

П. В. Кушняров

**ИССЛЕДУЕМ, ПРОЕКТИРУЕМ, СОЗДАЁМ**

# ИНФОРМАТИКА

Образовательные проекты

6 – 9 классы

**КОМПЛЕКТЫ ЗАДАНИЙ**

Пособие для учащихся учреждений образования, реализующих образовательные программы общего среднего образования, с белорусским и русским языками обучения и воспитания

*Рекомендовано государственным учреждением образования  
«Академия образования»*

Минск  
2026

## Оглавление

|                                                                                       |    |
|---------------------------------------------------------------------------------------|----|
| 1. БЕЗОПАСНЫЙ ИНТЕРНЕТ: ПРАВИЛА И СОВЕТЫ ДЛЯ УЧАЩИХСЯ .....                           | 3  |
| 2. ВИРТУАЛЬНЫЙ ПИТОМЕЦ: ОБУЧАЮЩЕЕ ИНТЕРАКТИВНОЕ ПРИЛОЖЕНИЕ НА SCRATCH .....           | 8  |
| 3. МОЯ ПЕРВАЯ КОМПЬЮТЕРНАЯ ИГРА .....                                                 | 13 |
| 4. ИНТЕРАКТИВНАЯ ТАБЛИЦА МЕНДЕЛЕЕВА В СРЕДЕ SCRATCH .....                             | 18 |
| 5. УМНЫЙ ПОМОЩНИК: КАК РАБОТАЕТ РОБОТ-ПЫЛЕСОС .....                                   | 22 |
| 6. СОЗДАНИЕ ЦИФРОВОГО ОПРЕДЕЛИТЕЛЯ ЗВУКОВ ЖИВОТНЫХ .....                              | 27 |
| 7. БЕЛАРУСКІ АРНАМЕНТ У ЛІЧБАВАЙ ГРАФІЦЫ.....                                         | 32 |
| 8. ИНФОГРАФИКА: ПРЕДСТАВЛЕНИЕ ДАННЫХ В ВИЗУАЛЬНОЙ ФОРМЕ .....                         | 36 |
| 9. ЗНАКОМСТВО С 3D-РЕДАКТОРОМ: МОДЕЛИРОВАНИЕ МАТЕМАТИЧЕСКИХ ФИГУР.....                | 41 |
| 10.МОДЕЛИРОВАНИЕ ДВИЖЕНИЯ ТЕЛА ПО ЗАКОНАМ ФИЗИКИ..                                    | 45 |
| 11.ХИМИЯ В ЧИСЛАХ: ИЗУЧЕНИЕ СВОЙСТВ ЭЛЕМЕНТОВ ЧЕРЕЗ ВИЗУАЛЬНОЕ ПРОГРАММИРОВАНИЕ ..... | 49 |

# БЕЗОПАСНЫЙ ИНТЕРНЕТ: ПРАВИЛА И СОВЕТЫ ДЛЯ УЧАЩИХСЯ

Интернет — это как огромный город. В нём есть полезные места: библиотеки (сайты для учёбы), парки развлечений (игры и видео), и места для общения (соцсети). Но, как и в любом городе, здесь есть свои правила и даже опасные районы. Часто мы заходим в сеть, не зная этих правил, и можем попасть в неприятности.

## **Цель проекта:**

Разработка интерактивной памятки/презентации «Безопасный интернет» для сверстников.

При выполнении проекта вы:

изучите основные виды интернет-угроз, с которыми могут столкнуться учащиеся;

проведете мини-опрос среди одноклассников по теме «Как мы используем Интернет»;

научитесь анализировать информацию из различных источников;

разработаете полезные советы и рекомендации;

оформите материалы в виде презентации и памятки.

## **Оборудование и технические материалы:**

Компьютеры с выходом в Интернет;

Программное обеспечение: текстовый редактор (MS Word), программа для создания презентаций (PowerPoint);

## **Ход выполнения проекта**

### ***Этап 1. Погружение в проблему***

Учитель расскажет о реальных случаях, когда учащиеся попадали в неприятности из-за незнания правил безопасности: кто-то скачал вирус, потеряв все фото; кому-то написали грубые сообщения; кого-то обманули, выманив деньги.

Ответь честно себе на вопросы:

Было ли с тобой что-то подобное? Может, тебе присылали странную ссылку или писал незнакомец?

Знаешь ли ты, что такое «фишинг» или «кибербуллинг»? Это не ругательства, а конкретные угрозы.

Кому бы ты рассказал, если бы столкнулся с проблемой в сети?

Наша миссия — создать «щит и меч» для каждого ученика: интерактивную памятку и презентацию «Безопасный интернет», которые научат распознавать угрозы и защищаться.

Как выполнить (твои шаги):

Заведи «Досье интернет-угроз». Это твоя рабочая тетрадь.

Начни первичный поиск. Вместе с учителем найди и запиши в досье 3-4 самых частых опасности.

Например:

Фишинг: Когда тебя обманом выманивают пароль, будто бы от имени «администрации ВКонтакте» или «банка».

Кибербуллинг: Травля, оскорбления, угрозы в чатах или соцсетях.

Вредоносные программы (вирусы): Программы, которые крадут твои данные или ломают компьютер.

Нежелательный контент: Сайты, где тебе может стать страшно или неприятно.

### ***Этап 2. Распределение ролей***

Объединись в отряд из 3-4 учащихся. Выбери свою специализацию в отряде:

Агент-исследователь (Разведка): Твоя зона — глубина интернета. Ты ищешь самую актуальную и проверенную информацию об угрозах на сайтах, подобных «Лиге безопасного интернета». Ты добываешь «сырые данные».

Агент-аналитик (Штаб): Твой инструмент — логика. Ты обрабатываешь результаты опросов, ищешь закономерности, превращаешь данные исследователя в понятные тезисы и выводы. Ты делаешь информацию готовой к использованию.

Агент-дизайнер (Техническая поддержка): Ты — креативный гений. Твоя задача — сделать так, чтобы никому не было скучно. Ты оформляешь памятку и презентацию в Canva или PowerPoint: подбираешь шрифты, цвета, иконки, расставляешь акценты. Ты создаёшь «лицо» проекта.

Создайте «План операции». Распишите по неделям:

Неделя 1. Провести опрос, собрать информацию.

Неделя 2. Проанализировать данные, написать тексты советов.

Неделя 3. Сверстать памятку и сделать презентацию.

Неделя 4. Отрепетировать защиту.

### ***Этап 3. Поэтапное выполнение проекта***

Это самый важный и большой этап. Выполняем его шаг за шагом.

Шаг 3.1: Проведение опроса — узнаём правду

Создайте анонимный опрос в Google Forms или на бумаге. Используйте вопросы из учебника (см. в конце). Важно: обещайте одноклассникам, что результаты будут общими и без имен!

Проведите опрос. Вежливо попросите одноклассников уделить 5 минут.

«Агент-аналитик», начинай! Собери все ответы. Посчитай, сколько человек сколько времени проводит в сети? Сколько уже сталкивались с проблемами? Самые частые ответы — это твоя золотая информация.

Шаг 3.2: Сбор и анализ информации — изучаем врага

«Агент-исследователь», в бой! Используй только проверенные источники:

Сайт «Детский правовой сайт» <https://mir.pravo.by/edu/internet-i-rebenok>.

Составь для каждой угрозы короткую памятку: 1) Как выглядит? 2) Чего хочет злоумышленник? 3) Что делать?

Шаг 3.3: Подготовка текстов и материалов — создаём оружие

Объедините усилия всей командой. На основе анализа опроса и исследований создайте:

Список из 5-7 железных правил. Они должны быть короткими и ясными, как команды. Например: «Правило 1: Пароль — как зубная щётка. Не давай никому и меняй раз в три месяца». Алгоритм действий в случае беды. Что делать, если тебя травят? Если выманивают пароль? План: «Стоп. Не отвечай. Сохрани скриншот. Расскажи взрослому».

Список тех, кто всегда поможет: родители, школьный психолог, учитель информатики, телефон доверия.

Шаг 3.4: Создание презентации и памятки — упаковываем знания

Для памятки:

Возьми шаблон.

Используй крупные заголовки, иконки, короткие абзацы. Не делай «много текста». Лучше 5 коротких советов с картинкой, чем 20 длинных. Убедись, что памятку интересно читать и всё понятно с первого взгляда.

Для презентации (6-8 слайдов):

Самый важный слайд — № 5 (Результаты опроса). Вставь туда настоящую диаграмму из Google Forms! Это покажет, что ваша работа основана на фактах из вашего же класса. Используй минимум текста на слайде. Текст — то, что ты говоришь. На слайде — только главная мысль и картинка.

**Этап 4. Презентация и оформление результатов, защита проекта**

Продумайте сценарий защиты. Это не доклад, а выступление. Начало (30 сек): «Представьте, что завтра вам в личку пишет незнакомец с мутной ссылкой. Вы знаете, что делать? Мы узнали и сейчас научим вас!» Середина (3-4 мин): Покажите ключевой слайд с результатами опроса: «Оказалось, 40%

нашего класса уже сталкивались с грубостью в сети. Это много! Поэтому смотрите на самые важные правила...». Задайте вопрос классу: «Как вы думаете, какой пароль надёжнее: '12345' или 'K0т\_В\_Тапк@х'?». Финал (1 мин): Раздайте или покажите на экране вашу памятку. Скажите: «Мы сделали эту шпаргалку для вас. Сохраните её на телефон или повесьте над столом. Берегите себя в сети!»

Распределите речь:

Исследователь: «Мы узнали, что самая частая угроза для нашего возраста — это...»

Аналитик: «Наш опрос показал, что большинство ребят в опасной ситуации...»

Дизайнер: «Мы оформили это вот так, чтобы было понятно и запомнилось...»

Говорите громко, чётко, смотрите на одноклассников, а не в экран.

### **Ожидаемые результаты**

В результате выполнения проекта вы освоите принципы безопасной работы в интернете, выработаете будущие стратегии поиска необходимой информации, научитесь работать в команде, совместно планировать и выполнять работу, представлять ее результаты. Кроме этого, вы разработаете памятку по правилам безопасного поведения в интернете.

**Приложение.** Твои инструменты для проекта

*Пример опросника:*

Опрос «Твой цифровой портрет»:

1. Сколько времени ты обычно проводишь в интернете после школы?

Меньше часа (я активный оффлайнер)

1-2 часа (для дел и развлечений)

3-4 часа (уже привычка)

Больше 4 часов (сеть — мой второй дом)

2. Зачем ты чаще всего заходишь в сеть? (можно несколько)

Играть

Смотреть видео (YouTube, TikTok)

Учиться (искать информацию, на «Скайсмарт»)

Общаться (ВКонтакте, Telegram, Discord)

Просто листать ленту

3. Случалось ли, что в интернете тебе было неприятно или страшно?

Нет, никогда

Да, мне писали грубости или угрожали (кибербуллинг)

Да, меня пытались обмануть, выманить данные (фишинг)

Да, я попадал(а) на странные или пугающие сайты

Другое: \_\_\_\_\_

4. Если случится проблема в сети, к кому ты пойдёшь в первую очередь?

К родителям

К лучшему другу/подруге

К классному руководителю или психологу

Разберусь сам(а)

Ни к кому, постесняюсь

**ПЯТЬ СУПЕРПРАВИЛ:**

Пароль: Делай длинным (12+ символов) и уникальным для каждого сайта.

Данные: Домашний адрес, номер школы, фото документов — ТОЛЬКО ДЛЯ СЕБЯ.

Незнакомцы: В сети — те же правила, что на улице. Не доверяй, не встречайся.

Скриншот — твой друг. При травле — сохрани доказательства (скрин, ссылку).

Думай перед публикацией: Выложил — не удалишь. Интернет помнит всё!

# ВИРТУАЛЬНЫЙ ПИТОМЕЦ: ОБУЧАЮЩЕЕ ИНТЕРАКТИВНОЕ ПРИЛОЖЕНИЕ НА SCRATCH

Ты играл в игры про виртуальных питомцев, где нужно кормить, играть и заботиться о симпатичном существе? А что, если ты сможешь создать такую игру САМ, и твой питомец будет уникальным?

Этот проект — твой шанс стать разработчиком собственного цифрового приложения. Ты не просто научишься собирать блоки в Scratch, ты поймёшь, как устроена логика любой игры: как персонаж "оживает", как кнопки управляют его состоянием, как всё взаимосвязано. Ты создашь не просто анимацию, а целый живой мир со своими правилами.

## **Цель проекта:**

Создание игры «Виртуальный питомец» с возможностью кормить, поить, играть и наблюдать за реакциями виртуального персонажа.

