

Пояснительная записка к заданию

<https://znaniya.com/task/18661278>.

Помогите создать мультфильм для урока информатики 5 класс на любую тему .

Предварительные замечания.

Хочу сразу предупредить, я не профессиональный художник аниматор (и даже не художник), но в разное время некоторые простые опыты проводил. Рабочие программы в задании не оговорены, тут у нас полная свобода. Я приведу несколько примеров результатов, а также пару примеров «проектов» рассмотрю немного подробнее. Т. е. хочу показать, что простенькую анимацию по силам создать любому, было бы желание.

Создание полноценного мультфильма довольно затратное (по времени) дело. В комментариях пользователь уточнил, что короткая анимация его тоже устроит. В любом случае, приступая к работе, главное определиться, что мы хотим получить на выходе. Разработать сценарий. Чем более сложной будет ваша задумка, тем подробнее придётся прорабатывать сценарий. При этом желательно уметь рисовать (ну хоть что-нибудь). Можно также «оживлять» готовые картинки. Соответственно умениям и разрабатывать сценарий. Не стоит сразу браться за сложный проект. Есть смысл попробовать что-то попроще. Возможно просто отработать приёмы работы.

Анимация или мультипликация осуществляется быстрой сменой картинок (кадров), на которых изображены различные положения анимируемых объектов. Для демонстрации ограничимся 2D графикой. Т.е. нам понадобится графический редактор. Просто картинки можно нарисовать и в Paint. А потом отдельной программой свести нарисованные кадры. Вопрос только в простоте удобстве, и доступности выбранных программы.

Я, например, в ранних опытах использовал GIMP (прошу не плевать :)). Для простых задач программа годится, тем более она свободно распространяемая (пиратить ничего не нужно). Она умеет работать с многослойными изображениями, что помогает при прорисовке кадров, поддерживает довольно много форматов графических файлов. Можно, раскидав кадры по слоям, выдать анимированный gif файл. А с дополнением GAP можно и видео (хоть и хлопотно). Однако это растровый редактор. В нем можно масштабировать, вращать, двигать части изображения, но некоторые огрехи при этом неизбежны. В сети встречалось предложение использовать в работе комбинацию растрового (GIMP) и векторного (Inkscape) редакторов. Простые кадры создаются в Inkscape . А затем загоняются слоями в GIMP.

Решения я приведу в общих чертах: идеи + результаты. Полагаю, что рисовать с помощью компьютера вы уже пробовали и сумеете разобраться с программа,

РЕШЕНИЕ

Итак Анимация или мультипликация осуществляется быстрой сменой картинок (кадров). Определяемся со сценарием. Например, хочу растущий подсолнух. Краткое словесное описание. В поле появляется росток, тянется, пускает листья, затем появляется и набухает бутон, раскрывается желтым цветком. Затем цветок растёт, поворачивается, его серединка из зелёно-жёлтой становится черной. Вот ранняя реализация данной задумки в GIMP. Кадры создавались один за другим на отдельных слоях. Четкого расписания по времени тут нет. Создаем анимацию длительностью примерно 3 секунды. Вначале создаём фон. (рисунок 3).

Затем на отдельных слоях прорисовываем положения ростка, изменения фона (трава зеленеет, меняется облачность).



Рисунок 1: Фон. Он же будет 1-м кадром.

Для плавной анимации число кадров в секунду должно быть как можно больше. Но тогда растёт объём работ (и файлов :). Необходимо разумное ограничение. В кино, помнится было принято 24 кадра в секунду. Ну для учебной мы ограничимся 10 кадрами в секунду. (Будут заметны рывки. Но зато на 3 секунды нам понадобится всего 30 кадров.)

Рисуем остальные кадры (рисунки 2, 3, 4). Порядок сменения кадров: слева направо, сверху вниз. Тут масштаб кадров строго не соблюлён. Смотрите исходный файл *flower22.xcf* (Это родной формат GIMP) и результат *flower22.gif* анимашка. Коротко и не мудрёно, но надеюсь суть ухватили.

