

Государственное бюджетное общеобразовательное учреждение
средняя общеобразовательная школа №422
Кронштадтского района Санкт-Петербурга

ПРИНЯТА

на заседании педагогического совета

Протокол № 1

от 29.08.2024г.

УТВЕРЖДЕНА

Директор ГБОУ СОШ №422

Кронштадтского района Санкт-Петербурга

 Богданова О.Н.

Приказом 96/Д-9 от 02.09. 2024г.



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«МОБИЛЬНАЯ РАЗРАБОТКА»**

Возраст учащихся: 14-17 лет

Срок реализации программы: 2 года

Составитель программы:

Кондратович Ю.С.

педагог дополнительного образования

Кронштадт, 2024

Распоряжение Правительства РФ от 29.05.2015 N 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»;

Приказ Министерства просвещения России от 09.11.2018 г. № 196 «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 труда и социальной защиты Российской Федерации от 05.05.2018 г. № 298н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»»;

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»;

Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242 «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»;

Методические рекомендациями для субъектов Российской Федерации по вопросам реализации основных и дополнительных общеобразовательных программ в сетевой форме, утвержденных Министерством просвещения России 28.06.2019г №МР-81/02вн;

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

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 учетом возможности работы на мобильных устройствах.

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 программы заключается в возможности получения обучающимися универсальных компетенций, необходимых при дальнейшем изучении не только информационных технологий. 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 IT-направлений.

Также данная программа является базой для перехода на более сложные программы обучения. Обучающиеся приобретают знания по основам IT, которые будут востребованы для дальнейшего обучения в профильных средних специальных и высших учебных заведениях.

В программе учтены требования, предъявляемые к участникам чемпионата WorldSkillsRussia по компетенции «Мобильная робототехника». Содержание и структура курса подготовки к чемпионатам направлены на формирование устойчивых представлений о мобильных робототехнических системах и применении их в различных сферах деятельности.

Адресат программы: Дополнительная общеразвивающая программа «Мобильная разработка» предназначена для детей в возрасте 14–17 лет, без ограничений возможностей здоровья. Количество обучающихся в группе – 13 человек. Состав групп постоянный.

Обучающимися могут стать подростки, являющиеся учениками общеобразовательных учреждений, с которыми заключён договор о сетевой форме реализации программ.

Место проведения занятий: г. Кронштадт, пр.Ленина д.6

Возрастные особенности группы

Содержание программы учитывает возрастные и психологические особенности детей в возрасте 14–17 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 14-17 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 детей. 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: длительность одного занятия составляет 2 академических часа по 45 минут, периодичность занятий – 1 раз в неделю.

Срок освоения общеразвивающей программы: определяется содержанием программы и составляет 2 года.

Форма обучения: очная, возможна реализация очно с применением электронного обучения и дистанционных образовательных технологий (Закон №273-ФЗ, гл.2, ст.17, п.2.).

Объём общеразвивающей программы: общее количество учебных часов, запланированных на весь период обучения, необходимых для освоения программы: 72 академических часа.

Форма организации образовательной деятельности – групповая.

По уровню освоения программа общеразвивающая, разноуровневая (стартовый и базовый уровни). Она обеспечивает возможность обучения детей с любым уровнем подготовки.

«Стартовый уровень» рассчитан на детей в возрасте с 14 лет, проявляющих интерес к аналитической и исследовательской деятельности, IT-технологиям, приобретению навыков программирования.

Зачисление детей на первый год обучения производится без предварительного отбора (свободный набор). К концу стартового уровня обучающиеся приобретут навыки поиска, анализа и использования информации, а также безопасного поведения в сети Интернет; получат навыки программирования в среде разработки IntelliJ IDEA IDE.

«Базовый уровень» рассчитан на детей, проявляющих интерес к созданию программ на языке Java для решения прикладных задач, желающих совершенствовать свои навыки программирования, имеющих опыт программирования в различных интегрированных средах разработки на языке Java.

Зачисление детей на базовый уровень после завершения стартового уровня производится по результатам успешной сдачи итогового тестирования.

2. Цели и задачи программы

Цель программы: формирование технической грамотности посредством приобщения обучающихся к разработке программ под современную платформу Android.

Для успешной реализации поставленной цели необходимо решить ряд педагогических, развивающих и воспитательных задач:

Обучающие:

- расширить знания о современных и популярных платформах;
- обучить языку программирования Java, языку разметки XML;
- обучить объектно-ориентированному подходу в проектировании и разработке программного обеспечения;
- познакомить с архитектурой приложения под Android;
- обучить программированию технических устройств.

Развивающие:

- сформировать трудовые умения и навыки, умение планировать работу, предвидеть результат и достигать его;
- сформировать и развить навыки работы с различными источниками информации, умение самостоятельно искать, извлекать и отбирать необходимую информацию;
- способствовать развитию умения излагать мысли в чёткой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путём логических рассуждений;
- познакомить с правилами индивидуального и коллективного безопасного поведения при работе с компьютерной техникой и оборудованием.

Воспитательные:

- способствовать воспитанию этики групповой работы, отношений делового сотрудничества, взаимоуважения;
- развивать основы коммуникативных отношений внутри проектных групп и в коллективе в целом;
- сформировать целеустремлённость, организованность, ответственное отношение к труду;
- способствовать воспитанию упорства в достижении результат.

3. Содержание общеразвивающей программы

1 год обучения «Стартовый уровень»

Таблица 1

№ п/п	Название раздела, темы	Кол-во часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
Модуль 1. Основы программирования.		30	8	22	
1.1	Вводное занятие. Инструктаж по ТБ. Введение в программу. Среда разработки	4	2	2	Знакомство. Опрос. Инструктаж по ТБ
1.2	Арифметика. Примитивные типы данных. Логика. Операции отношения и логические операции.	6	2	4	Опрос. Практическая работа
1.3	Условные конструкции. Блоки	4	1	3	Опрос. Практическая работа
1.4	Итеративные конструкции. Массивы. Списки.	4	1	3	Опрос. Практическая работа
1.5	Методы (функции). Видимость переменных. Рекурсия.	8	2	6	Опрос. Практическая работа
1.6	Повторение и закрепление	4		4	Практическая работа
Модуль 2. Объектно-ориентированное программирование		42	11	31	
2.1	Классы и объекты	4	1	3	Опрос. Практическая работа
2.2	Классы: конструкторы, статические методы. Начальные приёмы тестирования и отладки	4	1	3	Опрос. Практическая работа
2.3	Android. Структура. Активности. Интерфейс пользователя. Язык разметки XML.ООП.	16	6	10	Опрос. Практическая работа
2.4	Намерения. Фрагменты.	6	2	4	Опрос. Практическая работа
2.5	Практикум. Контрольное тестирование по темам 1 и 2 модулей	4	0	4	Выполнение контрольного тестирования
2.6	Профориентационное мероприятие	6	0	6	Экскурсии
2.7	Резервное время	2	1	1	
Итого 1 год		72	19	53	

2 год обучения «Базовый уровень»

Модуль 3. Основы программирования Android-приложений		26	6	20	
3.1	Ввод, вывод и исключения	4	1	3	Опрос. Практическая работа
3.2	Внутренние классы в обработке событий	4	1	3	Опрос. Практическая работа
3.3	Параллелизм и синхронизация. Поток	8	2	6	Опрос. Практическая работа
3.4	Двумерная графика в Android-приложениях	4	1	3	Опрос. Практическая работа
3.5	Реализация графики на основе SurfaceView	6	1	5	Опрос. Практическая работа
Модуль 4. Алгоритмы и структуры данных		32	11	21	
4.1	Массивы. Списки. Алгоритмы сортировки. Алгоритм двоичного поиска. Деревья	8	3	5	Опрос. Практическая работа
4.2	Адаптеры в Android	6	2	4	Опрос. Практическая работа
4.3	Ассоциативные массивы	6	2	4	Опрос. Практическая работа
4.4	Реляционная модель данных. СУБД. Введение в SQL	8	3	5	Опрос. Практическая работа
4.5	Контрольное тестирование по темам 3 и 4 модулей	4	1	3	Выполнение контрольного тестирования
Модуль 5. Проектная деятельность		14	1	13	
5.1	Консультации по ИП	8	0	8	Опрос. Подготовка итоговых проектов
5.2	Итоговая защита ИП	4	0	4	Защита итоговых проектов
	Резерв	2	1	1	
Итого 2 год		72	18	54	
Итого		144	37	107	

Содержание учебного плана

Стартовый уровень

Модуль 1. Основы программирования

Тема 1.1. Среда разработки

Теория: Среда разработки IntelliJ IDEA/Eclipse. Шаблон программы на Java с функцией main(). О среде разработки IntelliJ IDEA и Eclipse. Понятие проекта. Порядок создания, компиляции, сборки и запуска приложения. Порядок установки среды разработки на домашнем компьютере.

Практика: Написание проекта программы. Объяснение порядка создания, компиляции и сборки проекта на языке Java, порядка запуска проекта на выполнение.

Тема 1.2. Примитивные типы данных. Арифметика. Операции отношения и логические операции.

Теория: Системы счисления. Понятия переменных и константы. Целочисленные типы данных. Как задать значение константы в десятичной, двоичной, восьмеричной, шестнадцатеричной системе счисления. Как указать, что константа относится к типу long. Вывод на печать данных целого типа. Ввод данных целого типа.

Практика: Написание простейших программ, объявляющих переменные целого типа, присваивающих им значения. Вывод этих значений на печать. Наблюдение за поведением компилятора, когда переменной присваивается заведомо некорректное значение или выходящее за пределы диапазона для данного типа.

Тема 1.3. Условные конструкции. Блоки

Теория: Область действия блоков. Конструкция if-else. Конструкция switch-case. Мотивировка использования конструкции как упрощение сложных ветвлений.

Практика: Написание собственного примера на использование операторов ветвления.

Тема 1.4. Интеративные конструкции. Массивы. Списки.

Теория: Цикл с предусловием while. Цикл с постусловием do-while. Синтаксис. Объяснение логики работы, пример использования. Операторы прерывания логики управления программой. Безусловные операторы перехода break, continue.

Практика: Написание собственного примера на использование операторов цикла и операторов безусловного перехода. Написание программы по обработке массива с выводом на экран полученного результата.

Тема 1.5. Методы (функции). Видимость переменных. Рекурсия.

Теория: Определение функции как логически самостоятельной именованной части программы. Список формальных аргументов, список фактических аргументов. Методы с типом void и методы с пустым списком аргументов.

Практика: Выполнение практического задания на определение вида рекурсии.

Модуль 2. Объектно-ориентированное программирование

Тема 2.1. Классы и объекты.

Теория: Понятия «класс», «объект». Описание протокола, полей, метод класса. Общее понятие парадигм ООП инкапсуляция, полиморфизм и наследование на примерах из жизни.

Практика: Написание примеров классов и соответствующих им примеры объектов, полей и методов.

Тема 2.2. Классы: конструкторы, деструкторы и статические методы. Начальные приемы тестирования и отладки.

Теория: Конструкторы и деструкторы в Java, их разновидность и использование. Перегрузка методов. Понятие доступа класса.

Практика: Продолжение разработки класса, описывающего рациональную дробь. Разработка примеров программ, демонстрация изученных приёмов тестирования.

Тема 2.3. Архитектура приложений Android. Активности. Интерфейс пользователя. Язык разметки XML.

Теория: Знакомство со средой разработки Android-приложений и их общей структурой. Порядок создания, компиляции, сборки и запуска в среде. Порядок установки IDE и эмулятора для разработки приложений под Android на домашнем компьютере.

Практика: Разбор кода простейшего Android-приложения, иллюстрирующего общую схему, его запуск. Разбор примера проектирования игры-квеста. Самостоятельное проектирование UML-диаграммы классов приложения согласно заданию.

Тема 2.4. Намерения

Теория: Наследование классов как создание новых классов на основе существующих. Синтаксическое описание наследования классов и реализации интерфейсов. Защищенные члены класса.

Практика: Разработка примера с описанием классов, наследованием, переопределением метода, доступами и т. д.

Тема 2.5. Практикум. Контрольное тестирование по темам 1 и 2 модулей

Практика: Выполнение электронного теста, который охватывает все темы 1 и 2 модулей и в большей части направлен на оценку практических знаний и навыков учеников, полученных в ходе изучения модуля.

Тема 2.6. Профориентационное мероприятие

Практика: Посещение учреждения среднего профессионального образования или предприятия реального сектора экономики.

Базовый уровень

Модуль 3. Основы программирования Android приложений

Тема 3.1. Ввод, вывод и исключения

Теория: Файловый ввод вывод и механизм обработки исключений в Java, понятие «исключение» как объект.

Практика: Реализация обработки исключений в заготовке согласно заданию.

Тема 3.2. Внутренние классы в обработке событий

Теория: Понятие внутреннего класса. Локальные и анонимные внутренние классы. Обработка событий пользовательского интерфейса. Использование анонимных классов для реализации обработчиков событий.

Практика: Разбор примера кода с обработчиками событий. Реализация обработчиков событий с использованием анонимных классов согласно заданию.

Тема 3.3. Параллелизм и синхронизация. Потоки

Теория: Понятия потока, назначения многопоточности и структуры многопоточной программы. Потоки как средство реализации параллелизма в рамках одного процесса. Процессы и потоки в Android.

Практика: Разбор примера программы, совершающей загрузку картинки из интернета и устанавливающей её на экран.

Тема 3.4. Двумерная графика в Android приложениях

Теория: Класс Canvas – обзор методов и полей класса. Двумерная анимация.

Практика: Разбор предоставленного кода игрового приложения «Крестики – нолики». Внесение изменений в код, пересборка проекта и просмотр влияния изменений на поведение приложения.

Тема 3.5. Разработка игровых приложений. Реализация графики на основе SurfaceView

Теория: Общие подходы для реализации игровых приложений. Класс SurfaceView. Последовательные этапы проектирования и реализации игрового приложения.

Практика: Разбор примера простейшей игры с анимацией.

Модуль 4. Алгоритмы и структуры данных

Тема 4.1. Массивы. Алгоритм двоичного поиска. Алгоритмы сортировки.

Теория: Изучение массивов на примере библиотечных классов Arrays. Знакомство с идеей, применением и реализацией алгоритма двоичного поиска. Разбор алгоритмов сортировки и подходы к их реализации.

Практика: Практическое занятие по библиотечному классу Arrays,

реализующему массивы. Разбор примера использования `ArrayList<integer>`.

Тема 4.2. Адаптеры в Android

Теория: Адаптеры. Применение адаптеров для обработки событий пользовательского интерфейса. Готовые адаптеры в Android: `SimpleAdapter`, `ArrayAdapter`. Абстрактный класс `BaseAdapter`.

Практика: Разбор примера кода с реализацией `ListView` через `ArrayAdapter`.

Тема 4.3. Ассоциативные массивы

Теория: Понятие «ассоциативный массив» и его реализация в Java.

Практика: Разобрать пример использования `TreeMap` и `HashMap`.

Тема 4.4. Реляционная модель данных. СУБД. Введение в SQL

Теория: Локальная СУБД `SQLite`. Знакомство с локальной СУБД `SQLite`.

Практика: Создание БД `SQLite` «Записная книжка» по спроектированной ранее структуре. Разбор всех изученных инструкций SQL, создание простейшего Android-приложения..

Тема 4.5 Контрольное тестирование по модулю

Практика: Выполнение электронного теста, который охватывает все темы 3 и 4 модулей и в большей части направлен на оценку практических знаний и навыков учеников, полученных в ходе изучения модуля.

Модуль 5. Проектная деятельность

Тема 5.1 Консультации по ИП

Практика: консультация по вопросам индивидуального проекта.

Тема 5.2 Итоговая защита ИП

Практика: презентация и защита итогового проекта.

4. Планируемые результаты

Предметные результаты:

- знание основ языка программирования Java и языка разметки XML;
- умение использовать объектно-ориентированный подход в проектировании и разработке программного обеспечения;
- знание основ архитектуры приложения под Android;
- умение пользоваться ПК для программирования технических устройств;
- умение использовать разные алгоритмы в приёмах программирования;
- умение читать готовый код и находить в нем ошибки.

Личностные результаты:

- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата;
- сформированы коммуникативные компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности;
- сформировано осознанное позитивное отношение к другому человеку, его мнению, результату его деятельности;
- сформированное ответственное отношение к учению, целеустремленность и организованность.

Метапредметные результаты:

- умение искать, извлекать и отбирать нужную информацию из открытых источников;
- умение формулировать и излагать мысли в чёткой логической последовательности, аргументировать и отстаивать своё мнение;
- усвоены правила индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

II. Комплекс организационно-педагогических условий реализации общеразвивающей программы

1. Календарный учебный график на 2024-2025 учебный год

Таблица 2

№ п/п	Основные характеристики образовательного процесса	
1.	Количество учебных недель	36
2.	Количество учебных дней	36
3.	Количество часов в неделю	2
4.	Количество часов на учебный год	72
5.	Недель в I полугодии	17
6.	Недель во II полугодии	19
7.	Начало занятий	16 сентября
8.	Выходные дни	31 декабря – 8 января
9.	Окончание учебного года	31 мая

2. Календарный учебный график на 2025-2026 учебный год

Таблица 2

№ п/п	Основные характеристики образовательного процесса	
1.	Количество учебных недель	36
2.	Количество учебных дней	36
3.	Количество часов в неделю	2
4.	Количество часов на учебный год	72
5.	Недель в I полугодии	17
6.	Недель во II полугодии	19
7.	Начало занятий	10 сентября
8.	Выходные дни	31 декабря – 8 января
9.	Окончание учебного года	31 мая

3. Условия реализации 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:

- компьютеры и ноутбуки на каждого обучающегося и преподавателя;
- интерактивная панель-1 шт.
- маркерная доска – 1 шт;
- Смарт-часы APPLE Watch Series5 44мм серый космос/черный;
- Умные часы Samsung Galaxy Watch Active2 SM-R830N алюминий 40мм черный;
- Планшет Apple iPad Pro 11 (2020) Wi-Fi 512 ГБ серый (MXDE2RU/A) и Стилус ApplePen;
- Ноутбук с предустановленной ОС и манипулятором типа мышь, тип2LenovoThinkPad P72;
- Смартфон Samsung Galaxy S8 SM-G950FD 64Gb;
- Смартфон APPLE iPhone 7Plus;
- Планшет Samsung Galaxy Tab S3 9.7" 32Gb LTE Black (SM-T825).

Расходные материалы:

- whiteboard маркеры;
- бумага писчая;
- шариковые ручки;
- permanent маркеры.

Информационное обеспечение:

- операционная система Windows;

– программное обеспечение Eclips, Android Studio, объединенные в локальную сеть;

– Yandex Browser;

– ПК для педагога, объединённый с функцией сервера.

Кадровое обеспечение:

Программа реализуется Шаниным М. М., педагогом дополнительного образования.

Реализовывать программу могут педагоги дополнительного образования, обладающие достаточными знаниями в области педагогики, психологии и методологии, знающие особенности обучения основам мобильной разработки.

3. Формы аттестации и оценочные материалы

Система отслеживания результатов, обучающихся выстроена следующим образом:

- входная диагностика (Приложение 1);
- промежуточный контроль по первому и второму модулю – стартовый уровень (Приложение 2);
- промежуточный контроль по третьему и четвертому модулю – базовый уровень (Приложение 3);
- защита итогового проекта (Приложение 4).

Система контроля знаний и умений обучающихся представляется в виде учёта результатов по итогам выполнения контрольных тестов по модулям. Метод педагогического наблюдения помогает отслеживать динамику развития обучающегося. В конце обучения подростки проходят защиту индивидуальных/групповых проектов.

Итоговый проект оценивается формируемой комиссией по 50-бальной шкале. Состав комиссии (не менее 3 человек): в обязательном порядке входит педагог; приветствуется привлечение ИТ-профессионалов, представителей высших и других учебных заведений, администрации учебной организации.

Компонентами оценки индивидуального проекта являются (по мере убывания значимости): качество ИП, отзыв руководителя проекта, уровень презентации и защиты проекта. Если проект выполнен группой обучающихся, то при оценивании учитывается не только уровень исполнения проекта в целом, но и личный вклад каждого из авторов. Решение принимается коллегиально.

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

Минимальное количество баллов, которое возможно получить по результатам промежуточного контроля по первому и второму модулям - 1 балл, максимальное – 25 баллов.

Минимальное количество баллов, которое возможно получить по результатам промежуточного контроля по третьему и четвертому модулям - 1

балл, максимальное – 25 баллов.

Минимальное количество баллов, которое возможно получить по результатам защиты годового проекта - 1 балл, максимальное – 50 баллов.

Сумма баллов результатов промежуточного контроля и защиты итогового годового проекта переводится в один из уровней освоения образовательной программы согласно таблице 4:

Таблица 4

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

3.1. Перечень диагностического материала для осуществления мониторинга личностных и метапредметных планируемых результатов

1. Шкала поведенческих характеристик одаренных школьников (Приложение №5);

3. Мониторинг достижения обучающимися метапредметных результатов (Приложение №6);

5. Мониторинг достижения обучающимися личностных результатов (Приложение №7).

Методические материалы

Образовательный процесс осуществляется в очной форме с применением электронного обучения и дистанционных образовательных технологий.

В образовательном процессе используются следующие **методы**:

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

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 детей.

Методы воспитания: мотивация, убеждение, поощрение, упражнение, стимулирование, создание ситуации успеха и др.

Формы организации образовательного процесса: индивидуальная; групповая.

Формы проведения занятия:

В образовательном процессе помимо традиционного учебного занятия используются многообразные формы, которые несут учебную нагрузку и могут использоваться как активные способы освоения детьми образовательной программы, в соответствии с возрастом обучающихся, составом группы, содержанием учебного модуля: беседа, лекция, кейс, практическое занятие, защита проектов, тестирование.

Педагогические технологии: индивидуализации обучения; группового обучения; коллективного взаимообучения; дифференцированного обучения;

разноуровневого обучения; проблемного обучения; развивающего обучения; дистанционного обучения; игровой деятельности; коммуникативная технология обучения; коллективной творческой деятельности; решения изобретательских задач; здоровьесберегающая технология.

Здоровьесберегающая деятельность реализуется:

- через создание безопасных материально-технических условий;
- через включение в занятие динамических пауз, периодической смены деятельности обучающихся;
- через контроль педагога за соблюдением обучающимися правил работы за ПК;
- через создание благоприятного психологического климата в учебной группе в целом.

Дидактические материалы:

Методические пособия, разработанные преподавателем с учётом конкретных задач, варианты демонстрационных программ, материалы по терминологии ПО, учебная литература.

Пример вводного тестирования*(максимальное количество баллов – 20)*

Дата _____

ФИО _____ Группа _____

1) Набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения называется: (2 балла)

- а) Android SDK
- б) JDK
- в) плагин ADT
- г) Android NDK

2) С какой целью был создан Open Handset Alliance? (2 балла)

- а) писать историю развития ОС Android
- б) продавать смартфоны под управлением Android
- в) рекламировать смартфоны под управлением Android
- г) разрабатывать открытые стандарты для мобильных устройств

3) С какой целью инструмент Intel* Graphics Performance Analyzers (Intel* GPA) System Analyzer используется в среде разработки Intel* Beacon Mountain? (2 балла)

- а) позволить разработчикам оптимизировать загрузенность системы при использовании процедур OpenGL
- б) для ускорения работы эмулятора в среде разработки
- в) для оптимизированной обработки данных и изображений
- г) позволить разработчикам эффективно распараллелить C++ мобильные приложения

4) Какой движок баз данных используется в ОС Android? (2 балла)

- а) InnoDB
- б) DBM
- в) MyISAM
- г) SQLite

5) С какой целью инструмент Intel* Integrated Performance Primitives (Intel* IPP) используется в среде разработки Intel* Beacon Mountain? (2 балла)

- а) для оптимизированной обработки данных и изображений
- б) позволить разработчикам оптимизировать загруженность системы при использовании процедур OpenGL
- в) для ускорения работы эмулятора в среде разработки
- г) позволить разработчикам эффективно распараллелить C++ мобильные приложения

6) Intel XDK поддерживает разработку под: (2 балла)

- а) JavaFX Mobile
- б) Apple iOS, BlackBerry OS
- в) MtkOS, Symbian OS, Microsoft Windows 8
- г) Android, Apple iOS, Microsoft Windows 8, Tizen

7) Каждый приемник широковещательных сообщений является наследником класса ... (2 балла)

- а) ViewReceiver
- б) IntentReceiver
- в) ContentProvider
- г) BroadcastReceiver

8) Какой класс является основным строительным блоком для компонентов пользовательского интерфейса (UI), определяет прямоугольную область экрана и отвечает за прорисовку и обработку событий? (2 балла)

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

9) Какой слушатель используется для отслеживания события касания экрана устройства? (2 балла)

- a) OnPressListener
- б) OnTouchListener
- в) OnClickListener
- г) OnInputListener

10) Фоновые приложения ... (2 балла)

- а) после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии
- б) выполняют свои функции и когда видимы на экране, и когда скрыты другими приложениями
- в) небольшие приложения, отображаемые в виде графического объекта на рабочем столе
- г) большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки

Промежуточный контроль по первому и второму модулю*(максимальное количество баллов – 25)***Модуль 1. Основы программирования**

1. Типы данных Java. Фрагменты, которые не вызовут сообщение компилятора об ошибке преобразования типа: *(1 балл)*

- `int x = 21; double y = x;`
- `short z = 13; int x = z;`
- `long x = 15; int y = x;`
- `int x = 14; short z = x;`

2. Операции и выражения в Java. Выражения, результат которых имеет тип float или double: *(1 балл)*

- `5.0 * 3`
- `5 / 2`
- `5 * 0.5`
- `2.5 * 2`

3. Условные операторы в Java. Аналогом фрагмента кода `if (a == 2 || b == 3) { f(); }` является: *(1 балл)*

- `if (a == 2) { f(); }`
`else if (b == 3) { f(); }`
- `if (a == 2 && b == 3) { f(); }`
- `if (a != 2 && b != 3) { f(); }`
- `if (a == 2) if (b == 3) { f(); }`
- `if (a != 2) { }`
`else if (b != 3) { f(); }`

4. Циклы в Java. Значение переменной *a* после окончания выполнения цикла равно: *(1 балл)*

- `int a = 10;`
`for (int i = 1; i < 3; ++i) a *= 4;`

5. Массивы в Java. Правильное обращение к среднему элементу массива *a* нечетной длины приведено в: *(1 балл)*

- `a[a.length / 2]`

- `a[a.length / 2 + 1]`
- `a[a.length % 2]`
- `a.middle`
- `a[a.length/2-0.5]`

6. Видимость переменных. Фрагмент кода, который вызовет ошибку компиляции: (2 балл)

```
static int f() {
    int a = 3, b = 4;
    return a + b;}
    • static int a = 7;
```

```
static int f() {
    int b = 4;
    return a + b;}
    • static int a = 7;
```

```
static int f(int a) {
    int b = 4;
    return a + b;}
    • static int a = 7;
```

```
static int f(int a, int b) {
    int b = 4;
    return a + b;}
    • static int a = 7;
```

```
int b = 4;
return a + b;}
    • static int f(int a) {
```

7. Передача параметров в методы Java. В результате работы приведенного ниже фрагмента кода будет выведено: (2 балла)

```
static void swap(int a, int b) {
    int tmp = a;
    b = a;
    a = tmp;
}
```

```

public static void main(String[] args) {
int a = 5;
int b = 10;
swap(a, b);
System.out.print(a + ", " + b);
}

```

8. Многомерные массивы. Пропущенный фрагмент функции, подсчитывающей сумму элементов, находящихся под побочной диагональю квадратной матрицы $n \times n$ (побочной диагональю матрицы называют линию, соединяющую левый нижний и правый верхний угол матрицы) это: (2 балла)

```

int SumUnderSecondaryDiagonal(int n, int[][] matrix) {
int sum = 0;
if (n < 2) return 0;
return sum;

```

<pre> for(int j=n-i-1;j<n;j++) sum=sum+matrix[i][j]; </pre>	<ul style="list-style-type: none"> ● for(int i=0;i<n;i++)
<pre> for(int j=i;j>0;j--) sum+=matrix[i][j]; </pre>	<ul style="list-style-type: none"> ● for(int i=n-1;i>0;i--)
<pre> for(int j=0;j<i;j++) sum=sum+matrix[i][j]; </pre>	<ul style="list-style-type: none"> ● for(int i=0;i<n;i++)
<pre> for(int j=n-i;j<n;j++) sum+=matrix[i][j]; </pre>	<ul style="list-style-type: none"> ● for(int i=n-1;i>0;i--)

Модуль 2. Объектно-ориентированное программирование.

1. Передача параметров строк. В результате работы приведенного ниже фрагмента кода будет выведено: (1 балл)

```
public class Example{
    static String str = "ABC";
    public static void changeStr(String s) { s = "abc"; }
    public static void main(String[] args) {
        System.out.print(str);
        changeStr(str);
        System.out.print(str);
    }
}
```

2. Про приемы тестирования и отладки. Виды тестирования, которые могут заменить отладчик с применением точек останова с условием: (1 балл)

- Модульное тестирование;
- Стресс тестирование;
- Логирование;
- Проверка утверждения (assert);
- Интеграционное тестирование.

3. Язык разметки XML. Верные утверждения: (1 балл)

- XML-документ всегда должен содержать ровно один корневой элемент;
- В XML-документе не могут присутствовать цифры;
- В XML-документе могут присутствовать только латинские символы;
- XML-документ может содержать произвольное количество корневых элементов;
- Ни один из вариантов.

4. Layout. Свойство, позволяющее назначить элементу важность его размера относительно других элементов в контейнере: (1 балл)

- layout_weight
- layout_width
- layout_height
- layout_gravity

- layout_wrap

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

- наследование;
- полиморфизм;
- декомпозиция программы;
- инкапсуляция.

6. Классы. Правильное имя публичного класса в файле "Point.java" -(1 балл)

- Point
- Point.java
- point.java
- PointClass
- должно начинаться со слова Point

7. Перегрузка методов. Инициализация данных класса. Ошибка, к которой приводит отсутствие инициализации локальной переменной: (1 балл)

- происходит ошибка компиляции;
- переменная инициализируется случайными данными;
- переменная инициализируется null;
- происходит инициация исключения.

8. Программы на конструкторы и перегрузку. (1 балл)

```
● class Point {  
    float x, y;  
    // Конструктор  
}  
public class Example {  
    public static void main(String[] args) {  
        Point x = new Point(4.0f, 5.0f);  
        System.out.print(x.x + x.y);  
    }  
}
```

9. Анализ программ. (1 балл)

```
public class MyDate {
    private int d; // День
    private int m; // Месяц
    private int y; // Год
    static int[] dayInMonth = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
    public MyDate(int d, int m, int y) {
        if (d < 1 || m < 1 || m > 12 || y < 1) {
            System.out.println("Wrong date!");    return;    }
        If (d > whatIs2(m, y)) {    System.out.println("Wrong date!"); return;    }
        this.d = d;
        this.m = m;
        this.y = y;
    }
    public static boolean isLeap(int y) {
        return (y % 400 == 0) || ((y % 4 == 0) && (y % 100 != 0));
    }

    public static int whatIs2(int m, int y) {
        if (m == 2 && isLeap(y))    return dayInMonth[m - 1] + 1;
        return dayInMonth[m - 1];
    }

    public boolean whatIs(MyDate dt) {
        if (this.y == dt.y) {
            if (this.m == dt.m) return this.d > dt.d;
            else    return this.m > dt.m;
        }
        else    return this.y > dt.y;
    }
}
```

10. Метод класса MyDate whatIs() возвращает ложь, если: (1 балл)

- дата в параметре метода позже либо равна дате объекта, вызвавшего метод;
- дата в параметре метода строго позже, чем дата объекта, вызвавшего метод;
- дата в параметре метода строго раньше, чем дата объекта, вызвавшего метод;
- дата в параметре метода раньше либо равна дате объекта, вызвавшего метод;
- ни один из вариантов.

11. Наследование. Количество различных методов (из определенных в приведенном коде), которое может быть вызвано в методе *main* у переменной *x* без преобразования ее типа равно: (1 балл)

```
class Base {
    public void f1() {}
    private void f2() {}
    private void f3() {}
}
class Subclass extends Base {
    public void f1() {}
    public void f4() {}
}
class EntryPoint {
    public static void main(String[] argv) {
        Subclass x = new Subclass();
    }
}
```

12. Наследование. Правильный вызов конструктора базового класса *Base* из конструктора его наследника приведен в: (1 балл)

- `super(args);`
- `new Base(args);`
- `Base(args);`
- `new super(args);`

- ни один из вариантов.

13. Полиморфизм. Методы переменной x, которые могут быть вызваны без преобразования типа переменной x: (2 балла)

```
class Base {  
    public int f() { ... }  
    public int g() { ... }  
}  
class Derived extends Base {  
    public int g() { ... }  
    public int h() { ... }  
}
```

Base x = new Derived();

- f;
- g;
- h;
- ни один из вариантов.

- дата в параметре метода позже либо равна дате объекта, вызвавшего метод;
- дата в параметре метода строго позже, чем дата объекта, вызвавшего метод;
- дата в параметре метода строго раньше, чем дата объекта, вызвавшего метод;
- дата в параметре метода раньше либо равна дате объекта, вызвавшего метод;
- ни один из вариантов.

11. Наследование. Количество различных методов (из определенных в приведенном коде), которое может быть вызвано в методе *main* у переменной *x* без преобразования ее типа равно: (1 балл)

```
class Base {
    public void f1() {}
    private void f2() {}
    private void f3() {}
}
class Subclass extends Base {
    public void f1() {}
    public void f4() {}
}
class EntryPoint {
    public static void main(String[] argv) {
        Subclass x = new Subclass();
    }
}
```

12. Наследование. Правильный вызов конструктора базового класса *Base* из конструктора его наследника приведен в: (1 балл)

- `super(args);`
- `new Base(args);`
- `Base(args);`
- `new super(args);`

Промежуточный контроль по третьему и четвертому модулю

(максимальное количество баллов – 25)

Модуль 3. Основы Android

1. OO проектирование. Какой из вариантов лучше описывает отношение "Человек (man) проживает (live) по адресу (address)"? *(1 балл)*

- class Man extends Address{ };
- class Man { private Live address; };
- class Man implements Address{ };
- class Man { private Address live; };
- class Address{ private Man live; }.

2. Создание Андроид проекта. Основным классом Android приложения, имеющего интерфейс пользователя, является: *(1 балл)*

- Activity;
- KeyEvent;
- View;
- Fragment;
- Canvas.

3. Ввод, вывод и исключения. Ключевое слово, показывающее, что в процессе работы метода могут произойти не обработанные исключения: *(1 балл)*

- try;
- catch;
- finally;
- throws.

4. Класс(ы), используемый(е) для чтения файла. *(1 балл)*

- FileInputStream;
- FileOutputStream;
- FileInputStream, FileOutputStream;
- ни один из перечисленных.

5. Внутренние и анонимные классы. Пропущенный фрагмент, после

вставки которого программа выведет на экран значение 50 – это: (2 балла)

```
public class OuterClass {  
    private int x = 0;  
    private class InnerClass {  
        InnerClass() {  
            OuterClass.this.x = 50; }  
    }  
    private InnerClass xy;  
    public OuterClass() {____}  
    public static void main(String[] args) {  
        OuterClass obj = new OuterClass();  
        System.out.println(obj.x);  
    }  
}
```

- xy = new InnerClass();
- x = 50;
- this.xy = new InnerClass();
- xy = InnerClass();
- xy = 50.

6. Слушатели. Необходимо, чтобы когда пользователь отпускал кнопку myButton (класс android.widget.Button), запускался метод doIt(). Возможные способы реализации... (1 балл)

- Назначить в XML layout атрибуту android:onClick значение “doIt”;
- Назначить кнопке слушателя onClickListener и вызвать doIt() из метода onClick();
- Назначить кнопке слушателя onTouchListener и вызвать doIt() из метода onTouch();
- Вызвать startActivity с параметром myButton из myButton.onClick();
- ни один из вариантов.

7. Параллелизм и синхронизация. Поток. Операции, которые не должны выполняться одновременно в разных потоках: (1 балл)

- Два чтения одной и той же переменной;
- Запись и чтение одной и той же переменной;
- Две записи в одну и ту же переменную;
- Изменение двух разных экземпляров одного и того же класса.

8. Context и Intent. Верные утверждения: (1 балл)

- Класс android.context.Context используется для доступа к глобальной информации об окружении Android приложения;
- Для передачи намерения в Activity можно использовать метод startActivity();
- Метод putExtra() предназначен для сохранения данных в Intent;
- Явным (explicit) называется Intent, который содержит флаг Intent.FLAG_EXPLICIT;
- Для передачи намерения в Activity можно использовать любой метод.

9. Анализ программ с Canvas. (1 балл)

```

paint.setStyle(Paint.Style.FILL);
paint.setColor(Color.GREEN);
canvas.drawRect(50, 80, 400, 280, paint);
paint.setColor(Color.YELLOW);
canvas.drawCircle(150, 150, 50, paint);

```

10. Приведенный фрагмент рисует: (1 балл)

- желтый круг в верхнем левом углу зеленого прямоугольника;
- желтый круг в верхнем правом углу зеленого прямоугольника;
- желтый круг в нижнем правом углу зеленого прямоугольника;
- зеленый круг в верхнем левом углу желтого прямоугольника;
- ни один из вариантов.

Модуль 4. Алгоритмы и структуры данных

1. В поле `arr` переменной `a` после выполнения операции: `a = new A()`; будет храниться ... (1 балл)

- `null`;
- массив из 0 элементов;
- не определено;
- массив случайной длины из случайных чисел.

2. Списки (общие вопросы, теория). В двусвязном списке необходимо поменять два соседних элемента местами. Количество ссылок, которое будет изменено (если ни один из элементов не является ни первым ни последним) - (1 балл)

3. Списки: `LinkedList`, `Queue`, `Stack`. Операция взятия с удалением элемента из `Queue`: (1 балл)

- `push`;
- `pop`;
- `poll`;
- `peek`.

4. Деревья и двоичный поиск (общие вопросы, теория). Двоичное сбалансированное дерево обладает следующим свойством: (1 балл)

- Высота левого поддерева отличается от высоты правого поддерева не более, чем на 1;
- Все значения в вершинах левого поддерева не больше значений в вершинах правого поддерева;
- Все значения в вершинах левого поддерева больше значения в корне поддерева;
- Высота левого поддерева не меньше высоты правого поддерева.

5. Деревья: `TreeSet`. В результате работы ниже приведенного фрагмента на экран будет выведено: (1 балл)

```
TreeSet<Integer> tree = new TreeSet<Integer>();  
tree.add(15);
```

```
tree.add(25);
tree.add(15);
tree.add(11);
Iterator<Integer> iterator = tree.iterator();
while (iterator.hasNext()) System.out.print(iterator.next() + " ");
```

6. Рекурсия. Для данной функции количество рекурсивных вызовов (т. е. вызовов функции из самой себя) при $n=5$ равно... (1 балл)

```
int fib(int n) {
    if (n < 3) { return 1;}
    return fib(n - 1) + fib(n - 2);
}
```

7. Алгоритм двоичного поиска (класс Arrays и Collections).
Arrays.binarySearch(new int[]{-1, 3, 5, 5, 19}, -1) возвратит: (1 балл)

- 0;
- 1;
- -1;
- true;
- false.

8. Алгоритмы сортировки (класс Arrays). При вызове метода Arrays.sort(a); переменная a может быть следующего типа: (1 балл)

- int[];
- String[];
- ArrayList<Integer>;
- String;
- TreeMap<Integer, Integer>.

9. Хэш-таблица и функция хэширования (практика). Заданная программа выведет на экран число: (1 балл)

```
class Point {
int x, y;
public Point(int x, int y) {    this.x = x;    this.y = y; }
```

```

@Override
public int hashCode() {
    final int prime = 31;
    return prime + prime * x + prime * prime * y;
}
}public class Test {
public static void main(String[] args) {
    HashSet<Point> set = new HashSet<Point>();
    set.add(new Point(12, 13));
    set.add(new Point(9, 0));
    set.add(new Point(12, 13));
    set.add(new Point(9, 30));
    System.out.println(set.size());
}    }

```

10. Ассоциативные массивы: HashMap, TreeMap. Порядок, в котором появляются элементы при итерации по классу HashMap: (1 балл)

- зависит от реализации библиотеки;
- по неубыванию ключей;
- по невозрастанию ключей;
- по неубыванию значений;
- по невозрастанию значений.

11. Анализ программ. (1 балл)

```

class WhoAmI {
public int whatAmIDoing(int x) {
    set.add(x);
    return set.headSet(x).size();
}
}
public static long howDoIProcessThisArray(int[] a) {
    WhoAmI obj = new WhoAmI();
    long res = 0;

```

```

for (int i = a.length - 1; i >= 0; --i) {
    res += obj.whatAmIDoing(a[i]);
}
return res;
}
private SortedSet<Integer> set = new TreeSet<Integer>();
}

```

12. Вызов `WhoAmI.howDoIProcessThisArray(new int[]{2, 4, 7, 2, 5, 3, 1})` возвратит? (1 балл)

13. Реляционная модель, СУБД – понятия. Для отдельно взятой таблицы в реляционной модели баз данных верно утверждение: (1 балл)

- Каждая строка может иметь собственное количество столбцов;
- В одном столбце в разных строках могут быть записаны элементы разного типа;
- Все строки имеют одинаковое количество столбцов и в каждом столбце хранятся данные одного типа;
- Каждая строка содержит произвольное количество однотипных данных.

14. CREATE 2. Выполнен следующий запрос: (1 балл)

```

CREATE TABLE pupil (
    name varchar(30),      surname varchar(30),
    age integer,          birthday date,
    height float,         weight float ).

```

Количество колонок, которые будут иметь строковый тип в созданной таблице - ?

Запросы CREATE. Выберите таблицу, содержание которой соответствует запросу: `CREATE TABLE pupil (name varchar(3), age integer)?`

name	age
------	-----

Ivan	13
Wu	10

name	age
Hu	13
Wu	10

name	age
Ha	13.5
Wu	10

name	age
Ivan	twelve
Wu	eleven

Бланк оценки итоговых проектов

ФИО члена комиссии

Дата

№ п/п	ФИ автора (ов)	Название проекта	Время защиты	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 Правильность написание программы (0-10 б)	Критерий 5 Защита проекта (представление работы) (0-10 б)	Итого

_____ / _____
подпись / расшифровка

ШКАЛА ПОВЕДЕНЧЕСКИХ ХАРАКТЕРИСТИК ОДАРЕННЫХ ШКОЛЬНИКОВ

Авторы: Дж. Рензулли и соавторы, в адаптации Л.В. Поповой

Возраст детей: 14-17 лет

Цель: Эта шкала составлена для того, чтобы педагог мог оценить характеристики, обучающихся в познавательной, мотивационной, творческой и лидерской областях. Каждый пункт шкалы следует оценивать безотносительно к другим пунктам. Ваша оценка должна отражать, насколько часто вы наблюдали у обучающегося проявление каждой из характеристик. Так как четыре шкалы представляют относительно разные стороны поведения, оценки по разным шкалам не суммируются.

Инструкция: пожалуйста, внимательно прочитайте каждое утверждение и обведите соответствующую цифру согласно следующему описанию:

1 – если вы почти никогда не наблюдали этой характеристики;

2 – если вы наблюдаете эту характеристику время от времени;

3 – если вы наблюдаете эту характеристику довольно часто;

4 – если вы наблюдаете эту характеристику почти все время.

Каждый пункт шкалы следует оценивать безотносительно к другим пунктам. Ваша оценка должна отражать, насколько часто Вы наблюдали проявление каждой из характеристик. Так как четыре шкалы представляют относительно разные стороны поведения, оценки по разным шкалам не суммируются. Далее следуйте инструкции в таблице, чтобы узнать результаты.

Пример расчета результатов:

№	Утверждение	Выберите цифру			
		1	2	3	4
1.	Обладает необычно большим для этого возраста запасом слов; использует термины с пониманием; речь отличается богатством выражений, беглостью, сложностью.	1	②	3	4
2.	Обладает большим запасом информации по разнообразным темам (выходящим за пределы обычных интересов детей этого возраста).	1	2	③	4
3.	Быстро запоминает и воспроизводит фактическую информацию.	1	2	3	④

4.	Легко схватывает причинно-следственные связи; пытается понять «как» и «почему»; задает много стимулирующих мысль вопросов (в отличие от вопросов, направленных на получение фактов); хочет знать, что лежит в основе явлений или действий людей.	1	2	3	4
5.	Чуткий и сметливый наблюдатель; обычно "видит больше" или "извлекает больше", чем другие, из рассказа, фильма, из того, что происходит	1	2	3	4
Шаг 1. Подсчитайте число обведенных цифр по каждой колонке:		-	4	6	4
Шаг 2. Полученные значения умножьте на соответствующий коэффициент:		x1	x2	x3	x4
Шаг 3. Сложите полученные числа:		-	8	18	16
Шаг 4. Общий показатель равен:		42			

! После каждой шкалы есть интерпретация выраженности характеристик у обучающегося.

ШКАЛА I. ПОЗНАВАТЕЛЬНЫЕ ХАРАКТЕРИСТИКА УЧЕНИКА.

№	Утверждение	Выберите цифру			
		1	2	3	4
1.	Обладает необычно большим для этого возраста запасом слов; использует термины с пониманием; речь отличается богатством выражений, беглостью, сложностью.	1	2	3	4
2.	Обладает большим запасом информации по разнообразным темам (выходящим за пределы обычных интересов детей этого возраста).	1	2	3	4
3.	Быстро запоминает и воспроизводит фактическую информацию.	1	2	3	4
4.	Легко схватывает причинно-следственные связи; пытается понять «как» и «почему»; задает много стимулирующих мысль вопросов (в отличие от вопросов, направленных на получение фактов); хочет знать, что лежит в основе явлений или действий людей.	1	2	3	4
5.	Чуткий и сметливый наблюдатель; обычно "видит больше" или "извлекает больше", чем другие, из рассказа, фильма, из того, что происходит.	1	2	3	4
Шаг 1. Подсчитайте число обведенных цифр по каждой колонке:					
Шаг 2. Полученные значения умножьте на соответствующий коэффициент:		x1	x2	x3	x4
Шаг 3. Сложите полученные числа:					
Шаг 4. Общий показатель равен:					

Интерпретация Шкалы I:

0 - 10 баллов – низкий показатель

11 – 20 баллов – пониженный показатель

21 – 40 баллов – средний показатель

41-65 баллов – повышенный показатель

66-80 баллов – высокий показатель

ШКАЛА II. МОТИВАЦИОННЫЕ ХАРАКТЕРИСТИКИ

№	Утверждение	Выберите цифру			
		1	2	3	4
1.	Полностью уходит в определенные темы, проблемы; настойчиво стремиться к завершению начатого (трудно привлечь к другой теме, заданию).	1	2	3	4
2.	Легко впадает в скуку от обычных заданий.	1	2	3	4
3.	Стремиться к совершенству; отличается самокритичностью.	1	2	3	4
4.	Предпочитает работать самостоятельно; требует лишь минимального направления от педагога.	1	2	3	4
5.	Имеет склонность организовывать людей, предметы, ситуацию.	1	2	3	4
Шаг 1. Подсчитайте число обведенных цифр по каждой колонке:					
Шаг 2. Полученные значения умножьте на соответствующий коэффициент:		x1	x2	x3	x4
Шаг 3. Сложите полученные числа:					
Шаг 4. Общий показатель равен:					

Интерпретация Шкалы II:

0 - 10 баллов – низкий показатель

11 – 20 баллов – пониженный показатель

21 – 40 баллов – средний показатель

41-65 баллов – повышенный показатель

66-80 баллов – высокий показатель

ШКАЛА III. ЛИДЕРСКИЕ ХАРАКТЕРИСТИКИ.

№	Утверждение	Выберите цифру			
		1	2	3	4
1.	Проявляет ответственность; делает то, что обещает и обычно делает хорошо.	1	2	3	4
2.	Уверенно чувствует себя как с ровесниками, так и со взрослыми; хорошо себя чувствует, когда его просят показать свою работу группе.	1	2	3	4
3.	Ясно выражает свои мысли и чувства; хорошо и обычно понятно говорит.	1	2	3	4
4.	Любит находиться с людьми, общителен и предпочитает не оставаться в одиночестве.	1	2	3	4
5.	Имеет склонность доминировать среди других; как правило, руководит деятельностью, в которой участвует.	1	2	3	4
Шаг 1. Подсчитайте число обведенных цифр по каждой колонке:					
Шаг 2. Полученные значения умножьте на соответствующий коэффициент:		x1	x2	x3	x4
Шаг 3. Сложите полученные числа:					
Шаг 4. Общий показатель равен:					

Интерпретация Шкалы III:

- 0 - 10 баллов – низкий показатель
- 11 – 20 баллов – пониженный показатель
- 21 – 40 баллов – средний показатель
- 41-65 баллов – повышенный показатель
- 66-80 баллов – высокий показатель

ШКАЛА IV. ТВОРЧЕСКИЕ ХАРАКТЕРИСТИКИ.

№	Утверждение	Выберите цифру			
		1	2	3	4
1.	Проявляет большую любознательность в отношении многого; постоянно задает обо всем вопросы.	1	2	3	4
2.	Выдвигает большое количество идей или решений проблем и ответов на вопросы; предлагает необычные, оригинальные, умные ответы.	1	2	3	4
3.	Выражает свое мнение без колебаний; иногда радикален и горяч в дискуссиях, настойчив.	1	2	3	4
4.	Любит рисковать, имеет склонность к приключениям.	1	2	3	4

5.	Склонность к игре с идеями; фантазирует, придумывает («Интересно, что будет, если...»), занят приспособлением, улучшением и изменением общественных институтов, предметов и систем.	1	2	3	4
6.	Проявляет тонкое чувство юмора и видит юмор в таких ситуациях, которые не кажутся смешными остальным.	1	2	3	4
7.	Необычно чувствителен к внутренним импульсам и более открыт к иррациональному в себе (более свободное выражение «девчоночьих» интересов у мальчиков, большая независимость у девочек); эмоциональная чувствительность.	1	2	3	4
8.	Чувствителен к прекрасному; обращает внимание на эстетические стороны жизни.	1	2	3	4
9.	Не подвержен влиянию группы; приемлет беспорядок; не интересуется деталями; не боится быть отличным от других.	1	2	3	4
10	Дает конструктивную критику; не склонен принимать авторитеты без критического изучения.	1	2	3	4
Шаг 1. Подсчитайте число обведенных цифр по каждой колонке:					
Шаг 2. Полученные значения умножьте на соответствующий коэффициент:		x1	x2	x3	x4
Шаг 3. Сложите полученные числа:					
Шаг 4. Общий показатель равен:					

Интерпретация Шкалы IV:

0 - 30 баллов – низкий показатель

31 – 45 баллов – пониженный показатель

46 – 65 баллов – средний показатель

76 - 100 баллов – повышенный показатель

101 - 160 баллов – высокий показатель

**Мониторинг достижения обучающимися личностных результатов
за 20__-20__ учебный год**

№ п/п	Ф.И. обучающегося	возраст	самостоятельное планирование пути решения поставленной проблемы для получения эффективного результата			сформированность коммуникативных компетентностей в общении и сотрудничестве со сверстниками в образовательном процессе			сформированность осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности			сформированное ответственное отношение к учению, целеустремленность и организованность		
			входящий	промежуточный	итоговый	входящий	промежуточный	итоговый	входящий	промежуточный	итоговый	входящий	промежуточный	итоговый
1														
2														
3														
4														
5														
6														
7														
8														
9														
10														

Значение показателя по группе:

- 3 балла – качество проявляется систематически
- 2 балла – качество проявляется ситуативно
- 1 балл – качество не проявляется

- 1–1.7 – низкий уровень развития качества в группе
- 1.8–2.5 – средний уровень развития качества в группе
- 2.6–3 – высокий уровень развития качества в группе

**Мониторинг достижения обучающимися метапредметных результатов
за 20__-20__ учебный год**

№ п/п	Ф.И. обучающегося	возраст	умение искать, извлекать и отбирать нужную информацию из открытых источников			умение формулировать и излагать мысли в чёткой логической последовательности, аргументировать и отстаивать своё мнение			усвоены правила индивидуального и коллективного безопасного поведения при работе с компьютерной техникой		
			входящий	промежуточный	итоговый	входящий	промежуточный	итоговый	входящий	промежуточный	итоговый
1											
2											
3											
4											
5											
6											
7											

3 балла – качество проявляется систематически
2 балла – качество проявляется ситуативно
1 балл – качество не проявляется

Значение показателя по группе:
1–1.7 – низкий уровень развития качества в группе
1.8–2.5 – средний уровень развития качества в группе
2.6–3 – высокий уровень развития качества в группе

Список литературы

Нормативные документы:

1. Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 N 273-ФЗ.
2. Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребёнка в РФ».
3. Распоряжение Правительства РФ от 31.03.2022 №678-р «Об утверждении Концепции развития дополнительного образования детей до 2030 года».
4. Распоряжение Правительства РФ от 29.05.2015 N 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года».
5. Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
6. Приказ Министерства Просвещения Российской Федерации от 03.09.2019 г. №467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей».
7. Приказ Министерства труда и социальной защиты Российской Федерации от 05.05.2018 г. № 298н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»».
8. 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».
9. Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242 «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)».
10. Методические рекомендациями для субъектов Российской Федерации по вопросам реализации основных и дополнительных общеобразовательных программ в сетевой форме, утвержденных Министерством просвещения России 28.06.2019г №МР-81/02вн.

Список литературы, использованной при написании программы:

1. Блох Джошуа. Java. Эффективное программирование. Effective Java. Programming Language Guide. изд. «Лори». 2014. – 310 с. ISBN 978-5-85582-347-9.

2. Гослинг Джеймс, Билл Джой, Гай Л. Стил, Гилад Брача, Алекс Бакли. Язык программирования Java SE 8. Подробное описание, 5-е изд.: Пер. с англ. – Москва: ООО «И.Д. Вильямс», 2015. – 672с.

3. Медникс З., Дорнин Л., Мик Б., Накамура М. Программирование под Android. 2-е изд. – Санкт-Петербург, 2013. – 560 с.

4. Майер, Р. Android 2. Программирование приложений для планшетных компьютеров и смартфонов. Пер. с англ. – Москва, 2011. – 672 с.

5. Филлипс Б., Стюарт К., Марсикано К., Гарднер Б. Android. Программирование для профессионалов. 4-е издание. — Санкт-Петербург: Питер, 2021–704 с.

6. Эльконин Д.Б. Детская психология: учеб. пособие для студ. высш. учеб. заведений / Д. Б. Эльконин; ред.-сост. Б. Д. Эльконин. — 4-е изд., стер. — Москва: Издательский центр «Академия», 2007. — 384 с

Электронные ресурсы:

1. Code Basics: обучение базовым аспектам языков программирования от образовательной платформы Hexlet. // [Электронный ресурс] URL: <https://ru.code-basics.com/> (дата обращения: 20.04.2021);

2. Портал обучения «Информатикс» // [Электронный ресурс] URL: <https://informatics.msk.ru> (дата обращения: 20.04.2021);

3. Официальный сайт для разработчиков приложений для Android // [Электронный ресурс] URL: <https://developer.android.com> (дата обращения: 20.04.2021).

Список литературы для обучающихся:

1. Вязовов С.М., Калягина О.Ю., Слезин К.А., «Соревновательная робототехника: приемы программирования в среде EV3», М.: Издательство «Перо» 2015 – 132 с.
2. Детская энциклопедия «Открытия и изобретения».- М, РОСМЭН, 2011– 48 с.
3. Зайцева Н. Н., Цуканова Е. А.. «Конструируем роботов на LEGO MINDSTORMS Education EV3. Человек – всему мера?» М.: Лаборатория знаний, 2016. – 32 с.
4. Михеев П.М., Крылов С.И., Лукьянченко В.А., Урюпина Д.С., «Учебный курс. LabVIEW Основы I», М.: Издательство Московского государственного университета им. М.В. Ломоносова. 2015. – 29 с.
5. Филиппов С.А., «Уроки робототехники. Конструкция. Движение. Управление.», М.: Лаборатория знаний, 2017. – 176 с.

Аннотация

Задача инновационного развития программного обеспечения требует соответствующей образовательной среды, в том числе создания оптимальных условий детского технического творчества. Одной из наиболее инновационных областей в сфере детского технического творчества является мобильная разработка.

Для дальнейшего развития мобильных приложений существует широкий выбор направлений разработки. Каждому ребёнку интересно, как устроена Java платформа, как работает Java приложение на любой платформе и на смартфоне в том числе.

Изучение языка программирования Java по данной программе обучения даёт возможность пользователю освоить базовые навыки использования языка программирования, понять его особенности использования и выполнения на различных платформах.

Разработка мобильных приложений на базе Android на сегодняшний день очень востребована ввиду высокой популярности данной ОС. Поэтому обучение по данной программе – это самый первый, но важный шаг в изучении основ программирования на языке Java, для создания проектов и простейших программ в среде разработки на его основе.

Программа рассчитана на обучающихся 14–17 лет.