При выполнении проекта вы:

- познакомитесь с интерфейсом и основными блоками Scratch;
- разработаете алгоритм поведения виртуального питомца;
- создадите спрайты, фоны, анимации;
- реализуете логику взаимодействия (кнопки, счётчики, состояния);
- протестируете работу программы и устранить ошибки;
- оформите и презентовать готовое приложение.

## **Оборудование и необходимые материалы:**

ПК с установленным Scratch 3.0 или доступом к <https://scratch.mit.edu>;  
графический планшет/мышь (по желанию);  
микрофон и наушники (для записи/воспроизведения звуков — по необходимости).

## **Ход выполнения проекта**

### ***Этап 1. Погружение в проблему. Рождение идеи***

Посмотрим на известные игры с питомцами (тамагочи, Talking Tom). Обсудим, что в них цепляет: милая анимация, смешные реакции, чувство ответственности?

Задаём главный вопрос: *«Как сделать так, чтобы на экране жило существо, которое реально реагирует на мои действия?»*

Наша цель — создать в Scratch интерактивное приложение "Виртуальный питомец" с 3-4 действиями (кормить, играть, укладывать спать) и видимыми изменениями его состояния.

Как выполнить (твои шаги):



Заведи "Блокнот геймдизайнера". Это твой главный документ. Опиши своего будущего питомца. Это не обязательно кот или собака! Пусть это будет:

*Дракоша-сладкоежка*

*Инопланетянин-непоседа*

*Привидение-хохотун*

*Робот-спортсмен*

Определи, как он будет "жить". Запиши в блокнот:

Какие у него будут потребности? (Голод, Настроение, Энергия).

Как он будет показывать, что ему плохо? (Потухшие глаза, грустный вид, зелёный от голода).

Как он будет радоваться? (Подпрыгнет, улыбнётся, мигает).

### ***Этап . Организация деятельности. Создаём студию разработки***

Объединись в команду мечты (3-4 человека). Распределите роли, как в настоящей игровой студии:

Главный художник (Графический дизайнер): Король спрайтов и фонов! Ты рисуешь (или собираешь из библиотеки Scratch) все визуальные элементы: самого питомца в разных состояниях, кнопки, красивый задний план.

Главный логик (Алгоритмист): Мозг проекта. Твоя задача — продумать и запрограммировать, как всё работает. Ты создашь переменные "Голод", "Настроение" и свяжешь кнопки с реакциями питомца.

Главный по атмосфере (Звукорежиссёр): Ты отвечаешь за звуковое сопровождение. Подберёшь или запишешь звуки: "ням-ням" при кормлении, веселую музыку для игры, храп для сна.

Главный продюсер (Презентатор и тестер): Ты видишь проект целиком. Ты протестируешь игру, найдёшь баги, подготовишь крутую презентацию и будешь ярко представлять ваше творение миру.

Создайте "Дизайн-документ". В общем файле опишите:

Имя и вид питомца.

Список действий игрока (3-4 кнопки).

Список состояний питомца (3-4 эмоции/состояния).

Результат этапа:

У вас есть укомплектованная студия, дизайн-документ и распределённые зоны ответственности.

### ***Этап 3. Проектирование алгоритма (таблица)***

Проведите мозговой штурм. Соберите всю команду и ответьте на вопросы:

Что будет, если питомца не кормить? Переменная Голод уменьшается. Если Голод = 0, питомец грустит и вяло двигается.

Что будет, если с ним постоянно играть? Переменная Энергия уменьшается. Если Энергия = 0, питомец засыпает прямо во время игры.

Как параметры влияют друг на друга?

(Например, если Голод низкий, то и Настроение падает).

Создайте "Паспорт параметров". Таблица — твой лучший друг.

*Таблица. Проектирование алгоритма*

| Параметр<br>(Переменная) | Мин. | Макс. | Как увеличивается                                | Как уменьшается                   | Если MIN                   |
|--------------------------|------|-------|--------------------------------------------------|-----------------------------------|----------------------------|
| Голод                    | 0    | 10    | Кнопка "Покормить" (+3)                          | Со временем (-1 каждые 30 сек)    | Грустный вид, нет движения |
| Настроение               | 0    | 10    | Кнопка "Поиграть" (+2)                           | Если Голод < 3 (-1 каждую минуту) | Плачет, отворачивается     |
| Энергия                  | 0    | 10    | Кнопка "Уложить спать" (восстанавливается до 10) | Кнопка "Поиграть" (-2 за игру)    | Засыпает на месте          |

Нарисуйте блок-схему для одной кнопки. Например, для кнопки "Покормить": Начало -> Проверить, если Голод < 10 -> Да: увеличить Голод на 3 -> Показать анимацию "ест" -> Проверить, если Голод > 7 -> Да: увеличить Настроение на 1 -> Конец.

#### ***Этап 4. Реализация проекта. Магия в Scratch!***

Часть А: Графика и интерфейс (работает Художник и Продюсер)

Создайте спрайты:

Спрайт "Питомец": Сделай для него несколько костюмов (costumes): "нормальный", "радостный", "грустный", "спит", "ест".

Спрайты-кнопки: "Кормить", "Играть", "Спать". Их можно нарисовать или взять из библиотеки.

Создайте красивый фон (stage background). Может быть, уютная комната или фантастический пейзаж.

Часть Б: Программирование логики (работает Логик и вся команда)

Создайте переменные. В меню "Переменные" сделай Голод, Настроение, Энергия. Поставь галочку, чтобы они были видны на экране.

Напишите скрипт для ЗЕЛЁНОГО ФЛАГА:

когда щёлкнут по флагу  
установить [Голод v] в (8)  
установить [Настроение v] в (8)  
установить [Энергия v] в (8)  
перейти на костюм [нормальный v]  
повторять всегда  
ждать (30) секунд  
изменить [Голод v] на (-1)  
конец  
(Этот код задаст начальные значения и будет медленно уменьшать голод).

Запрограммируйте кнопку "Покормить":

когда спрайт [Кнопка\_Еда v] нажат  
если  $\langle \text{Голод} \rangle < [10] >$  , то  
изменить [Голод v] на (3)  
перейти на костюм [ест v]  
играть звук [ням-ням v]  
ждать (1) секунд  
перейти на костюм [нормальный v]  
конец

Добавьте реакцию питомца на низкие показатели. Создай отдельный скрипт, который постоянно проверяет состояние:

когда щёлкнут по флагу  
повторять всегда  
если  $\langle \text{Голод} \rangle < [3] >$  , то  
перейти на костюм [грустный v]  
end  
если  $\langle \text{Настроение} \rangle > [7] >$  , то  
перейти на костюм [радостный v]  
end  
конец

Часть В: Тестирование и полировка (работает вся команда)

Нажимай на зелёный флаг и ТЕСТИРУЙ!

Корми питомца 10 раз подряд. Сломается ли что-то? (Переменная Голод не должна становиться больше 10 — добавь проверку!). Играй, пока энергия не кончится. Уснул ли он? Ищи баги и улучшай: Добавь больше звуков, сделай плавные переходы между костюмами, придумай бонусы.

### ***Этап 5. Подготовка и защита проекта. Премьера вашей игры!***

Создайте короткую, но эффектную презентацию (3-4 слайда в Canva или прямо в Scratch добавьте титульный экран):

Слайд 1: Афиша. Крупное название игры и ваш крутой питомец. Девиз: "Мы создали жизнь в цифре!"

Слайд 2: Механика за 30 секунд. Покажите вашу таблицу параметров и объясните её просто: "Вот три шкалы жизни нашего дракоши. Всё, как у нас: нужно есть, веселиться и отдыхать".

Слайд 3: Живая демонстрация. Самое главное. не рассказывайте, а запустите ваш проект в SCRATCH на большом экране и продемонстрируйте его вживую. покормите питомца, поиграйте с ним.

Слайд 4: Команда и вызов. Фото/имена команды. И финальный вопрос классу: "А какого питомца создали бы ВЫ?".

Распределите речь:

Художник: "Мы хотели, чтобы наш питомец был милым и выразительным, поэтому мы нарисовали ему 5 разных эмоций..."

Логик: "Самым сложным было заставить параметры влиять друг на друга. Мы использовали блок 'если-то' чтобы..."

Продюсер: "В итоге у нас получилась стабильная игра. Мы протестировали 10 сценариев, и главный баг, который мы нашли, был..."

Будьте готовы к вопросам: "А что будет, если не укладывать его спать?", "А можно добавить мини-игру?".

#### **Ожидаемые результаты**

Выполняя проект, вы изучите основы визуального программирования в среде Scratch, научитесь работать в команде, выдвигать и обсуждать идеи, отстаивать свою точку зрения, представлять результаты совместной работы. В результате у вас получится готовое интерактивное приложение – виртуальный домашний питомец.

# МОЯ ПЕРВАЯ КОМПЬЮТЕРНАЯ ИГРА

Ты проводишь часы, играя в мобильные игры на телефоне? А что, если ты сможешь не просто играть, а создать свою собственную игру?

Этот проект — твой первый серьёзный шаг. Ты не просто изучишь теорию, а сразу начнёшь творить. Ты станешь дизайнером, сценаристом и программистом в одном лице. К концу проекта у тебя на телефоне будет работать твоя собственная игра, которой ты сможешь похвастаться перед друзьями.

Ты пройдёшь полный путь разработчика: от пустого экрана до работающего игрового приложения. Ты освоишь профессиональный, но дружелюбный инструмент — MIT App Inventor, научишься оживлять персонажей и поймёшь, как устроена логика любой игры. Готов стать творцом?

## **Цель проекта:**

Разработка собственной простой мобильной игры с помощью среды визуального программирования App Inventor.

При выполнении проекта вы:

познакомитесь с интерфейсом App Inventor и основами блочного программирования;

определите идею игры (жанр, цель, правила);

разработаете дизайн и интерфейс приложения;

запрограммируете поведение объектов в игре;

протестируете и внесете улучшения в игру;

оформите результаты проекта и представите игру классу.

## **Оборудование и необходимые материалы**

Компьютер/ноутбук с доступом в Интернет;

Среда программирования: MIT App Inventor  
(<https://appinventor.mit.edu/>);

Устройство Android для тестирования игры (по возможности);

Планшет/телефон, USB-кабель, приложение MIT AI2 Companion

## **Ход выполнения**

### ***Этап 1. Погружение в проблему. От игрока — к создателю***

Анализируем игры как профессионалы: Разберём несколько простых игр (например, «Змейка», «Поймай яблоко», «Бесконечный раннер»). Зададимся вопросами: Как игра понимает, что я нажал на экран? Как объекты двигаются? Что такое «очки» и «жизни» с точки зрения кода? Поймём, что любая игра — это набор чётких правил (алгоритмов), прописанных на особом языке.

Наша цель — создать работающее мобильное приложение-игру в MIT App Inventor, упаковать его в APK-файл и запустить на своём телефоне.

Заведи «Геймдизайнерский блокнот». Это твой главный документ. Выбери жанр для своей первой игры.

Начни с простого, но эффектного:

«Избегай препятствий» (Космический корабль уворачивается от астероидов).

«Поймай предмет» (Корзинка ловит падающие фрукты).

«Викторина» (Игра с вопросами и вариантами ответов).

«Лабиринт» (Шар, который нужно провести к выходу, наклоняя телефон).

Опиши игру в трёх пунктах: Запиши в блокноте:

Главный герой / объект управления: Кто/что находится на экране? (Корабль, корзинка, шарик).

Цель игры: Что нужно сделать? (Набрать 50 очков, пройти уровень, ответить на 10 вопросов).

Правила: Что приводит к победе, а что — к поражению? (Коснулся астероида — игра окончена. Поймал яблоко — +10 очков).

## ***Этап 2. Организация деятельности. Освоение инструментария***

Зарегистрируйся и познакомься с MIT App Inventor. Перейди на сайт [appinventor.mit.edu](http://appinventor.mit.edu). Войди через свой Google-аккаунт. Создай новый проект (Start new project). Дай ему имя (только английские буквы, без пробелов, например, MyFirstSpaceGame). Изучи интерфейс.

Перед тобой две основные вкладки:

Designer (Дизайнер): Здесь ты будешь расставлять кнопки, картинки, надписи на экран. Это «внешность» твоей игры.

Blocks (Блоки): Здесь ты будешь собирать логику из разноцветных блоков, как из Лего. Это «мозг» твоей игры.

Создай раскадровку (storyboard). Нарисуй на листе бумаги или в графическом редакторе:

Экран 1: Заставка. Большое название, кнопка «Старт».

Экран 2: Игровое поле. Где будет герой? Где будут появляться препятствия/предметы? Где будут отображаться очки?

Экран 3: Конец игры. Надпись «Игра окончена» и счёт.

Создай список задач (To-Do List). Например:

Добавить фоновую картинку.

Загрузить спрайт для корабля.

Сделать так, чтобы корабль двигался за пальцем.

Настроить появление астероидов сверху.

Запрограммировать столкновение.

Добавить звуки и счётчик очков.

### ***Этап 3. Поэтапное выполнение проекта. Магия создания игры***

Это самый важный и интересный этап. Мы разобьём его на ключевые шаги.

#### **Шаг 3.1: Создание интерфейса в Designer**

Работаем во вкладке Designer.

Добавь компоненты с палитры:

Из раздела User Interface возьми Label (надпись) для названия игры и счётчика очков. Из раздела Drawing and Animation возьми Canvas (холст). Это самое главное! Растяни его на весь экран. На этом холсте будет происходить вся анимация. На этот же Canvas добавь два компонента ImageSprite (спрайт с картинкой): один для героя, второй для предмета/препятствия. Из Media добавь Sound (звук) для эффектов. Настрой свойства каждого компонента в правой колонке:

Для Label со счётчиком: в свойстве Text напиши Очки: 0.

Для ImageSprite героя: в свойстве Picture загрузи свою картинку (можно нарисовать в Paint или найти в интернете с прозрачным фоном PNG).

Для Canvas: задай цвет фона или загрузи фоновую картинку.

#### **Шаг 3.2: Программирование логики в Blocks**

Переключись во вкладку Blocks. Здесь начинается настоящее волшебство.

Основной принцип: Всё строится на событиях и действиях. Событие — это триггер (например, «когда холст Canvas1 тронут пальцем»). Действие — что происходит в ответ (например, «переместить спрайт Герой в точку касания»).

Ключевые скрипты для игры «Избегай препятствий»:

Управление героем:

blocks

[Когда] Canvas1.Touched [x] [y]

[Выполнить] Hero.MoveTo [x] [y]

(Этот код заставит спрайт Hero двигаться к точке, где ты коснулся экрана).

Движение астероидов:

blocks

[Когда] Clock.Timer [таймер сработал]

[Выполнить] Asteroid.ChangeY [на] [5] # Астероид летит вниз

[Если] Asteroid.TouchingEdge # Если достиг края

[То] Asteroid.MoveTo [случайное X] [-50] # Возвращаем наверх

Столкновение и счётчик:

blocks

[Когда] Hero.CollidedWith [другим спрайтом Asteroid]

[Выполнить] Sound1.Play # Проигрываем звук взрыва

[Выполнить] Notifier1.ShowAlert [сообщение "Игра окончена!"]

[Выполнить] Clock.TimerEnabled [в] [ложь] # Останавливаем игру

Начисление очков (для игры «Поймай предмет»):

blocks

[Когда] Basket.CollidedWith [другим спрайтом Apple]

[Выполнить] Global Score [установить в] [Score] + [10]

[Выполнить] LabelScore.Text [в] [объединить "Очки: " [Score]]

#### ***Этап 4. Тестирование и итеративное улучшение***

Протестируй игру вживую!

Установи на свой Android-смартфон приложение MIT AI2 Companion из Google Play. В App Inventor нажми Connect -> AI Companion. Отсканируй QR-код с телефона. Твоя игра запустится прямо на телефоне в реальном времени! Все изменения в коде будут сразу отображаться. Играй и ищи баги: Что-то двигается не так? Столкновение не срабатывает? Очки не начисляются? Возвращайся в Blocks и ищи ошибку в логике. Добавь крутые детали: Звуки, смену скорости препятствий, увеличение счёта за комбо, несколько жизней, кнопку «Рестарт».

#### ***Этап 5. Презентация результатов. Защита проекта***

Собери свой APK-файл (как настоящий разработчик):

В меню App Inventor выбери Build -> App (save .apk to my computer).

Это финальный, установочный файл твоей игры! Его можно отправить другу, и он установит игру себе на телефон без всяких компаньонов.

Подготовь презентацию-демонстрацию (4 слайда в PowerPoint или просто речь):

Часть 1: Идея. Покажи свою первоначальную раскадровку. Объясни жанр и цель игры.

Часть 2: Демонстрация — ГЛАВНОЕ! ПОДКЛЮЧИ СВОЙ ТЕЛЕФОН К ПРОЕКТОРУ или запусти эмулятор на большом экране. СЫГРАЙ В СВОЮ ИГРУ НА ГЛАЗАХ У КЛАССА. Покажи, как всё работает.

Часть 3: За кулисами. Покажи 1-2 самых важных или сложных скрипта из вкладки Blocks. Объясни: «Вот этот блок отвечает за движение, а этот — за столкновение».

Часть 4: Вывод и инсайты. Расскажи, что было самым трудным (отладка? поиск картинок?), а что — самым крутым (момент, когда игра впервые запустилась на телефоне).



Будь готов к вопросам: «А как добавить новый уровень?», «А можно сделать, чтобы препятствия двигались быстрее со временем?».

Результат этапа: Эффектная живая демонстрация, файл APK в портфолио и статус самого крутого разработчика в классе.

### **Ожидаемые результаты**

Рабочий файл проекта в MIT App Inventor (.aia файл) или установочный APK-файл.

Готовая игра, которую можно запустить на Android-устройстве.

Краткая устная презентация о процессе создания игры.

Понимание основных этапов разработки программного приложения.

# ИНТЕРАКТИВНАЯ ТАБЛИЦА МЕНДЕЛЕЕВА В СРЕДЕ SCRATCH

Представь, что таблица Менделеева — это не скучная таблица на стене, а живая, интерактивная карта химической вселенной. Каждый элемент — это не просто символ в клетке, а персонаж со своей историей, свойствами и суперспособностями. А что, если ты сможешь создать такую карту сам, где по клику на элемент он оживёт и расскажет о себе всё самое интересное?

## **Цель проекта:**

Создание интерактивной таблицы Менделеева в Scratch, содержащей сведения о химических элементах и позволяющей в игровой форме изучать их свойства.

При выполнении проекта вы:

выберете формат представления информации об элементах (в виде карточек, меню, викторины и т.д.);

создадите спрайты для элементов (группы или отдельные клетки);

реализуете алгоритм отображения информации при клике на элемент (например: название, символ, атомный номер, группа, использование);

добавите визуальные и звуковые эффекты;

оформите и презентуете проект.

## **Оборудование и необходимые материалы:**

компьютеры с доступом к Scratch 3.0 (установленному или онлайн);

таблица Менделеева (бумажная или цифровая);

шаблоны элементов в виде карточек (свойства, символы, атомный номер);

средства для презентации (PowerPoint, онлайн-сервисы).

## **Ход выполнения проекта**

*Этап 1. Погружение в проблему. От скучной таблицы к цифровому миру*

Обсудим, что сложного в запоминании элементов. Согласись, просто смотреть на клетки с буквами — не очень вдохновляет? Посмотрим на крутые цифровые приложения и игры, которые делают учёбу интересной. Как превратить статичную таблицу в интерактивное приложение, где каждый элемент "отвечает" на твой клик? Наша цель — создать в Scratch интерактивную модель таблицы Менделеева, где при клике на элемент появляется карточка с ключевой информацией о нём.

Как выполнить (твои шаги):

Заведи "Дневник химического дизайнера". Выбери "героев" своей таблицы: Не пытайся охватить все 118 элементов сразу! Выбери 10-15 самых интересных или важных:

Известные (Золото Au, Кислород O, Углерод C)

Опасные (Ртуть Hg, Уран U)

Технологические (Кремний Si для процессоров, Литий Li для батареек)

Придумай, как они будут "общаться": Запиши в дневнике: "Когда я кликаю на Железо (Fe), оно должно показать карточку, где написано: 'Я — основа стали! Меня много в метеоритах!'".

### ***Этап 2. Планирование и организация. Собираем научную экспедицию***

Собери команду первооткрывателей (3-5 человек). Распредели роли, как в настоящей научной лаборатории:

Химик-исследователь: Твой инструмент — учебник и научные сайты. Ты собираешь точные и интересные факты о каждом выбранном элементе: символ, номер, свойства, где применяется. Ты — хранитель истины!

Программист-конструктор: Ты — инженер, который строит логику. Твоя задача — сделать так, чтобы по клику на спрайт появлялась нужная информация. Ты работаешь с переменными, списками и событиями в Scratch.

Дизайнер-визуализатор: Ты — художник проекта. Ты создаёшь внешний вид: красивую сетку таблицы, стильные карточки для информации, иконки элементов. Ты делаешь так, чтобы всё выглядело круто и профессионально.

Редактор-сценарист: Ты — голос проекта. Ты пишешь короткие, ёмкие и понятные тексты для карточек, подбираешь звуки (например, "шипение" для щелочных металлов), готовишь текст для презентации.

Создайте "Техническое задание" (ТЗ) проекта. В общем документе укажите:

Список элементов (например, H, O, C, Fe, Au, Si, Na, Cl, Al, Cu).

Формат карточки: Что в ней будет? (1. Название, 2. Символ, 3. Номер, 4. Интересный факт).

Визуальный стиль: Какие цвета возьмём за основу?

### ***Этап 3. Реализация проекта. Строим химическую вселенную***

#### **Шаг 3.1: Проектирование интерфейса**

Вместе решите, как будет выглядеть ваша таблица в Scratch.

Вариант А (Простой и эффективный): Создаёте один спрайт в виде большой сетки-фона. На нём рисуете или размещаете отдельные спрайты-кнопки для каждого элемента.

Вариант Б (Более продвинутый): Создаёте множество одинаковых спрайтов-"клеточек", которые клонируются и размещаются в нужных позициях, образуя таблицу.

Дизайнер, нарисуй макет на бумаге: где будет сетка, где — место для всплывающей карточки.

### Шаг 3.2: Создание графики

Создай или найди в библиотеке Scratch спрайты для элементов. Это могут быть:

- Простые цветные квадраты/кружки с символом элемента внутри.
- Стилизованные иконки (например, кусочек золота для Au, батарейка для Li).

Создай спрайт "Информационная карточка". Это окно, которое будет появляться при клике. Оно должно быть чистым и читаемым. Подготовь красивый фон для сцены (Stage).

### Шаг 3.3: Программирование логики — самое интересное!

Создай списки (Lists) для хранения данных. Это твоя база данных! Создай списки:

Названия = ["Водород", "Кислород", "Золото"...]

Символы = ["H", "O", "Au"...]

Факты = ["Самый лёгкий газ!", "Нужен для дыхания", "Не ржавеет!"...]

Присвой каждому спрайту-элементу свой порядковый номер (например, Водород = 1, Кислород = 2). Это можно сделать через переменную мой\_номер.

Напиши для каждого спрайта-элемента скрипт:

*когда спрайт нажат*

*передать [показать\_инфо v] и ждать*

Создай скрипт для спрайта "Карточка", который ловит это сообщение:

*когда я получу [показать\_инфо v]*

*задать [номер\_элемента v] в (мой\_номер) // если номер передали вместе с сообщением*

*сказать (объединить [Элемент: ] (элемент (номер\_элемента) из [Названия v])) в течение (2) сек*

*сказать (объединить [Символ: ] (элемент (номер\_элемента) из [Символы v])) в течение (2) сек*

*сказать (элемент (номер\_элемента) из [Факты v]) в течение (3) сек*

Добавь визуальных элементов: Сделай так, чтобы при наведении курсора на элемент он немного увеличивался (эффект рыбий глаз) или менял цвет.

### Шаг 3.4: Тестирование и отладка

Проверьте ВСЕ элементы по чек-листу:

Кликается ли спрайт?

Появляется ли карточка?

Информация в карточке соответствует элементу? (Золото не должно рассказывать про батарейки!).

Нет ли опечаток?

Упростите, если что-то не работает: Если списки кажутся сложными, можно начать с простого: для каждого элемента сделать свой уникальный скрипт, который говорит его свойства.

#### ***Этап 4. Подготовка презентации. Научный доклад о вашем открытии***

Создайте презентацию-историю (4-5 слайдов):

Слайд 1: Проблема и решение. "Таблица Менделеева казалась скучной. Мы сделали её живой и интерактивной!"

Слайд 2: Наша методология. Покажите скриншот ваших списков в Scratch и объясните: "Мы создали цифровую базу данных элементов — вот её сердце".

Слайд 3: Живая демонстрация — ГЛАВНОЕ! ЗАПУСТИТЕ ВАШ ПРОЕКТ В SCRATCH НА БОЛЬШОМ ЭКРАНЕ. Продемонстрируйте, как вы кликаете на 3-4 разных элемента и получаете информацию. Сделайте это динамично!

Слайд 4: Чему мы научились. "Мы не только узнали про элементы, но и поняли, как устроены базы данных и интерактивные интерфейсы".

Распределите речь:

Химик: "Мы отбирали элементы так, чтобы показать разнообразие мира: от лёгкого Водорода до радиоактивного Урана..."

Программист: "Ключевой задачей было связать нажатие на кнопку с выводом правильного текста. Мы использовали систему списков и номеров..."

Дизайнер: "Мы выбрали такую цветовую схему, чтобы элементы разных групп визуально отличались..."

Редактор: "Наша задача — сделать факты короткими и запоминающимися. Например, про Натрий: 'Взрывается в воде, но есть в поваренной соли!'".

#### **Ожидаемые результаты**

Рабочий интерактивный проект в Scratch, представляющий собой фрагмент таблицы Менделеева с функцией просмотра свойств элементов.

Краткая устная презентация проекта.

Понимание связи между структурой данных (списки в программировании) и систематизацией научной информации.

# УМНЫЙ ПОМОЩНИК: КАК РАБОТАЕТ РОБОТ-ПЫЛЕСОС

Ты видел, как робот-пылесос сам объезжает всю комнату, не врезаясь в ножки стульев и возвращаясь на базу? Это кажется магией, но на самом деле внутри него нет волшебства — только точная наука, умные алгоритмы и хитрые датчики. А что, если ты сможешь заглянуть «в голову» такому роботу и понять, как он мыслит? Ещё лучше — сам создать его виртуальную копию и запрограммировать её логику?

## **Цель проекта:**

Разработка модели алгоритма работы робота-пылесоса в симуляторе или визуальной среде программирования с учетом анализа принципов его передвижения, избегания препятствий и очистки помещения.

При выполнении проекта вы:

изучите устройство и алгоритмы работы современных роботов-пылесосов;

изучите, как работают датчики (инфракрасные, ультразвуковые, гироскоп, GPS и др.);

познакомитесь с алгоритмами навигации: зигзагообразный маршрут, обход по периметру, работа по карте;

разработаете и смоделируете алгоритм передвижения робота-пылесоса в программной среде;

создадите презентацию проекта и защитите её.

## **Оборудование и необходимые материалы**

Компьютеры с доступом в Интернет;

Программное обеспечение для моделирования (например, Scratch, Tinkercad, LEGO Mindstorms или аналог);

Видео и фотоматериалов по работе роботов.

## **Ход выполнения проекта**

### ***Этап 1. Погружение в проблему. С чего начинается «ум»?***

Наблюдаем за работой реального робота (видео, демонстрация, если есть устройство). Вместо «О, как круто!» задаём вопросы: Как он понимает, где стена? Почему он едет зигзагом? Откуда знает, что уже тут убирал?

Разбираем мифы: Нет, у него нет глаз-камер как у человека (чаще всего). Нет, он не запоминает каждый сантиметр наизусть. Его «интеллект» основан на чём-то другом.

Наша цель — разработать и запрограммировать виртуальную модель робота-пылесоса в симуляторе (например, в Coppeliasim, Scratch или

специальной образовательной среде), которая сможет автономно очищать заданную площадку, избегая препятствий.

Как выполнить (твои шаги):

Заведи «Инженерный журнал исследований». Проведи «Мозговой штурм вопросов».

Запиши в журнал всё, что хочешь выяснить:

Какие датчики у него есть и зачем каждый?

Какой алгоритм движения самый эффективный для уборки?

Как он строит карту комнаты?

Что такое «возврат на базу» с точки зрения программирования?

Сформулируй гипотезу: «Я думаю, что для эффективной уборки робот должен сначала объехать комнату по периметру, а потом заполнить середину параллельными линиями».

## ***Этап 2. Организация деятельности. Формируем отдел разработки***

Собери команду инженеров (3-4 человека).

Распределите роли в проектной лаборатории:

Системный аналитик («Заказчик»): Ты ставишь задачи. Ты глубоко изучаешь, как устроены реальные роботы-пылесосы (типы датчиков, виды навигации), и формулируешь, как должна вести себя наша модель. Ты — главный по теории.

Архитектор алгоритмов («Мозговой центр»): Ты проектируешь логику. На основе данных аналитика ты придумываешь и рисуешь блок-схемы алгоритмов: движения, обхода препятствий, возврата на базу. Ты — главный по логике.

Программист-реализатор («Кодер»): Ты воплощаешь идеи в код. Твоя задача — выбрать среду программирования (например, Scratch для простой 2D-модели или Python для более сложной симуляции) и написать рабочий скрипт для виртуального робота.

Тестировщик и визуализатор («Инженер по качеству»): Ты отвечаешь за результат. Ты тестируешь модель в симуляторе, ищешь баги, предлагаешь улучшения, а также готовишь крутые визуализации и презентацию для защиты.

Создайте «Техническое задание (ТЗ) на виртуального робота». Миссия: Автономная уборка прямоугольной площадки с 3-5 статичными препятствиями. Требования к «сенсорам»: Модель должна «чувствовать» стены и препятствия (например, с помощью виртуальных лучей или касаний). Алгоритм: Реализовать один из изученных методов (например, движение по спирали или «змейка»). Критерий успеха: За отведённое время робот должен «очистить» (побывать на) не менее 80% свободной площади.

### ***Этап 3. Поэтапное выполнение проекта. От чертежа до виртуального железа***

Это самый сложный и интересный этап. Разделим его на логические модули.

Модуль 3.1: Теоретическое исследование. Изучаем «анатомию» робота (3-4 часа)

Изучите «органы чувств» робота: Инфракрасные (ИК) датчики и «бамперы»: Для обнаружения препятствий вплотную. Лидар (LIDAR): Лазерный «радар», который строит точную карту окружения. Гироскоп и одометрия: Чтобы понимать, насколько и куда он повернулся.

Разберитесь с алгоритмами навигации. Случайное движение. Просто, но неэффективно. Движение по спирали или зигзагу (лавинная уборка) систематично, но можно пропустить углы. Построение карты (SLAM): Сначала строит карту, потом убирает по ней — самое продвинутое. Создайте совместный документ с выводами. Кратко, с картинками: «Мы выяснили, что для нашей модели, где точная карта не нужна, достаточно алгоритма зигзага с датчиками касания».

Модуль 3.2: Проектирование алгоритма. Рисуем «инструкцию для мозга»

Нарисуйте блок-схему основного цикла работы.

Например:

НАЧАЛО -> Ехать вперёд -> Есть препятствие впереди? -> ДА: Повернуть на 90 градусов -> НЕТ: Продолжать движение -> Прошло N секунд? -> ДА: Возврат на базу -> КОНЕЦ.

Продумайте детали: Как именно поворачивать, чтобы не заикнуться в углу? Как имитировать «уборку» (например, менять цвет пройденной клетки)?

Модуль 3.3: Программирование и симуляция. Оживляем модель

Вариант А (Начальный уровень, Scratch):

Создайте спрайт «Робот» и несколько спрайтов-препятствий.

Используйте блоки идти, повернуть, а также блок касается цвета? или касается спрайта? для имитации датчиков.

Запрограммируйте движение по алгоритму из блок-схемы. Чтобы показать уборку, можно использовать блок штамп или менять цвет фона позади робота.

Вариант Б (Продвинутый уровень, Python + библиотека Pygame):

python

# Псевдокод-подсказка

while not mission\_complete:

    robot.move\_forward()



```
if sensor.detect_obstacle(): # Если датчик "увидел" препятствие
    robot.rotate(random.choice([90, -90])) # Повернуть направо или
налево
if battery_low() or area_cleaned(): # Если села батарея или всё убрано
    robot.return_to_home()
```

Не забывайте тестировать после каждого добавленного блока или функции! «Тестировщик» должен вести баг-репорт: «После поворота робот застревает в углу. Нужно добавить случайный угол поворота».

#### ***Этап 4. Тестирование, анализ и оптимизация***

Смотрите на работу модели. Она убирает всё или заикливается? Анализируйте эффективность: Засеките время, за которое робот убирает 80% площади. Можно ли улучшить алгоритм? Добавьте «фишки»: Звук при столкновении, индикатор оставшейся площади, красивую анимацию возврата на базу.

Результат этапа: Рабочая, протестированная виртуальная модель робота-пылесоса, записанное видео её работы и отчёт о проведённых тестах.

#### ***Этап 5. Подготовка презентации. Защита проекта «Инженерные решения»***

Создайте презентацию в стиле стартапа или научно-технического отчёта (5-7 слайдов):

Слайд 1: Проблема. «Миллионы людей используют роботов-пылесосов, не понимая, как они мыслят. Мы это исправили».

Слайд 2: Анализ и проектирование. Покажите свою лучшую блок-схему и кратко объясните, почему выбрали именно этот алгоритм и «сенсоры».

Слайд 3: Демонстрация — ГЛАВНЫЙ ЭКСПОНАТ! ВСТАВЬТЕ В ПРЕЗЕНТАЦИЮ ВИДЕОЗАПИСЬ РАБОТЫ ВАШЕЙ МОДЕЛИ. Пусть все увидят, как ваш виртуальный робот выполняет миссию.

Слайд 4: Данные и оптимизация. Покажите простой график или таблицу: «Первая версия убирала 60% за 5 минут. После оптимизации алгоритма поворота — 85% за те же 5 минут».

Слайд 5: Выводы и перспективы. «Мы доказали, что базовый алгоритм на простых сенсорах способен на эффективную уборку. Дальше можно добавить построение карты (SLAM)».

Распределите речь:

Аналитик: «Мы изучили рынок и выяснили, что основные типы навигации — это... Для нашей цели мы выбрали...».

Архитектор: «Логической основой нашей модели стал алгоритм, который... Его преимущество в...».

Программист: «Технически мы реализовали это через... Самая сложная часть кода была...».

Тестировщик: «В процессе мы столкнулись с проблемой... Решили её так... Эффективность модели подтверждена тестами...».

Будьте готовы к «вопросам инвесторов»: «А что, если препятствие появится внезапно?», «Как ваш алгоритм масштабируется на большую площадь?», «В чём ваше преимущество перед случайным движением?».

### **Ожидаемые результаты**

Рабочая виртуальная модель робота-пылесоса, запрограммированная на уборку заданной площади.

Презентация проекта, содержащая описание исследования, выбранного алгоритма и результаты работы модели.

Понимание связи между аппаратным обеспечением (датчики), программным алгоритмом и поведением робота.

# СОЗДАНИЕ ЦИФРОВОГО ОПРЕДЕЛИТЕЛЯ ЗВУКОВ ЖИВОТНЫХ

Окружающее пространство наполнено многообразием звуковых волн. Современные компьютерные технологии позволяют автоматизировать процесс их анализа и классификации. Данный учебный проект посвящён созданию прототипа системы автоматического распознавания видов животных по их звуковым проявлениям. Работа лежит на междисциплинарной границе биоакустики (раздела биологии, изучающего звуковое общение организмов) и компьютерных наук.

В процессе выполнения проекта вы освоите ключевые этапы обработки аудиосигналов: от первичного сбора и анализа звуковых записей до визуализации аудиоданных в виде спектрограмм, которые отображают частотные характеристики сигнала во времени.

Итогом работы станет функционирующий прототип программы-идентификатора, способный распознавать звуки различных животных. Приобретённые компетенции будут полезны как в области информатики (обработка сигналов, алгоритмизация, основы искусственного интеллекта), так и в биологии, предоставляя инструмент для объективного анализа акустического поведения.

## **Цель проекта:**

Создать прототип программы-определителя звуков животных, которая по загруженному аудиофайлу определяет принадлежность к виду (например: ворона, соловей, лягушка, сверчок, кошка, собака).

При выполнении проекта вы:

изучите цифровое представление звука и основные характеристики аудиофайлов;

познакомитесь с программами для анализа звука (Audacity, Sonic Visualiser) и библиотеками Python (librosa);

соберете небольшую базу аудиофайлов животных (5–6 видов);

выделите признаки звуков: диапазон частот, ритм, длительность;

разработаете упрощённый алгоритм классификации звуков по признакам;

создадите простой интерфейс программы («Загрузить файл → Определённое животное»);

подготовите презентацию с описанием работы программы и биологическими комментариями.

## **Оборудование и необходимые материалы**

Компьютеры с установленными программами для анализа звука: *Audacity*, *Sonic Visualiser* (для спектрограмм), для создания прототипа определителя: *Python* с библиотеками *librosa*, *Tkinter* или блокное программирование в *Scratch*. Колонки или наушники для прослушивания звуков.

### **Ход выполнения проекта**

#### ***Шаг 1. Погружение в проблему***

Живые организмы существуют в насыщенной звуками среде. Звуковая коммуникация является одним из фундаментальных способов взаимодействия животных. Биоакустика — научная дисциплина, изучающая эти процессы. Каждый вид в процессе эволюции выработал уникальные акустические "подписи" — комбинации частотных, временных и амплитудных характеристик, оптимальные для его экологической ниши и образа жизни.

Необходимо прослушаешь записи разных животных (птица, собака, кошка, лягушка, сверчок). Можно ли заставить компьютер автоматически понимать, кому принадлежит этот звук? Для этого потребуется проанализировать различные звуковые файлы и найти отличия между ними.

#### ***Шаг 2. Изучение основ звука***

Звук в компьютер — это цифры. Необходимо из предложенных звуковых файлов выделить основные характеристики. Частота (Гц): Определяет высоту звука. Высокие звуки (птицы) имеют высокую частоту, низкие (лягушки) — низкую. Амплитуда: Определяет громкость.

Длительность: Как долго длится звук. Спектрограмма: Это «фотография» звука, где по вертикали — частота, по горизонтали — время, а цветом показана громкость.

Животные издают звуки не просто так. Это их язык: для привлечения партнера, защиты территории или предупреждения об опасности.

Для более продуктивной работы необходимо разделить направления работы:

Биолог: Отвечает за изучение видов животных и их повадок.

Звукоинженер: Работает с аудиофайлами, строит спектрограммы.

Программист: Пишет код алгоритма.

Дизайнер/Докладчик: Готовит презентацию и интерфейс.

Необходимо открой программу *Audacity*. Загрузи любой звуковой файл и найди функцию «Спектрограмма». Посмотри, как выглядят знакомые звуки.

Попробуй определить на спектрограмме, у какого звука частота выше, а у какого ниже. Запиши длительность и ритм нескольких звуков.

#### ***Шаг 3. Сбор базы данных***

Используя набор аудиофайлов с подписями: ворона.wav, лягушка.wav и т.д. запустите Audacity или аналогичную программу, открой каждый файл и построй его спектрограмму.

Вместе с командой заполни общую таблицу.

*Таблица. Пример таблицы для заполнения*

| Животное | Частотный диапазон (Гц) | Ритм                        | Длительность |
|----------|-------------------------|-----------------------------|--------------|
| Ворона   | 400–2000                | одиочные крики              | 0,5–1 с      |
| Соловей  | 2000–6000               | мелодичные трели            | 1–3 с        |
| Лягушка  | 100–500                 | повторяющиеся «ква»         | 0,5–1 с      |
| Сверчок  | 2000–4000               | быстро повторяющиеся щелчки | 0,2–0,5 с    |
| Кошка    | 300–1000                | отдельные мяукания          | 0,5–1 с      |
| Собака   | 200–1500                | гавканье, ритмичное         | 0,5–2 с      |

#### ***Шаг 4. Анализ аудио и выделение признаков***

Сравни спектрограммы из своей базы. Положи их рядом. Найди закономерности и запиши их. Например: Птицы (соловей): Их звуки на спектрограмме находятся высоко (частоты >2000 Гц) и выглядят как сложные узоры. Лягушки: Их звуки находятся низко (100-500 Гц) и выглядят как повторяющиеся «столбики» с паузами. Сверчки: Это сплошная «щетка» из очень частых повторяющихся сигналов на высоких частотах (~3000 Гц).

Необходимо определить, по каким трём главным признакам будет отличать звук одного животного от другого. (Подсказка: Частотный диапазон, Ритм, Длительность).

#### ***Шаг 5. Разработка алгоритма классификации***

Посмотри на свою таблицу признаков. Видишь, все животные там уникальны. Преврати эти знания в инструкцию для компьютера. Это и есть алгоритм.

Пример алгоритма (запиши его в дневник):

ЕСЛИ (частота > 2000 Гц) И (ритм == «мелодичный») ТО это СОЛОВЕЙ  
 ЕСЛИ (частота от 100 до 500 Гц) И (ритм == «повторяющийся») ТО это

## ЛЯГУШКА

ЕСЛИ (частота от 2000 до 4000 Гц) И (ритм == «быстро повторяющийся») ТО это СВЕРЧОК

ЕСЛИ (частота от 400 до 2000 Гц) И (ритм == «одиночные крики») ТО это ВОРОНА

... и так для всех животных.

Выбери уровень сложности:

Начальный (Scratch): Создай блок-схему с условиями если ... то.

Продвинутый (Python): Начни писать код. Вот как будет выглядеть твой алгоритм в виде заготовки:

Python

# Это псевдокод - тебе нужно будет перевести его на настоящий Python!

звук = загрузить\_аудиофайл(«мой\_файл.wav»)

средняя\_частота = вычислить\_среднюю\_частоту(звук)

ритм = определить\_ритм(звук) # Это самая сложная часть, можно начать с простых проверок

if средняя\_частота > 2000 and ритм == «мелодичный»:

    животное = «Соловей»

elif средняя\_частота < 500 and ритм == «повторяющийся»:

    животное = «Лягушка»

elif 2000 <= средняя\_частота <= 4000 and ритм == «быстро повторяющийся»:

    животное = «Сверчок»

# ... и так далее по твоей таблице

print(«Определённое животное:», животное)

### ***Шаг 6. Создание интерфейса***

Выберите инструмент для создания интерфейса программы.

Scratch: Создай спрайт-кнопку. При нажатии на нее программа может проанализировать заранее заданный звук и сказать спрайтом: «Это лягушка!».

Python + Tkinter: Создай простое окно с кнопкой «Выбрать файл» и текстовой меткой для вывода результата.

Собери все вместе: Свяжи свой интерфейс с алгоритмом, который ты написал на предыдущем этапе.

Попробуйте загрузить в программу разные звуки из своей базы. Правильно ли она угадывает? Если нет — вернись к этапу 4 или 5 и уточни признаки.

### ***Шаг 7. Подготовка и защита проекта***

Создайте презентацию (3-4 слайда):

Слайд 1: Проблема и цель. («Мы решили научить компьютер понимать животных!»).

Слайд 2: Наш метод. Покажи свою таблицу признаков и несколько спектрограмм. Объясни, как ты отличал животных.

Слайд 3: Наш алгоритм. Покажи блок-схему или кусочек кода.

Слайд 4: Демонстрация! Покажи, как работает твоя программа вживую.

Распредели речь: Каждый член команды должен рассказать о своей части работы.

Подготовься к вопросам: Подумай, что могут спросить у тебя? («А почему вы решили, что это именно ворона?», «Что было самым сложным?»).

### **Ожидаемые результаты**

В результате выполнения проекта вы изучите характеристики звука, познакомитесь с применением Python для обработки аудиофайлов, научитесь моделировать звуки, и, наконец, создадите действующий прототип программы, имитирующей звуки, издаваемые различными животными.

# БЕЛАРУСКІ АРНАМЕНТ У ЛІЧБАВАЙ ГРАФІЦЫ

Данный проект — это мост между прошлым и будущим. На основе белорусского орнамента мы создадим современный цифровой продукт: открытку, логотип или плакат. Ты не просто нарисуешь картинку, ты оживишь наследие с помощью компьютерной графики.

## **Цель проекта:**

Создание цифрового изображения белорусского орнамента (узора) с использованием графического редактора и его оформление для дальнейшего применения (в виде постера, открытки, заставки, логотипа).

При выполнении проекта вы:

1. Изучите значение и символику традиционных белорусских орнаментов.
2. Познакомьтесь с возможностями графических редакторов для построения узоров и симметричных изображений.
3. Создайте цифровой макет орнамента или узора, используя инструменты графического редактора.
4. Оформите итоговый продукт (цифровой плакат, открытку или логотип).
5. Подготовьте и проведете презентацию с обоснованием выбора символики и способов работы.

## **Оборудование и необходимые материалы**

Компьютеры, проектор/интерактивная доска, графические редакторы (Paint.NET, GIMP, Inkscape), доступ к интернет-ресурсам (сайты о белорусской культуре, онлайн-библиотеки), презентационные программы (PowerPoint, LibreOffice Impress).

## **Ход выполнения проекта**

### ***Шаг 1. Погружение в проблему. Расшифровка кода предков***

Откроем тайный язык: Мы вместе посмотрим на самые известные белорусские узоры. Ты увидишь, что в них нет ничего случайного. Ответь для себя на вопросы:

*Это просто «красивые завитушки» или в них есть глубокий смысл?*

*Где в современном мире можно встретить такие узоры?* (На бренде одежды, сувенирах, в логотипах кафе или фестивалей).

*Как с помощью компьютера можно сохранить и передать эту красоту новым поколениям?*

Сформулируем задачу: Наша цель — создать цифровой продукт (открытку, логотип, плакат) на основе белорусского орнамента.



Как выполнить (твои шаги):

Заведи «Дневник символиста»: Это будет твоя главная тетрадь для проекта.

Начни исследование: Найди в интернете или книгах 3-5 самых известных белорусских орнаментов. Выпиши в дневник, что означают их основные элементы:

*Ромб/Квадрат — символ засеянного поля, плодородия, матери-земли.*

*Круг/Солнце — символ жизни, света, тепла, гармонии.*

*Восьмиконечная звезда — символ надежды, путеводности.*

*«Ёлочка»/«Папараць-кветка» — символ роста, развития, связи с природой.*

*Крест — символ огня, солнца, защиты.*

Придумай идею: Какой продукт тебе хочется создать? Может, открытку ко Дню матери с символом плодородия? Или логотип для школьного экологического кружка с «ёлочкой»?

Результат: В твоём дневнике есть расшифрованные символы и первая идея для будущего продукта.

## ***Шаг 2. Организация деятельности. Создаем творческий альянс***

Собери команду единомышленников (3-4 человека). Важно, чтобы вам нравилась общая тема. Распределите роли:

Историк-аналитик: Глубоко изучает символику, ищет аутентичные орнаменты, готовит историческую справку.

Художник-иллюстратор: Отвечает за эстетику. Придумывает, как традиционные мотивы превратить в современный дизайн, подбирает цвета.

Цифровой мастер (Графический дизайнер): Технический специалист, который будет работать в программе (Paint.NET, GIMP, Canva), воплощая эскиз в пиксели и векторы.

Арт-директор/Презентатор: Следит за целостностью идеи, готовит презентацию и ярко представляет ваш проект аудитории.

Определите продукт: Что именно вы создадите всей командой? Выберите одно:

*Праздничная открытка (ко Дню Победы, 8 Марта, Новому году).*

*Логотип (для вашего класса, школьной спортивной команды, театрального кружка).*

*Плакат (ко Дню родного языка, «Дню вышиванки»).*

*Обложка для виртуального дневника или альбома.*

Составьте техническое задание (ТЗ):

Напишите простой план:

1) Собрать орнаменты;

- 2) Нарисовать эскиз;
- 3) Сделать в программе;
- 4) Оформить презентацию.

### ***Шаг 3. Поэтапное выполнение проекта. От идеи до пикселя***

Это самый большой и творческий этап. Разбиваем его на шаги.

#### **Шаг 3.1: Сбор и анализ информации**

Создайте «Мудборд» (доску вдохновения): В отдельном документе или на листе ватмана соберите все найденные изображения орнаментов, их описания, цветовые палитры.

Сделайте выбор: Какой конкретный мотив (ромб, солнце, елочка) или их комбинацию вы возьмете за основу? Запишите в ТЗ, почему именно этот выбор (например: «Выбираем солнце и ромб, чтобы передать идею плодородия и света для открытки ко Дню матери»).

#### **Шаг 3.2: Создание эскиза**

Не включайте компьютер! Возьмите карандаш и бумагу.

Набросайте композицию: Где будет главный элемент? Как будет построена симметрия? Где разместится текст (если он нужен)?

Поэкспериментируйте с цветом: Помните о традиционной палитре: красный (жизнь, красота), белый (свет, чистота), черный (земля), зеленый (природа, рост). Раскрасьте эскиз.

#### **Шаг 3.3: Работа в графическом редакторе (3-4 часа)**

Выберите ваш «цифровой мольберт»:

*Paint.NET / GIMP — отлично подходят для растровой графики (как рисунок кистью).*

*Inkscape / Canva — хороши для векторной графики (четкие линии и формы, которые можно увеличивать без потери качества).*

Освойте ключевые инструменты: геометрические фигуры, кисть/карандаш, заливка, ластик, выделение и отражение (для симметрии!), работа со слоями (каждый элемент на отдельном слое — это магия!).

Переносите эскиз в программу: Начните с фона, потом добавьте основные элементы. Используйте копирование и отражение, чтобы добиться идеальной симметрии.

Работайте с цветом: Используйте палитру или «пипетку», чтобы точно подобрать оттенки.

#### **Шаг 3.4: Финальные штрихи и сохранение**

Добавьте текст (если нужно): выберите подходящий, легко читаемый шрифт.

Проверьте композицию: Уберите лишнее. Все ли гармонично?

Сохраните правильно:

Основной файл с слоями (.psd, .xcf, .svg) — чтобы можно было исправить.

Итоговое изображение для показа (.png — если нужен прозрачный фон, .jpeg — для фотокачества).

#### ***Шаг 4. Защита проекта. Твой вернисаж***

Создайте короткую, но эффектную презентацию (3-5 слайдов):

Слайд 1: Название и команда. Красиво представьте ваш проект и себя.

Слайд 2: Идея и символика. Покажите орнамент-прототип и расскажите главную мысль: «Мы взяли символ солнца (круг), потому что он означает жизнь и радость, и добавили ромб — символ плодородия, чтобы создать открытку для самой родной человека».

Слайд 3: Процесс создания. Покажите ваш бумажный эскиз и скриншот работы в программе. Это очень интересно!

Слайд 4: Готовый шедевр! Во весь экран продемонстрируйте вашу открытку, логотип или плакат.

Распределите речь: Каждый рассказывает о своей части работы. Говорите с увлечением!

*Историк: «Мы узнали, что этот узор раньше вышивали на свадебных рушниках...»*

*Художник: «Мы решили использовать красно-белую гамму, потому что...»*

*Дизайнер: «Самым сложным было сделать идеальную симметрию с помощью инструмента...»*

Отрепетируйте! Уложите в 5-7 минут. Следите за осанкой и уверенным голосом.

Результат этапа: Яркая презентация и уверенная защита вашего культурного проекта.

#### **Ожидаемые результаты**

В результате выполнения проекта вы создадите комплексный культурно-цифровой продукт, объединяющий историческое наследие и современные технологии. Результаты проекта являются наглядными, презентуемыми и имеют практическую применимость.

Итоговым продуктом проекта является не просто «картинка», а осмысленный цифровой артефакт, готовый к использованию в современной коммуникативной среде (социальные сети, школьные мероприятия, сувенирная продукция) и представляющий собой результат синтеза культурологического исследования и компьютерного творчества.

# ИНФОГРАФИКА: ПРЕДСТАВЛЕНИЕ ДАННЫХ В ВИЗУАЛЬНОЙ ФОРМЕ

Каждый день на нас обрушивается поток информации: цифры, факты, новости, но есть способ не просто рассказать об этой информации, а показать её так, чтобы она была понятна, интересна и запоминалась с первого взгляда. Этот способ — *инфографика*.

В данном проекте каждый может почувствовать себя настоящим медиа-художником, который превращает скучные цифры в яркую историю. Вы научитесь находить данные, анализировать их и создавать визуальные шедевры, которые смогут увлечь друзей, одноклассников и даже учителей.

Реализация проекта позволяет пройти весь путь создания профессионального продукта: от рождения идеи до ее публичной защиты. Вы будете искать информацию как детективы, структурировать ее как ученые и оформлять, как дизайнеры.

**Цель проекта:** научиться собирать, анализировать и визуализировать информацию, создавая инфографику по актуальной теме.

**При выполнении проекта вы:**

соберете и систематизируете статистические данные из разных источников;

освоите основные правила визуализации информации (выбор цвета, шрифта, структуры);

создадите инфографику по выбранной теме, используя цифровые инструменты;

оформите и защитите итоговый продукт в виде презентации.

**Оборудование и необходимые материалы**

Компьютеры, проектор/интерактивная доска, офисные пакеты (MS Excel, PowerPoint, LibreOffice Calc/Impress), интернет-ресурсы (онлайн-сервисы для инфографики — Canva, Piktochart и др.).

**Ход выполнения проекта**

**Этап 1. Погружение в проблему (индивидуальная работа)**

Сначала разберись, что такое крутая инфографика, выбери тему, от которой «загорятся глаза».

Вопросы для выбора темы:

*Что быстрее и проще понять: страницу с таблицей из 100 цифр или один яркий график?*

*Где в жизни мы сталкиваемся с инфографикой?*

*(Подсказка: карта метро, инструкция к лекарству, график погоды в*

*приложении, посты в Instagram).*

Наша общая миссия — создать собственную инфографику на актуальную тему.

Как выполнить проект (твои шаги)?

Включаем критическое мышление. Когда смотрим примеры, задаем себе вопросы: «Ясно ли мне сразу, о чем это?», «Красиво ли это?», «Не перегружено ли это лишними деталями?».

Заведи «Блокнот идей». Запиши в него темы, которые представляются интересными. Например:

*«Какой соцсетью и сколько времени пользуются учащиеся нашего класса?»*

*«Сколько пластика мы выбрасываем за неделю?»*

*«Сравнение режима дня отличников и тех, кто учится хуже»*

*«Беларусь: самые интересные цифры (население, природа, IT-достижения)»*

Выбери одну тему, которая волнует лично тебя и твою команду. От этого зависит весь успех проекта!

Результат этапа. В твоём блокноте есть четко сформулированная тема и цель проекта.

### ***Этап 2. Организация деятельности. Создаем команду мечты***

Необходимо стать слаженной командой, где у каждого есть своя важная роль и понятные задачи.

Как выполнить эту задачу (твои шаги):

1. Объединись с 3-4 одноклассниками. Важно, чтобы вам было комфортно работать вместе.

2. Распредели роли. Примерные роли:

*Аналитик/Исследователь.* Этот человек обожает копаться в фактах. Он ищет достоверные данные, проверяет источники, как Шерлок Холмс.

*Дизайнер/Визуализатор.* Это творческий гений. Он придумывает, как будут выглядеть графики, подбирает цвета, шрифты и иконки. У него хороший вкус.

*Оформитель/Технический специалист.* Этот человек дружит с технологиями. Он именно тот, кто будет работать в программе (Canva, PowerPoint), собирая все идеи в единый цифровой продукт.

*Презентатор/Спикер.* Это «лицо» команды. Умный, харизматичный, он сможет ярко и уверенно представить ваше творение на защите.

*(Важно: роли можно совмещать!)*

1. Составь «Дорожную карту». На листе бумаги или в онлайн-документе напиши план:

Что делать? (Найти статистику, провести опрос, нарисовать эскиз, создать в Canva).

Кто отвечает? (Имя участника).

К какому сроку? (Например, «сбор данных до пятницы»).

Результат этапа: У тебя есть команда, распределенные роли и план-график работы.

### ***Этап 3. Сбор и анализ информации. В погоне за цифрами***

Цель: найти надежную информацию и превратить ее в готовые к визуализации данные.

Как выполнить (твои шаги):

Отправляйся на «охоту» за данными:

Официальные источники: Сайты Белстата (Национальный статистический комитет), министерств.

Социологические опросы. Можно провести свой! Создай короткую анкету в Google Формах и узнай мнение одноклассников.

Научные и образовательные порталы: Проверь, чтобы у информации был автор и источник.

Любую цифру, найденную в интернете, старайся проверить по другому источнику. Занеси все найденные данные в единую таблицу (в Excel, Google Таблицах или даже в тетради).

*Таблица. Пример таблицы для твоей темы «Соцсети»*

| Вопрос / Параметр                 | Данные                                     | Примечание                       |
|-----------------------------------|--------------------------------------------|----------------------------------|
| Самая популярная соцсеть в классе | Instagram (60%)                            | По результатам опроса 25 человек |
| Среднее время в соцсетях в день   | 2,5 часа                                   |                                  |
| Самая быстрорастущая сеть         | TikTok                                     | По данным за 2024 год            |
| Источник                          | [Ссылка на ваш опрос],<br>[Ссылка на сайт] |                                  |

Посмотри на свою таблицу и ответь на вопрос: «И что?». Что эти цифры значат? Какая главная мысль? Например: «Хотя Instagram лидирует, TikTok стремительно набирает популярность среди подростков».

Результат этапа. У тебя есть таблица с проверенными

данными и ключевые выводы, которые ты хочешь донести.

#### ***Этап 4. Создание эскиза инфографики. Рисуем макет***

Создать «черновик» или «скелет» твоей будущей инфографики, чтобы не запутаться при цифровом оформлении.

Как выполнить (твои шаги):

1. Возьми лист бумаги и карандаш. Не нужно ничего рисовать идеально!

2. Разметь пространство:

Где будет главный заголовок?

Куда ты поместишь самую важную и яркую диаграмму?

Где будут второстепенные графики или факты в рамочках?

Как ты будешь вести взгляд зрителя: сверху вниз или слева направо?

3. Подбери визуальные элементы:

Круговая диаграмма — идеально, чтобы показать доли (проценты).

Столбчатая диаграмма — чтобы сравнить несколько величин.

Иконки — чтобы быстро донести смысл (телефон для времени в сети, земной шар для глобальной статистики).

Стрелки и линии — чтобы связать разные части.

4. Подумай о цвете: Набросай, каким цветом ты выделишь главное, а каким — второстепенное.

Результат этапа. У тебя на руках есть бумажный эскиз — понятный план для следующего этапа.

#### ***Этап 5. Создание инфографики в цифровом виде. Волшебство в действии***

Цель: вдохнуть жизнь в твой эскиз, создав яркий и профессиональный цифровой продукт.

Как выполнить (твои шаги):

1. Выбери свой инструмент:

[Canva.com](https://www.canva.com) (рекомендуется): Огромная библиотека шаблонов, иконок, шрифтов. Очень простой и интуитивный интерфейс.

PowerPoint: Ты умеешь в нем работать, в нем тоже можно создавать крутые диаграммы и композиции.

2. Следуй эскизу: Открой программу и начинай переносить свой бумажный макет. Не отклоняйся от плана!

3. Помни о золотых правилах:

Меньше — значит лучше: Не превращай инфографику в «помойку» из элементов. Оставляй пустое пространство.

Шрифты: Используй не больше 2-3 разных шрифтов. Главное — чтобы текст был хорошо читаем.

Цвета: Подбери гармоничную палитру. Можно найти готовые цветовые

схемы в интернете.

Единство стиля: Все иконки и графики должны быть выдержаны в одном стиле.

4. Собери все вместе: Добавь диаграммы (их можно построить прямо в Canva, введя свои цифры), вставь иконки, размести текст.

Результат этапа: Готовая, сохраненная в виде картинки или PDF-файла цифровая инфографика — главный продукт твоего проекта!

### ***Этап 6. Подготовка презентации и защита проекта. Твой звездный час***

Цель: блестяще представить свою работу и доказать, что она важна и интересна.

Как выполнить (твои шаги):

1. Создай короткую презентацию (3-4 слайда):

Слайд 1: Приветствие и проблема. Кто мы и какую проблему/тему исследуем.

Слайд 2: Как мы это делали. Покажи самый интересный момент из твоего исследования (например, скриншот опроса или таблицы). Расскажи, откуда данные.

Слайд 3: Наше открытие и инфографика. Сделай главный вывод и ПРЕДСТАВЬ саму инфографику во всей красе!

2. Распиши речь: Каждый участник команды должен говорить о том, что делал он.

Аналитик: «Мы нашли данные вот здесь... и были удивлены, что...»

Дизайнер: «Мы выбрали такие цвета и графики, потому что...»

Презентатор: «И как вы видите, наша инфографика наглядно показывает...»

3. Отрепетируй! Прорепетируйте выступление несколько раз. Уложите в время (5-7 минут). Попроси друга посмотреть и дать feedback.

Результат этапа: Готовая презентация, отрепетированное уверенное выступление и моральная готовность к любым вопросам.

### **Ожидаемые результаты**

В результате вы создадите реальный продукт, позволяющий представить огромное количество информации в компактном и привлекательном виде. Вы приобретете навыки, которые особенно ценятся в современном мире: умение находить нужную информацию, выделять в ней главное. Вы освоите новые инструменты для творчества и создадите с их помощью «говорящее» изображение. Вы научитесь работать в команде, чувствовать себя ее частью, станете увереннее в себе.



# ЗНАКОМСТВО С 3D-РЕДАКТОРОМ:

## МОДЕЛИРОВАНИЕ МАТЕМАТИЧЕСКИХ ФИГУР

Ты играл в компьютерные игры, смотрел мультфильмы, видел макеты новых зданий? Всё это начинается здесь и сейчас — с создания трёхмерной модели. Представь: ты сможешь взять абстрактные формулы из учебника математики и превратить их в осязаемый, пусть и цифровой, объект. Этот проект — твой первый шаг в мир 3D-моделирования, где твоим инструментом будут не только компьютерные программы, но и твои знания по геометрии. Ты пройдёшь путь от идеи до готовой модели и её защиты. Ты научишься видеть в окружающих объектах простые геометрические формы, освоишь базовый 3D-редактор и поймёшь, как знание площади и объёма делает тебя настоящим профессионалом.

### **Цель проекта:**

Освоение базовых навыков работы в 3D-редакторе через создание моделей математических тел и исследование их свойств.

При выполнении проекта вы:

1. Научитесь создавать и изменять базовые геометрические объекты (куб, цилиндр, сфера, пирамида).
2. Используйте знания по математике для расчёта параметров фигур (объём, площадь поверхности, соотношения размеров).
3. Разработаете собственную 3D-модель, включающую несколько геометрических тел.
4. Научитесь оформлять результаты в виде презентации и защищать проект.

### **Оборудование и технические материалы**

Компьютеры с установленным 3D-редактором (Blender / Tinkercad / FreeCAD), офисные программы (PowerPoint, Word).

### **Ход выполнения проекта**

#### ***Этап 1. Погружение в проблему: включаем воображение***

Смотрим на мир глазами 3D-дизайнера: Мы посмотрим, как 3D-графика используется в кино, играх, архитектуре и даже медицине.

Ответь себе на вопросы:

*Какой предмет в этой комнате можно было бы смоделировать?*  
(Подсказка: всё состоит из простых фигур!)

*Зачем вообще уметь это делать? (Создать модель для печати на 3D-принтере, для своей игры или просто для крутой презентации).*

*Формулируем главную задачу: Наша цель — создать 3D-модель реального или придуманного объекта, используя знания геометрии.*

Выбери объект для моделирования. Начни с простого, состоящего из понятных форм.

Например:

Шахматная фигура (король = цилиндр + пирамида + кубик).

Сторожевая башня (цилиндр или куб + конус).

Домик с крышей (куб + треугольная призма).

Футуристическая ваза (сфера + цилиндр).

### ***Этап 2. Организация деятельности. Собираем команду мечты***

Объединись с единомышленниками (3-4 человека). Важно, чтобы вам нравилась общая идея. Распределите роли, как в студии разработки:

3D-Моделлер (Дизайнер): Главный волшебник в программе. Отвечает за создание и визуализацию модели.

Инженер-расчётчик (Аналитик): Главный математик. Считает объёмы, площади, проверяет пропорции. Его девиз: «Цифры не врут».

Гейм-дизайнер (Оформитель): Создаёт историю вокруг модели. Готовит презентацию, придумывает, как эффектно всё подать.

Технический директор (Докладчик): Знает проект лучше всех. Уверенно представляет его миру, отвечает на каверзные вопросы.

Создайте «Дорожную карту проекта»: Напишите простой план в общем документе:

*Неделя 1. Освоить редактор, сделать первые фигуры.*

*Неделя 2. Рассчитать математику для простых фигур.*

*Неделя 3. Создать итоговую модель, сделать скриншоты.*

*Неделя 4. Сделать презентацию и отрепетировать защиту.*

### ***Этап 3. Освоение инструментария 3D-редактора: первые шаги в новом мире***

Выберите «цифровой пластилин», чаще всего используют:

*Tinkercad* — самый простой, браузерный, идеален для новичков.

*Blender* — мощный, профессиональный, но сложнее.

Освой базовые движения:

Добавить фигуру: Найди панель с кубом, сферой, цилиндром и перетащи на рабочую плоскость.

Выделить объект: Щёлкни по нему.

Перемещать (Grab/Move): Зажми и тащи по осям (X — красная, Y — зелёная, Z — синяя).

Вращать (Rotate): Покрути объект.

Масштабировать (Scale): Растяни или сожми.

Выполни задание-разминку: Создай сцену «Геометрический город»:

Куб со стороной 4 см (это дом). Цилиндр радиусом 1 см и высотой 6 см (труба на доме). Сферу радиусом 2 см (солнце или шар).

#### ***Этап 4. Построение математической модели: цифры оживают***

Вернись к своей сцене «Геометрический город». Достань учебник геометрии или шпаргалку с формулами. Рассчитай объём и площадь поверхности для каждой фигуры в твоей сцене. Работай вместе с Инженером-расчётчиком.

Для куба (дом):  $V = a^3$ ,  $S = 6a^2$ .

Для цилиндра (труба):  $V = \pi R^2 h$ ,  $S_{\text{бок.}} = 2\pi R h$  (+ площади оснований).

Для сферы (шар):  $V = (4/3)\pi R^3$ ,  $S = 4\pi R^2$ .

В редакторе (например, в Tinkercad) часто можно посмотреть свойства объекта. Сравни свои расчёты с тем, что показывает программа (они могут немного отличаться из-за округлений). Это твоя математическая проверка модели.

Результат этапа: Ты видишь прямую связь между формулой и объектом на экране. Твоя модель теперь не просто картинка, а точный расчёт.

#### ***Этап 5. От простого к сложному: создание итоговой 3D-модели***

Открой эскиз выбранного объекта (из Этапа 1). Разбей его мысленно на простые геометрические тела. Начни моделировать в редакторе.

Последовательность — твой друг: Создай самую большую, базовую деталь (например, основу башни — большой цилиндр). Постепенно добавляй и «прикрепляй» к ней более мелкие детали (балконы, окна, крышу). Используй инструменты вычитания (Hole) и объединения (Group) для сложных форм. Веди «Дневник разработки»: Делай скриншоты после каждого важного шага. Это пригодится для презентации! Не забывай про математику: После создания каждой крупной детали, Инженер-расчётчик должен посчитать её приблизительный объём и площадь. Это помогает держать модель в точных пропорциях.

#### ***Этап 6: Подготовка и презентация результатов***

Создай презентацию, которая «зацепит» (3-4 слайда):

Слайд 1: Идея. Крутая картинка-ассоциация и название проекта. «Мы строим не просто башню, а хранилище знаний!».

Слайд 2: Магия геометрии. Покажи фотографию реального прототипа (или эскиз) и рядом — его разбор на простые фигуры (куб, цилиндр и т.д.) с подписями формул для расчёта.

Слайд 3: Процесс. Коллаж из твоих скриншотов «До/После». Покажи, как из кучи фигур родился шедевр.

Слайд 4: Наш шедевр! Вставь финальный рендер (красивый скриншот) своей модели. Можно даже показать короткую вращающуюся анимацию!

Продумай речь: Каждый говорит о своём.

*Моделлер: «Самым сложным было точно совместить крышу и стены, пришлось использовать инструмент...»*

*Расчётчик: «Мы рассчитали, что объём основной башни составил 1250 см<sup>3</sup>, это как 5 пачек сока!»*

*Докладчик: «Таким образом, наша модель доказывает, что даже сложный объект можно разложить на математические формулы и создать в цифровом мире».*

### **Ожидаемые результаты**

В результате выполнения проекта вы освоите первоначальные навыки 3D-моделирования, научитесь строить объемные фигуры, освоите приемы масштабирования и комбинирования фигур; закрепите свои познания в математике. Вы создадите готовую, сохранённую 3D-модель задуманного вами объекта, папку со скриншотами этапов работы и презентацию результатов.

# МОДЕЛИРОВАНИЕ ДВИЖЕНИЯ ТЕЛА ПО ЗАКОНАМ ФИЗИКИ

Как компьютерные игры так точно рассчитывают полёт стрелы, прыжок персонажа или траекторию мяча? Секрет в том, что игра «решает» те же уравнения, что и ты на уроке физики!

## **Цель проекта:**

Научиться моделировать механическое движение тела по законам физики и визуализировать результаты в виде анимации или интерактивной программы.

При выполнении проекта вы:

Углубите знания по физики (скорость, ускорение, масса, сила, сопротивление).

Освоите численные методы интегрирования (метод Эйлера как вводный) для решения уравнений движения.

Реализуете в программной среде модель движения в нескольких задачах: свободное падение, бросок под углом, движение с сопротивлением.

Построите графики зависимости координаты, скорости и ускорения от времени.

Сравните аналитические решения с численными результатами и проанализировать погрешности.

Подготовите презентацию с пояснениями.

## **Оборудование и необходимые материалы**

Компьютеры с установленными средами: Python (Jupyter/IDLE) + библиотеки matplotlib, numpy, tkinter / или среда Scratch (упрощённый вариант) / или GeoGebra (динамическое моделирование), калькуляторы, тетради.

## **Ход выполнения проекта**

### ***Этап 1. Погружение в проблему: ставим физический эксперимент***

Посмотрим на видео, как летит мяч, падает яблоко, движется стрела. Зададимся главным вопросом: «Можно ли предсказать, где окажется мяч через секунду, зная только силу и угол броска?». Создадим команду исследователей: Чтобы ответить на такой сложный вопрос, нужны специалисты.

Главный физик (Аналитик): Твой инструмент — учебник и формулы. Ты будешь диктовать законы, по которым живёт наша виртуальная вселенная. Твоя задача — не ошибиться в уравнении.

Волшебник кода (Программист): Ты — переводчик. Ты превратишь слова и формулы физика на язык, понятный компьютеру (Scratch, Python). Без тебя модель останется на бумаге.

Художник данных (Визуализатор): Мир твоей симуляции должен быть красивым и понятным. Ты создашь графики, траектории, подпишешь оси. Ты сделаешь так, чтобы все увидели физику.

Испытатель и оратор (Тестировщик/Презентатор): Ты — связь с реальным миром. Ты проверишь, не «слетает» ли программа, и блестяще представишь наш проект, отвечая на любые вопросы.

Сформулируй конечную цель: Запиши её в общем документе: «Создать программу, которая визуализирует движение брошенного тела и позволяет менять его параметры (скорость, угол)».

### ***Этап 2. Изучение основ движения: пишем «правила игры»***

Веди за собой! Объясни команде простыми словами: Скорость: Насколько быстро тело меняет своё место. Ускорение: Насколько быстро меняется сама скорость. Самое главное ускорение —  $g = 9.8 \text{ м/с}^2$  (сила тяжести, всегда тянет вниз!). Траектория: Линия, которую «рисует» движущееся тело. Проведи мысленный эксперимент: Представь, что бросаешь мяч. Если бросить его сильнее (больше начальная скорость) — он улетит дальше. Если бросить под углом  $45^\circ$  — он улетит максимально далеко (это твоя гипотеза, которую можно проверить в модели!). Если бы не было воздуха (сопротивления), мяч летел бы по идеальной параболе. Начни думать о формуле. Самое простое движение, которое мы будем моделировать, описывается двумя волшебными строками для каждой координаты (X и Y):

Новая координата X = Старая координата X + (Начальная скорость по X) \* время

Новая координата Y = Старая координата Y + (Начальная скорость по Y) \* время - ( $g$  \* время<sup>2</sup>) / 2

Результат этапа: У всей команды есть общее понимание физических законов, которые лягут в основу программы.

### ***Этап 3. Сбор исходных данных: выбираем героя и его характеристики***

Выбери «героя» своей симуляции:

*Мяч (классика, понятно всем).*

*Снаряд из катапульты (эпично!).*

*Бумажный самолётик (с ним можно потом добавить больше сопротивления воздуха).*

Задай ему стартовые характеристики (это твои «рычаги управления»):

*Начальная высота: С земли (0 м) или с крыши (5 м)?*

*Начальная скорость: Слабо (5 м/с) или сильно (20 м/с)?*

*Угол броска: Почти горизонтально (10°), под оптимальным углом (45°) или почти вертикально (80°)?*

Создай «Паспорт объекта»: Запиши всё это в таблицу. Эти числа ты позже вставишь в свою программу как начальные значения переменных.

#### ***Этап 4. Создание компьютерной модели движения: оживляем формулы***

Выбери свой инструмент:

Scratch (самый наглядный): Используй блоки «изменить  $x$  на ...», «изменить  $y$  на ...». Силу тяжести сделай блоком «изменить  $y$  на -1» каждый шаг цикла.

Python (более мощный): Используй библиотеку `pygame` или `turtle`. Создай цикл, где на каждом шаге пересчитываешь координаты по формулам из Этапа 2.

Двигайся шаг за шагом:

Сначала сделай простое падение вниз (только начальная скорость по  $Y$ ).

Потом добавь горизонтальный полёт (только начальная скорость по  $X$ ).

А теперь объедини их — и получишь бросок под углом! Для этого начальную скорость нужно разложить на  $V_x$  и  $V_y$ , используя  $\sin$  и  $\cos$ .

#### ***Этап 5: Проверка работы модели***

Проведи серию виртуальных экспериментов:

Эксперимент 1: Брось мяч с одной и той же скоростью, но под углами 30°, 45° и 60°. При каком угле он улетел дальше всех? Совпадает ли с теорией (45°)?

Эксперимент 2: Брось мяч вертикально вверх. Вернулся ли он точно в точку старта? Достигает ли он пика на половине общего времени полёта?

Найди «баги» в реальности (или в модели): Если мяч в программе летит НЕ так, как подсказывает интуиция или учебник, значит:

Либо «Главный физик» ошибся в формуле.

Либо «Волшебник кода» неверно её запрограммировал.

Либо вы забыли про какой-то фактор (например, силу трения о воздух).

Скорректируйте модель. Идеал недостижим, но приближение к реальности — отличный результат!

#### ***Этап 6. Визуализация результатов: превращаем числа в историю***

Научи программу рисовать линию по следу движения мяча. Получится его траектория — та самая парабола.

Создай два отдельных графика: как меняется высота ( $Y$ ) со временем и как меняется дальность ( $X$ ). График высоты будет похож на гору: взлёт — пик — падение.

Цветовая динамика: пусть цвет мяча меняется в зависимости от его скорости (быстро = красный, медленно = синий).

Подписи: Добавь на экран цифры: текущая скорость, время полёта, пройденное расстояние.

### ***Этап 7. Подготовка презентации и защита проекта***

Создайте презентацию (3-4 слайда):

Слайд 1: Вопрос. «Мы хотели узнать, можно ли предсказать полёт мяча. И мы научили это делать компьютер!».

Слайд 2: Закон. Покажи главную формулу и объясни её суть (без сложных выводов). Скажи: «Это правило, которое мы вложили в программу».

Слайд 3: Магия. Покажи скриншот или гифку своей работающей симуляции с траекторией. Скажи: «Вот как это правило выглядит в движении».

Слайд 4: Открытие. Покажи результат своего ключевого эксперимента (например, что угол  $45^\circ$  — действительно самый выгодный для дальности).

Распределите речь:

Физик объясняет законы.

Программист рассказывает, как их «оживил».

Художник показывает, какую красоту удалось создать.

Испытатель подводит итоги и держит удар на вопросах.

Не просто рассказывайте, а запускайте симуляцию на большом экране в реальном времени. Меняйте параметры и показывайте, как мгновенно меняется траектория.

Результат этапа: Уверенная, эффектная защита, которая доказывает, что вы — команда, способная смоделировать кусочек мира.

### **Ожидаемые результаты**

В результате выполнения проекта вы ближе познакомитесь с законами движения тел. Использование таких инструментов, как Scratch, GeoGebra или Python позволит вам создать модели движения тел, визуализировать траектории, получать количественные характеристики движения. Работая совместно со своими товарищами, вы получите навыки коллективного обсуждения полученных результатов, нахождения оптимального решения поставленной задачи и, в итоге, представите результат выполнения проекта в виде готовой модели, презентации и сообщения.



# ХИМИЯ В ЧИСЛАХ: ИЗУЧЕНИЕ СВОЙСТВ ЭЛЕМЕНТОВ ЧЕРЕЗ ВИЗУАЛЬНОЕ ПРОГРАММИРОВАНИЕ

Ты когда-нибудь задумывался, что числа в таблице Менделеева — это не просто скучные цифры? Это целые истории: атомная масса рассказывает о «тяжеловесах» и «невесомых» элементах, температура плавления — о том, кто выдержит адский жар, а кто растает в руках. Но как эти истории увидеть и рассказать другим?

Этот проект — твой шанс стать цифровым алхимиком и дата-художником. Ты научишься превращать абстрактные химические данные в яркие, понятные и даже интерактивные визуализации. Ты создашь не просто презентацию, а настоящую цифровую лабораторию в Scratch, где можно исследовать и сравнивать элементы вживую!

## **Цель проекта:**

Создать в среде Scratch интерактивный проект (например, «сравнительную панель»), который наглядно показывает различия в свойствах выбранных химических элементов с помощью анимации, графиков или шкал.

## **При выполнении проекта вы:**

Выберете группу родственных или контрастных элементов (5-8 штук) и соберёшь точные данные об их ключевых свойствах (атомная масса, температура плавления/кипения, плотность).

Подготовите данные к визуализации:

Проанализируете собранные числа: найдёте максимальные и минимальные значения, чтобы с помощью простой формулы «перевести» реальные величины в размеры спрайтов или координаты в Scratch (например, превратить температуру от  $-200^{\circ}\text{C}$  до  $+200^{\circ}\text{C}$  в длину столбика от 10 до 300 пикселей).

Разработаете и запрограммируете проект в Scratch:

Создадите сцену с элементами управления (кнопки, переключатели) и запрограммируете спрайты так, чтобы они меняли свой вид (размер, цвет, положение) в зависимости от выбранного свойства элемента.

## **Оборудование и необходимые материалы**

Компьютер с доступом к Scratch 3.0;

таблица Менделеева (бумажная или цифровая);

справочные данные: атомная масса, температура плавления/кипения, плотность веществ;

линейка, таблицы, калькулятор (для преобразования и нормировки данных).

## Ход выполнения проекта

### *Этап 1. Погружение в проблему. Разгадываем числовые шифры элементов*

Смотрим на строки чисел (атомная масса, плотность). Задаём вопросы: Почему у Осмия (Os) плотность  $22,59 \text{ г/см}^3$  — это много? А у Лития (Li) всего  $0,534 \text{ г/см}^3$  — это как? Как это вообще представить? Где мы встречаем крутые графики и диаграммы? (Спорт: графики скорости, соцсети: диаграммы просмотров). Как они помогают понять информацию?

Наша цель — создать в Scratch интерактивный проект-визуализатор, который превращает числовые свойства 5-10 химических элементов в наглядные, сравнимые графики, диаграммы или анимированные шкалы.

Как выполнить (твои шаги):

Заведи «Лабораторный журнал визуализатора».

Проведи «разведку» свойств. Выпиши в журнал 3 самых удивительных, на твой взгляд, свойства из химии и подумай, как их можно показать:

Атомная масса: Можно показать шариками разного размера?

Плотность: Столбиками разной высоты или цветом (чем плотнее — тем «тяжелее» цвет)?

Температура плавления: Шкалой термометра или цветом (от синего = холодный, до красного = очень горячий)?

Сформулируй гипотезу: «Я думаю, что если показать плотности элементов столбиковой диаграммой, сразу будет видно, что металлы тяжелее неметаллов».

### *Этап 2. Организация деятельности. Формируем научно-визуальную лабораторию*

Собери команду единомышленников (3-5 человек). Распределите роли в лаборатории:

Химик-датель (Сборщик данных): Твой инструмент — таблицы и справочники. Ты находишь точные, проверенные числа: атомную массу, плотность, температуры. Ты — гарант точности информации.

Математик-масштабатор (Аналитик): Твой инструмент — калькулятор и логика. Ты превращаешь абсолютные значения (например, плотность от 0,5 до 22) в относительные, удобные для Scratch (например, от 10 до 300 пикселей). Ты делаешь числа «рисовабельными».

Программист-визуализатор (Кодер): Твой инструмент — Scratch. Ты оживляешь расчёты математика: создаёшь спрайты, которые меняют размер/цвет в зависимости от числа, программируешь кнопки и слайды.

Дизайнер-интерфейсер (Художник): Ты отвечаешь за красоту и понятность. Ты создаёшь стиль проекта: фон, шрифты, цвета диаграмм, расположение элементов, чтобы всё было ясно и приятно глазу.

Создайте «Техническое задание проекта».

Тема: Визуализация свойств элементов 1-го и 17-го групп (Щелочные металлы и Галогены) — они очень разные и интересные для сравнения!

Что визуализируем: Атомную массу (столбиковая диаграмма) и температуру плавления (анимированный термометр).

Интерфейс: Две кнопки для выбора свойства, при нажатии — появляется диаграмма с подписями.

### ***Этап 3. Реализация проекта. Алхимия данных и кода***

Модуль 3.1: Сбор и анализ данных — находим «золото» в цифрах

Создайте таблицу данных в Google Таблицах или на бумаге.

*Таблица. Пример таблицы данных в Google Таблицах*

| Химический элемент | Символ | Атомная масса (а.е.м.) | t плавления (°C) | Масштаб: высота столбика (от 50 до 300) | Масштаб: длина термометра (от 10 до 200) |
|--------------------|--------|------------------------|------------------|-----------------------------------------|------------------------------------------|
| Литий              | Li     | 6.94                   | 180.5            | ?                                       | ?                                        |
| Натрий             | Na     | 22.99                  | 97.8             | ?                                       | ?                                        |
| ...                | ...    | ...                    | ...              | ...                                     | ...                                      |
| Фтор               | F      | 18.99                  | -219.6           | ?                                       | ?                                        |
| Хлор               | Cl     | 35.45                  | -101.5           | ?                                       | ?                                        |

Найдите MAX и MIN значение для атомной массы в вашем списке.

Формула для высоты спрайта в Scratch:  $\text{Высота} = 50 + ((\text{Масса} - \text{MIN}) / (\text{MAX} - \text{MIN})) * 250$ .

*Пример: Если масса от 6.94 до 85.47, то для Na (22.99):  $\text{Высота} = 50 + ((22.99 - 6.94) / (85.47 - 6.94)) * 250 \approx 50 + 51 \approx 101$  пиксель.*

Сделайте выводы: «У элементов 1 группы атомная масса растёт сверху вниз, а температура плавления падает!».

Модуль 3.2: Программирование и визуализация в Scratch — даём данным форму

Подготовьте «сцену»:

Создайте два фона: титульный слайд и слайд с рабочей областью.

Нарисуйте или найдите спрайты для кнопок «Атомная масса» и «Температура».

Создайте «умные» спрайты-столбики:

Сделайте один спрайт «Столбик». Для каждого элемента создайте его клон.

В момент создания клона задайте ему уникальные свойства через переменные-списки или отдельные переменные:

*scratch*

*когда я клоном*

*задать [мой\_номер v] в (1) // для Лития*

*перейти в x: (-150) y: (-100)*

*изменить размер на (50) // базовый размер*

*изменить размер на (элемент (мой\_номер) из [Высоты\_массы v]) // вот тут магия масштабирования!*

*сказать (объединить (элемент (мой\_номер) из [Названия v]) [ ( ] (элемент (мой\_номер) из [Массы v]) [ ]))*

Запрограммируйте переключение:

При нажатии на кнопку «Температура» отправьте сообщение Переключить на температуру.

Все клоны столбиков, получив это сообщение, должны изменить свою высоту, взяв новое значение из списка Высоты\_температуры.

Добавьте интерактива: Сделайте так, чтобы при наведении курсора на столбик появлялась подробная подсказка в отдельном поле.

Модуль 3.3: Тестирование и «полировка»

Вся команда тестирует продукт.

Все ли кнопки работают?

Столбики соответствуют данным? Не «прыгает» ли Литий выше Цезия?

Удобно ли читать подписи? Красиво ли сочетаются цвета?

Добавьте «вау-эффект»: Звук переключения, плавную анимацию изменения высоты столбиков, интересный факт об элементе в подсказке.

**Этап 4. Презентация и защита. Доклад в Академии Наук**

Создайте презентацию в стиле научного открытия (4-5 слайдов):

Слайд 1: Вопрос. «Мы устали от скучных цифр в таблице. Мы решили дать химическим свойствам голос и цвет!»

Слайд 2: Методология — ключ к успеху. Покажите вашу таблицу с исходными данными и результатами расчёта масштабов. Скажите: «Чтобы

нарисовать эти столбики, нам пришлось превратить реальные величины в пиксели по этой формуле...». Это покажет вашу математическую работу.

Слайд 3: Живая демонстрация — ГЛАВНЫЙ ЭКСПЕРИМЕНТ! ПЕРЕЙДИТЕ ИЗ ПРЕЗЕНТАЦИИ ПРЯМО В ВАШ SCRATCH-ПРОЕКТ. Запустите его и в реальном времени покажите: «Смотрите, вот атомные массы. А теперь мы переключаемся на температуру плавления — и видим, что у галогенов она отрицательная!».

Слайд 4: Научный вывод. «Наша визуализация наглядно подтвердила, что в группе щелочных металлов с ростом атомного номера температура плавления падает, а у галогенов — резко возрастает от Фтора к Йоду».

Слайд 5: Команда и перспективы. «Нашу модель можно расширить, добавив все 118 элементов и новые свойства, например, электропроводность».

Распределите речь:

Химик: «Мы выбрали эти группы, потому что они демонстрируют противоположные тенденции...».

Математик: «Самым сложным было корректно масштабировать отрицательные температуры плавления...».

Программист: «Мы использовали клонирование, чтобы не создавать 10 одинаковых скриптов...».

Дизайнер: «Мы использовали холодные цвета для низких температур и тёплые для высоких, чтобы интуитивно было понятно...».

Будьте готовы к вопросам: «Почему вы не взяли плотность?», «Как вы учли, что размер экрана у всех разный?», «Можно ли сделать, чтобы программа сама считывала данные из файла?».

### **Ожидаемые результаты**

Выполняя проект, вы получите следующие продукты:

Рабочую модель в Scratch, представляющую собой интерактивную сравнительную визуализацию свойств химических элементов.

Презентацию, раскрывающую этапы работы и демонстрирующую итоговый продукт.

Результатом проекта также будет углубление понимания взаимосвязи между числовыми данными, их математической обработкой и визуальным представлением.