Смотрите ещё примеры (только результаты). В папке **ДОЖДЬ** файл *Rain.gif*, был создан по аналогичной «технологии». Подсолнух и дождь быстрые, топорные зарисовки создавались за вечер. Естественно точно кадры не просчитывались. Поэтому что выросло, то выросло.

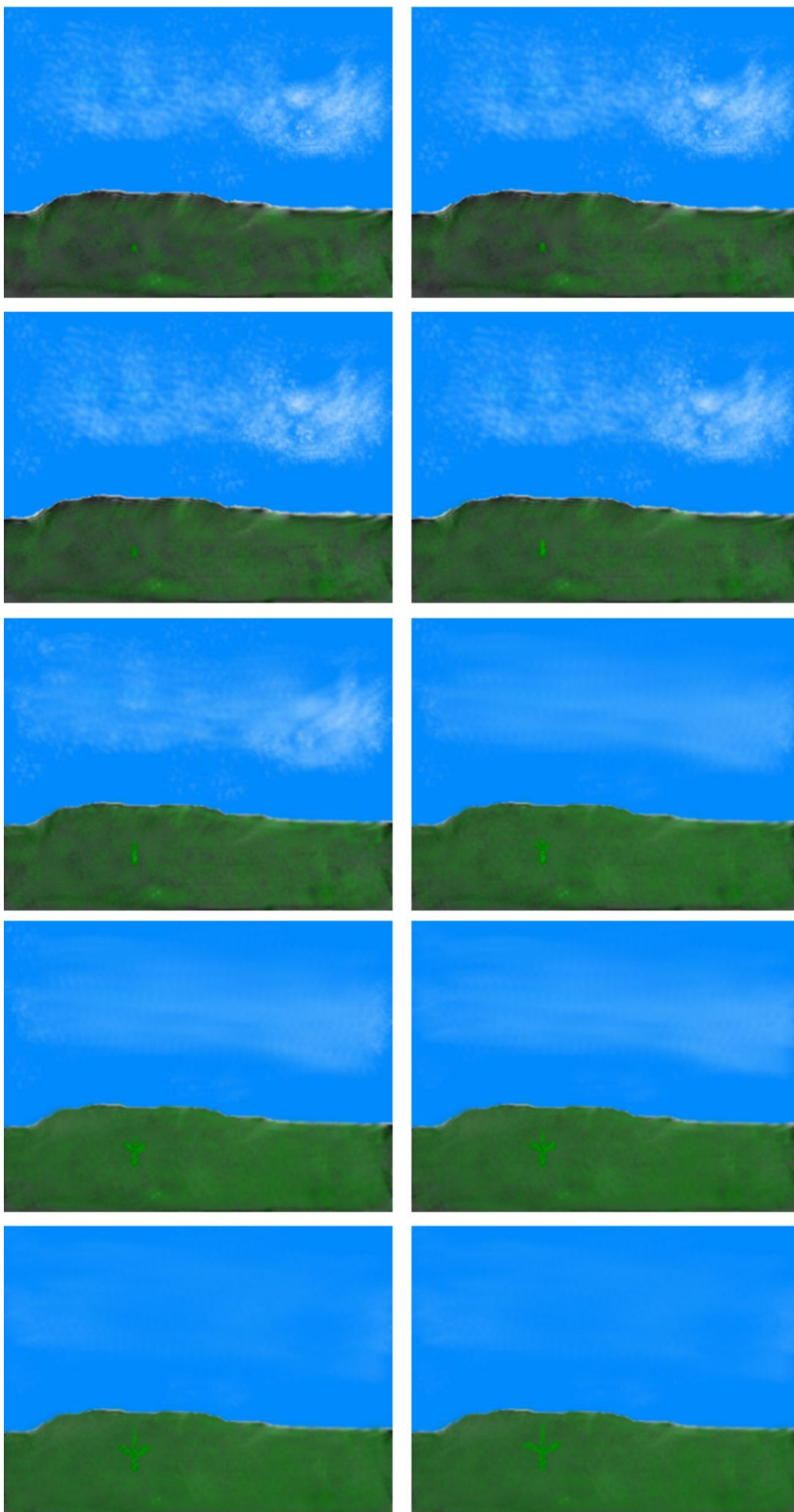


Рисунок 2: Кадры 2 -11. Проклюнулся росток, тянется и пускает листья.

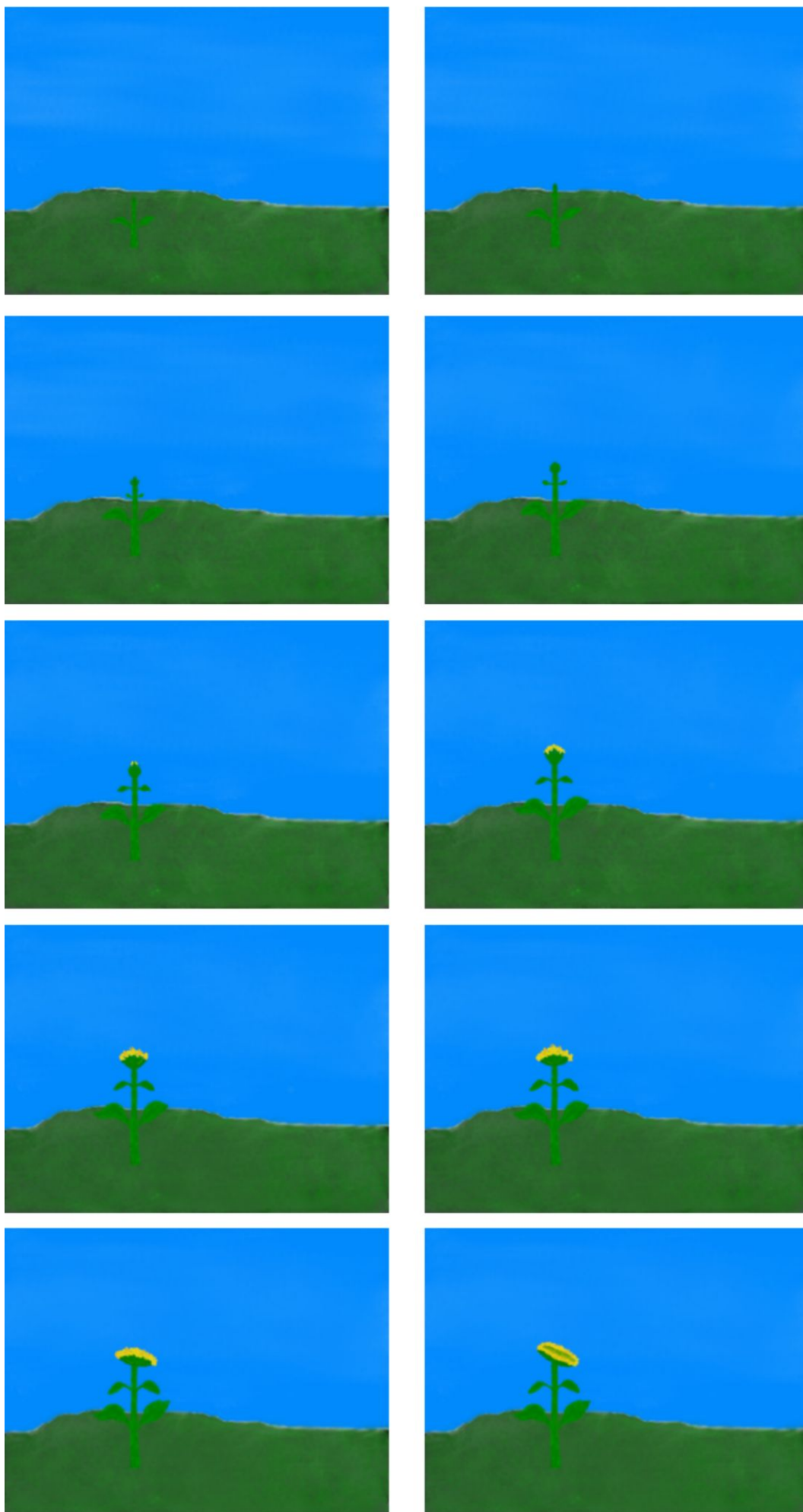


Рисунок 3: Кадры 12-22. Появляется и раскрывается бутон.

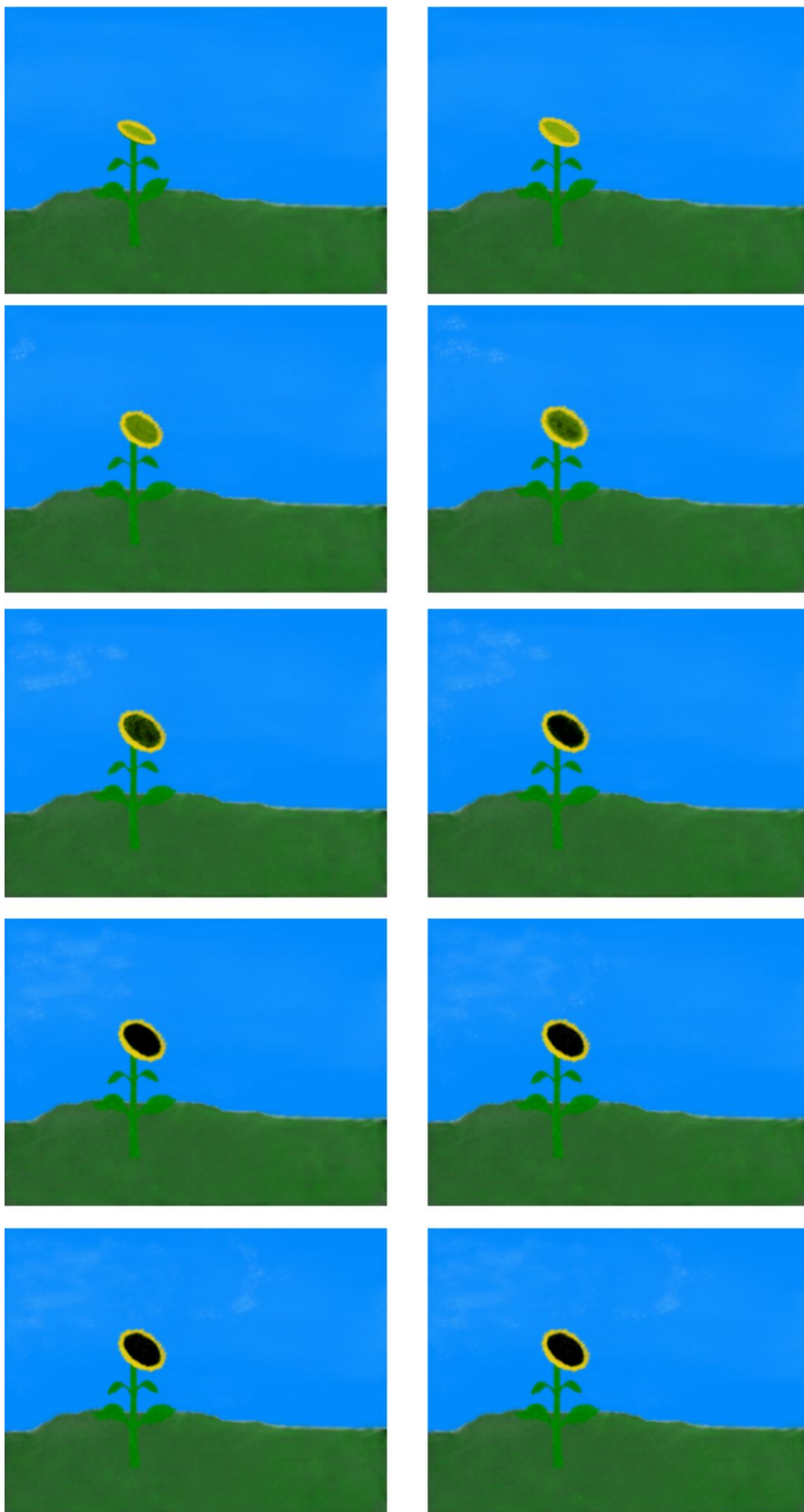


Рисунок 4: Кадры 22-31. Подсолнух зреет. На последних 4х кадрах созрел.

В папке **Вертолёт** более сложный пример *Helicop.gif*. Тут уже кадры рисовались в векторном редакторе Inkscape, выгружались в отдельные файлы кадров. А затем объединялись GIMP. Тут возни уже изрядно. Особенно порадовали винты. Пришлось рассчитывать положения винтов при раскрутке и положения корпуса при взлёте и повороте.

В трёх предыдущих примерах, нам приходилось прорисовывать каждый кадр. Да мы могли использовать во вновь создаваемом кадре часть изображения от готовых кадров. И всё таки довольно хлопотно это. Современные средства для работы с анимацией позволяют сократить число кадров, прорисовываемых «вручную». И упростить этот процесс. Например, техника «ключевых кадров» позволяет пользователю задавать только отдельные («ключевые») положения движущегося объекта, а требуемое число промежуточных кадров, программа рассчитывает сама.

В качестве примера приведу программу Synfig Studio (опять же она свободная). Она ещё в разработке. Возможно, не до конца «вызрела», но уже кое-что она умеет. В программе можно создавать векторные фигурки объекты, а затем заставлять их двигаться, менять форму цвет и размеры. Программа тоже может загрузить растровые изображения. Набор доступных операций с растровыми картинками не столь богат, но подвигать картинки можно.

В папке Поезд исходные картинки, проект для Synfig Studio версии 1.0.2 (Чтоб открыть требуется версия не младше указанной) и его результаты *Trains.mp4* или *Trains.gif*. Тут видео создавалось с плотностью в 25 кадров в секунду. При этом на 10,8 секунд видео вместо прорисовывания 271 кадра мы фактически обошлись 8-ю ключевыми кадрами. Условные их названия смотрите в таблице 1. Эта таблица уже представляет собой некоторое подобие сценария. Смотрите также список ключевых кадров в проекте и комментарии к ним. Буквы f после цифр означают, что длительность действий, в фильме мы измеряем в кадрах.

Порядковый № ключевого	№ кадра в «фильме»	«Длительность»	Комментарии
1	0 f	25 f	Дальний поезд стоит. Ближний стоит за кадром и не виден. Ждём-с.
2	25 f	13 f	Дальний поезд тронулся
3	38 f	13 f	Дальний ускорился 1
4	50 f	16 f	Дальний ускорился 2 (скорость возросла).
5	66 f	114 f	Ближний поезд тронулся.
6	180 f	70 f	Дальний ушёл в «конечную точку» за кадр.
7	250 f	20 f	Ближний поезд ушёл в «конечную точку» за кадр.
8	270 f	0f	Пауза

Таблица 1: Ключевые кадры проекта Поезда

В данном проекте фон нарисует в самом Synfig Studio, а поезда импортируем как растровые картинки. «Слоистая структура» позволяет разместить объекты так, чтобы имитировать различное удаление их от зрителя. На переднем плане разместим пару деревьев, затем насыпи с путями для «ближнего поезда» и для «дальнего поезда». На заднем плане разместим ещё одно дерево и здание. Смотрите рисунки 5, 6, 7.

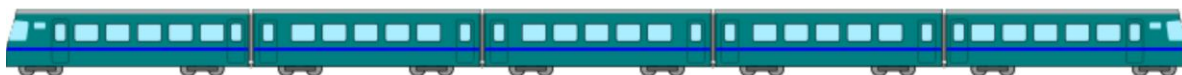


Рисунок 5: Дальний поезд.



Рисунок 6: Ближний поезд.

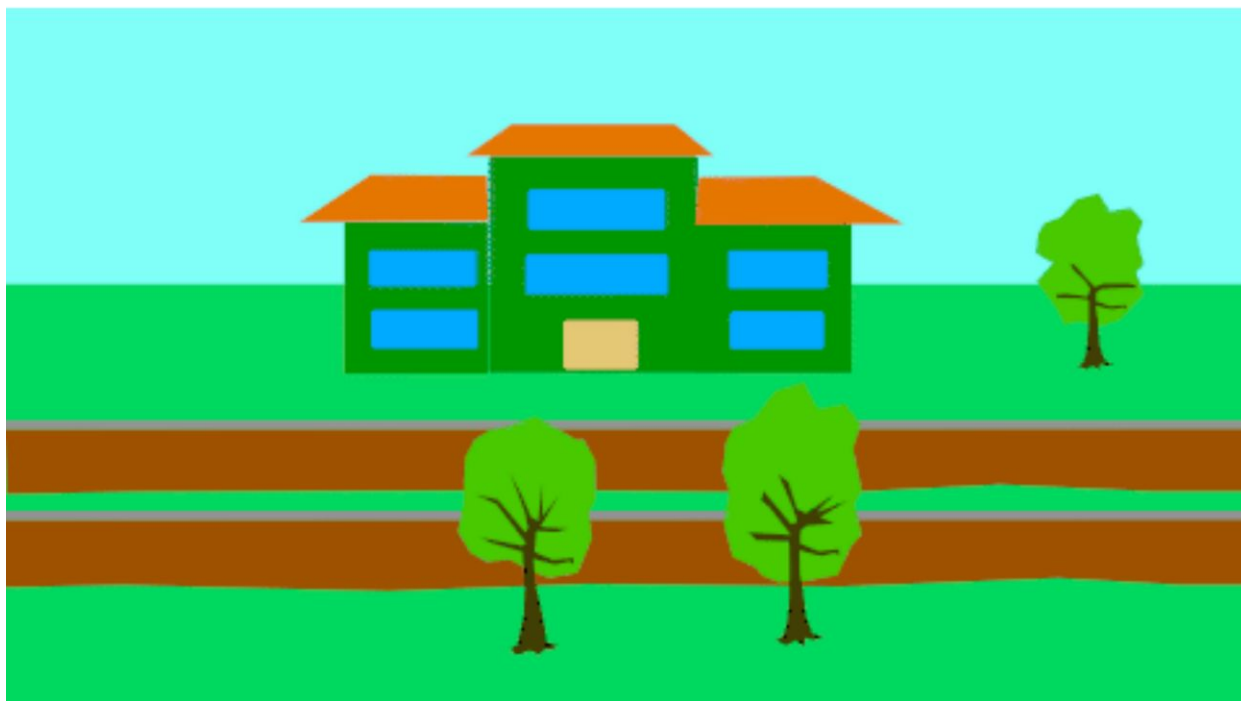


Рисунок 7: Фон для проекта "Поезда".

Строго говоря, можно было обойтись и меньшим число ключевых кадров, например 2-мя: «начальное положение» и «конечное положение». Но мы хотим получить более «естественное» движение, а потому будем следовать таблице 1. Рассмотрим вкратце, что мы размещаем (двигаем) на первых «ключевых» кадрах, и что будет происходить на экране.

0f. Это исходное положение. На этом кадре помещаем дальний поезд (с синей полосой) на 2й путь так, чтоб «голова» поезда выглядывала из-за левого края кадра. Ближний поезд ставим на свой путь, а затем, смещаем его по горизонтали так, чтобы задвинуть его за правый край кадра.

25f. Просто копируем исходный кадр. При расчёте анимации программа сама разместит между нулевым и 25-м кадром ещё 24 копии. Таким образом. В течение 26 кадров (1й секунды) у нас на экране ничего не происходит.

38f. На этом кадре немного сместим дальний поезд вправо. При расчёте анимации программа разместит между этим и 25-м кадром кадры с промежуточными положениями. Таким образом, начиная с кадра 26f дальний поезд начнёт двигаться и на кадре 38f попадёт в заданное нами положение.

50f. Сместим дальний поезд ещё правее, причем так, чтобы расстояние между положениями поезда на кадрах 38f 50f, было заметно больше расстояния между положениями 25f 38f. