориентированного подхода в программировании (теория). Создание нового класса на основе уже существующего называют: (2 балла)

- | | |
|-----------------|---------------------------|
| а) наследование | в) декомпозиция программы |
| б) полиморфизм | г) инкапсуляция |

4. Ввод, вывод и исключения. Ключевое слово, показывающее, что в процессе работы метода могут произойти не обработанные исключения: (2 балла)

5. Выберите верные утверждения для Context и Intent: (2 балл)

- а) класс android.context.Context используется для доступа к глобальной информации об окружении Android приложения;
- б) для передачи намерения в Activity можно использовать метод startActivity();
- в) метод putExtra() предназначен для сохранения данных в Intent;
- г) явным (explicit) называется Intent, который содержит флаг Intent.FLAG_EXPLICIT;
- д) для передачи намерения в Activity можно использовать любой метод.

6. Напишите программу что будет выведено пользователю в результате работы этой программы: (3 балла)

```
fun main() {
    val tom = Person()
    tom.name = "Tom"
    tom.age = 37

    tom.sayHello()
}
```

```

    tom.go("the shop")
    println(tom.personToString())

}

class Person{
    var name: String = "Undefined"
    var age: Int = 18

    fun sayHello(){
        println("Hello, my name is $name")
    }

    fun go(location: String){
        println("$name goes to $location")
    }

    fun personToString() : String{
        return "Name: $name Age: $age"
    }
}

```

7. Напишите программу, которая считывает целое число, после чего на экран выводится следующее и предыдущее целое число с пояснительным текстом (3 балла).

8. Напишите простой приложение-калькулятор, которое выполняет базовые арифметические операции (сложение, вычитание, умножение, деление). (6 баллов)

Пример промежуточного контроля
Модуль 2 (базовый уровень)
(максимальное количество баллов – 25)

1. Какой класс используется для создания адаптера для *RecyclerView*?
(2 балла)

- | | |
|-------------------------|----------------|
| а) ArrayAdapter | в) ListAdapter |
| б) RecyclerView.Adapter | г) BaseAdapter |

2. Какой из следующих методов используется для обработки нажатия кнопки в *View.OnClickListener*? *(2 балла)*

- | | |
|--------------|---------------|
| а) onClick() | в) onPress() |
| б) onTouch() | г) onSelect() |

3. Какой метод используется для выполнения HTTP-запроса с помощью библиотеки *Retrofit*? *(2 балла)*

- | | |
|--------------|--------------|
| а) execute() | в) enqueue() |
| б) call() | г) request() |

4. Какой из следующих классов используется для работы с базами данных в *Android*? *(2 балла)*

- | | |
|-------------------|--------------------|
| а) SQLiteDatabase | в) RoomDatabase |
| б) DatabaseHelper | г) ContentProvider |

5. Какой метод используется для добавления элемента в *ArrayList*?
(2 балла)

- | | |
|-------------|-------------|
| а) add() | в) append() |
| б) insert() | г) push() |

6. Какой метод используется для получения данных из *Bundle*?
(2 балла)

- | | |
|----------------|---------------|
| а) get() | в) retrieve() |
| б) getString() | г) fetch() |

7. Какой из следующих классов используется для создания HTTP-сервера на *Kotlin*? *(2 балла)*

- | | |
|-------------|-----------|
| а) Ktor | в) OkHttp |
| б) Retrofit | г) Volley |

8. Какой из следующих классов используется для работы с JSON в Android? (2 балла)

а) JsonParser

в) JsonReader

б) Gson

г) JsonObject

9. Напишите простое приложение, которое подгружает фрагменты. Первый из которых содержит формулу, объясняющую перевод часов в минуту и минуту в секунду. Второй фрагмент содержит калькулятор по переводу значений времени. Например: при вводе 60 минут в поле час выводилось 1, а в поле секунды выводилось 3600. (9 баллов)

Пример итогового контроля
Модуль 2 (базовый уровень)
(максимальное количество баллов – 25)

1. Какой из следующих классов используется для работы с базами данных в Android? (1 балл)

- | | |
|-------------------|--------------------|
| а) SQLiteDatabase | в) RoomDatabase |
| б) DatabaseHelper | г) ContentProvider |

2. Какой из следующих классов используется для создания HTTP-сервера на Kotlin? (2 балла)

- | | |
|-------------|-----------|
| а) Ktor | в) OkHttp |
| б) Retrofit | г) Volley |

3. Какой протокол используется для передачи данных между клиентом и сервером в веб-приложениях? (2 балла)

- | | |
|---------|---------------|
| а) FTP | в) HTTP/HTTPS |
| б) SMTP | г) DNS |

4. Зачем нужен контроль версий (например, Git) при совместной разработке проект? (2 балла)

- а) только для отслеживания изменений в коде одним разработчиком
- б) чтобы создать возможность восстановления предыдущих состояний проекта в случае ошибок
- в) исключительно для автоматизации процесса деплоя приложения
- г) только для ускорения процесса тестирования приложения

5. Что такое Rest API? (2 балла)

- а) архитектурный стиль для построения распределенных систем, использующий стандартные методы HTTP
- б) технология шифрования данных при обмене информацией
- в) методика оптимизации изображений для веб-сайтов
- г) алгоритм сортировки массивов

6. Что понимается под Material Design в Android-разработке? (2 балла)

- а) библиотека для анимации UI-элементов
- б) стандарт оформления интерфейса приложений, разработанный Google
- в) механизм автоматической адаптации макетов под разные размеры экранов
- г) фреймворк для рендеринга трехмерных моделей

7. Опишите структуру и принцип работы мобильного приложения ToDo List (4 балла)

8. Напишите простое приложение с News API или любым другим на ваш выбор. Приложение, отправляет запрос к новостному сервису (через внешний API) и выводит заголовки новостей по определенной категории (например, technology, sports, health и др.). (10 баллов)

Входная диагностика
Модуль 3 (продвинутый уровень)
(максимальное количество баллов – 10)

1. Какой архитектурный паттерн предполагает разделение на слои Model, View, ViewModel? (1 балл)

- | | |
|---------|-----------------------|
| а) MVP | в) MVI |
| б) MVVM | г) Clean Architecture |

2. Какая аннотация в Hilt используется для обозначения класса, который будет предоставлять зависимости? (1 балл)

- | | |
|---------------|-----------------------|
| а) @Module | в) @Inject |
| б) @Singleton | г) @AndroidEntryPoint |

3. Для чего используется оператор suspend в Kotlin? (1 балл)

- а) для приостановки выполнения программы навсегда
- б) для объявления функции, которая может быть приостановлена и возобновлена без блокировки потока
- в) для остановки корутины
- г) для синхронизации потоков

4. Какие методы HTTP чаще всего используются в REST API? (1 балл)

- | | |
|--------------------------------|-----------------------------------|
| а) GET, POST, PUT, DELETE | в) FETCH, SEND, REMOVE, CHANGE |
| б) READ, WRITE, UPDATE, DELETE | г) SELECT, INSERT, UPDATE, DELETE |

5. Что такое data class в Kotlin? (1 балл)

- а) класс для работы с базами данных
- б) класс, автоматически генерирующий методы equals(), hashCode(), toString(), copy()
- в) класс, который может хранить только данные примитивных типов
- г) класс с обязательным конструктором без параметров

6. Какой компонент Jetpack Compose используется для отображения больших списков с эффективной переработкой элементов? (1 балл)

- | | |
|-----------|---------------|
| а) Column | в) LazyColumn |
| б) Row | г) Box |

7. Для чего используется Firebase Crashlytics? (1 балл)

- а) для анализа поведения пользователей
- б) для отправки push-уведомлений
- в) для сбора отчётов о сбоях приложения
- г) для аутентификации пользователей

8. Какая команда Git используется для загрузки изменений из удалённого репозитория в локальный? (1 балл)

- | | |
|-------------|---------------|
| а) git push | в) git commit |
|-------------|---------------|

б) git pull

г) git clone

9. Что такое ViewModel в Android? (1 балл)

а) класс для отображения данных на экране

б) компонент, сохраняющий данные при изменении конфигурации (поворот экрана)

в) часть базы данных Room

г) библиотека для работы с сетью

10. Какой метод используется для запуска корутины в viewModelScope?

(1 балл)

а) runBlocking

в) start

б) launch

г) execute

Пример промежуточного контроля
Модуль 3 (продвинутый уровень)
(максимальное количество баллов – 25)

Разработать небольшое приложение для отображения списка постов из открытого API с использованием Jetpack Compose, ViewModel, корутин и Retrofit. Например: «Книга рецептов», «Сборник новостей», «Сборник кинофильмов».

Приложение должно состоять из двух экранов:

1. Экран списка постов (отображает заголовки постов).
2. Экран деталей поста (при клике на пост открывается экран с заголовком, текстом поста и идентификатором пользователя).

Требования:

1. Использовать архитектуру MVVM.
2. Реализовать сетевой слой с Retrofit.
3. Применить корутины для асинхронных запросов.
4. Добавить обработку состояний загрузки и ошибок (прогресс-бар при загрузке, сообщение об ошибке при сбое).
5. Реализовать навигацию между экранами с передачей данных.

Пример итогового контроля
Модуль 3 (продвинутый уровень)
(максимальное количество баллов – 25)

1. Какой слой в Clean Architecture отвечает за бизнес-логику и не зависит от фреймворков? (1 балл)

- | | |
|-----------------|-------------------|
| а) Presentation | в) Domain |
| б) Data | г) Infrastructure |

2. Какая аннотация в Hilt используется для указания, что класс должен быть доступен для внедрения зависимостей в конструкторе? (1 балл)

- | | |
|--------------|---------------|
| а) @Inject | в) @Module |
| б) @Provides | г) @Singleton |

3. Что такое MVI (Model-View-Intent)? (1 балл)

- а) архитектурный паттерн с однонаправленным потоком данных и неизменяемым состоянием
- б) библиотека для тестирования пользовательского интерфейса
- в) система контроля версий
- г) фреймворк для верстки интерфейсов

4. Какой метод в Retrofit используется для загрузки файла на сервер с передачей нескольких частей (multipart)? (1 балл)

- | | |
|-----------------------|---------------|
| а) @Multipart и @Part | в) @Streaming |
| б) @Body | г) @File |

5. Для чего используется WorkManager? (1 балл)

- а) для выполнения отложенных и фоновых задач с гарантией выполнения
- б) для управления потоками в реальном времени
- в) для синхронизации данных с сервером в реальном времени
- г) для создания виджетов

6. Какая библиотека используется для профилирования сетевых запросов в Android Studio? (1 балл)

- | | |
|---------------------|-------------------------|
| а) LeakCanary | в) Firebase Performance |
| б) Network Profiler | г) Crashlytics |

7. Что такое baseline profiles в Android? (1 балл)

- а) набор классов и методов, компилируемых заранее для ускорения запуска приложения
- б) файлы с настройками сборки
- в) профили безопасности приложения
- г) шаблоны для создания новых проектов

8. Какая команда Git используется для объединения изменений из одной ветки в другую? (1 балл)

- а) git merge
- б) git branch
- в) git checkout
- г) git rebase (оба варианта возможны, но выберите наиболее распространённый)

9. Какой компонент Jetpack Compose используется для создания виджетов (App Widgets) в современном подходе? (1 балл)

- а) Glance
- б) ComposeView
- в) WidgetCompose
- г) RemoteViews

10. Что такое Flow в Kotlin? (1 балл)

- а) асинхронный поток данных, поддерживающий корутины
- б) графическая библиотека для анимации
- в) тип данных для работы с базами данных
- г) инструмент для логирования

11. Какая аннотация используется в Hilt для создания синглтона? (1 балл)

- а) @Singleton
- б) @Provides
- в) @Inject
- г) @Module

12. Какой инструмент позволяет анализировать потребление памяти приложением в реальном времени? (1 балл)

- а) Android Profiler
- б) Logcat
- в) Layout Inspector
- г) APK Analyzer

13. Для чего используется Paging 3 библиотека? (1 балл)

- а) для эффективной постраничной загрузки и отображения больших списков
- б) для пагинации в базах данных
- в) для навигации между экранами
- г) для работы с сетью

14. Что из перечисленного является целью тестирования программного обеспечения? (1 балл)

- а) проверка корректности работы отдельных компонентов программы
- б) проверка взаимодействия между компонентами системы
- в) проверка удобства использования интерфейса
- г) проверка работы программы в условиях, приближённых к реальным

15. Какая команда Git создаёт новую ветку и сразу переключается на неё? (1 балл)

- а) git checkout -b <branch>
- б) git branch <branch>
- в) git switch -c <branch> (оба варианта правильные, но выберите тот, который чаще используется в классическом синтаксисе)
- г) git merge <branch>

16. Какой метод используется для повторной попытки сетевого запроса в OkHttp? (2 балла)

- а) retryOnConnectionFailure
- б) addInterceptor
- в) authenticator
- г) connectionSpec

17. Что такое Compose UI Testing? (2 балла)

- а) библиотека для тестирования пользовательского интерфейса, написанного на Jetpack Compose
- б) инструмент для профилирования Compose-приложений

- в) фреймворк для создания анимаций в Compose
- г) средство отладки Compose-кода

18. Какой инструмент Android Studio позволяет в реальном времени отслеживать использование CPU, памяти и сети приложением? (2 балла)

- а) Android Profiler
- б) Logcat
- в) Layout Inspector
- г) APK Analyzer

19. Для чего используется DataStore (Preferences DataStore)? (2 балла)

- а) для хранения пар «ключ-значение» с поддержкой корутин
- б) для работы с реляционными базами данных
- в) для кэширования изображений
- г) для сохранения состояния при повороте экрана

20. Как в Clean Architecture организуются зависимости между слоями? (2 балла)

- а) зависимости направлены от внешних слоёв к внутренним (бизнес-логика не зависит от фреймворков и деталей реализации)
- б) зависимости направлены от внутренних слоёв к внешним (бизнес-логика зависит от фреймворков)
- в) зависимости отсутствуют, слои полностью изолированы
- г) зависимости могут быть любыми, в зависимости от требований

Бланк оценки итоговых проектов
(максимальное количество баллов - 50)

№ п/п	ФИ автора (ов)	Название проекта	Критерий 1 Актуальность проекта (0-10 б)	Критерий 2 Используемые инструменты (0-10 б)	Критерий 3 Практическая реализация, визуальная составляющая (0-10 б)	Критерий 4 Правильност ь написания кода программы (0-10 б)	Критерий 5 Защита проекта (представлени е работы) (0-10 б)	Итого
1.								
2.								
3.								
4.								
5.								
6.								
7.								
8.								

Мониторинг достижения обучающимися личностных результатов
Модуль 1 (стартовый уровень)

№ п/п	ФИО	ПОКАЗАТЕЛИ КАЧЕСТВА ЛИЧНОСТНОГО РАЗВИТИЯ ОБУЧАЮЩИХСЯ											
		Уважительное отношение при учебном сотрудничестве и совместной деятельности со сверстниками в процессе проектной и учебной деятельности			Проявление целеустремлённости, организованности, ответственного отношения к труду и уважительного отношения к окружающим			Проявление уважительного, позитивного отношения к окружающим, их мнению и деятельности			Итог		
		Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый
1													
2													
3													
4													
5													
6													
7													
8													

3 балла – качество проявляется систематически

2 балла – качество проявляется ситуативно

1 балл – качество не проявляется

Мониторинг достижения обучающимися метапредметных результатов
Модуль 1 (стартовый уровень)

№ п/п	ФИО	ПОКАЗАТЕЛИ КАЧЕСТВА МЕТАПРЕДМЕТНЫХ НАВЫКОВ ОБУЧАЮЩИХСЯ											
		Проявление навыка поиска, извлечения и отбора необходимой информации из открытых источников			Знание основ проектной деятельности, этапов создания проекта			Знание правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой			Итог		
		Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый
1													
2													
3													
4													
5													
6													
7													
8													

3 балла – качество проявляется систематически

2 балла – качество проявляется ситуативно

1 балл – качество не проявляется

Мониторинг достижения обучающимися личностных результатов

Модуль 2 (базовый уровень)

№ п/п	ФИО	ПОКАЗАТЕЛИ КАЧЕСТВА ЛИЧНОСТНОГО РАЗВИТИЯ ОБУЧАЮЩИХСЯ											
		Проявление ответственного отношения к учению, целеустремленности и организованности			Проявление навыков культуры командной работы и уважительного взаимодействия			Участие в социально значимой деятельности, защита проектов			Итог		
		Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый
1													
2													
3													
4													
5													
6													
7													
8													

3 балла – качество проявляется систематически

2 балла – качество проявляется ситуативно

1 балл – качество не проявляется

Мониторинг достижения обучающимися метапредметных результатов
Модуль 2 (базовый уровень)

№ п/п	ФИО	ПОКАЗАТЕЛИ КАЧЕСТВА МЕТАПРЕДМЕТНЫХ НАВЫКОВ ОБУЧАЮЩИХСЯ											
		Проявление алгоритмического мышления			Проявление навыка исследовательской и проектной деятельности			Проявление умения самостоятельно ставить и формулировать новые задачи, развивать мотивы своей познавательной деятельности			Итог		
		Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый
1													
2													
3													
4													
5													
6													
7													
8													

3 балла – качество проявляется систематически

2 балла – качество проявляется ситуативно

1 балл – качество не проявляется

Мониторинг достижения обучающимися личностных результатов
Модуль 3 (продвинутый уровень)

№ п/п	ФИО	ПОКАЗАТЕЛИ КАЧЕСТВА ЛИЧНОСТНОГО РАЗВИТИЯ ОБУЧАЮЩИХСЯ											
		Проявление упорства в достижении результата			Проявление организованности и ответственного отношения к труду			Проявление уважительного, позитивного отношения к окружающим, их мнению и деятельности			Итог		
		Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый
1													
2													
3													
4													
5													
6													
7													
8													

3 балла – качество проявляется систематически

2 балла – качество проявляется ситуативно

1 балл – качество не проявляется

**Мониторинг достижения обучающимися метапредметных результатов
Модуль 3 (продвинутый уровень)**

№ п/п	ФИО	ПОКАЗАТЕЛИ КАЧЕСТВА МЕТАПРЕДМЕТНЫХ НАВЫКОВ ОБУЧАЮЩИХСЯ											
		Проявление умения искать, извлекать и отбирать нужную информацию из открытых источников			Проявление навыка самостоятельно формулировать цели и задачи проекта, доводить его до полной реализации			Использование навыка предвидеть результат и достигать его, умение вносить коррективы в первоначальный замысел			Итог		
		Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый	Входной	Промежуточный	Итоговый
1													
2													
3													
4													
5													
6													
7													
8													

3 балла – качество проявляется систематически

2 балла – качество проявляется ситуативно

1 балл – качество не проявляется

Аннотация

Одним из основных аспектов, влияющих на внедрение информационных технологий в жизнедеятельность общества, является умение оперативно и качественно работать с мобильным устройством, используя для этого все современные технические и программные средства и методы.

Программа «Мобильная разработка» предназначена для изучения основных этапов разработки мобильных приложений и является унифицированной, что позволяет использовать при обучении разные языки программирования. Программа состоит из трёх модулей и рассчитана на три года обучения, что позволяет более подробно изучить создание мобильных приложений, освоить язык программирования. В результате освоения программы, обучающиеся приобретут теоретические знания и практические навыки программирования, а также смогут освоить основные принципы дизайна пользовательского интерфейса, смогут создавать прототипы и разрабатывать базовые приложения.

Программа «Мобильная разработка» имеет техническую направленность, ориентирована на развитие навыков программирования и проектирования программ под мобильные платформы.

Программа рассчитана на обучающихся 13 – 17 лет.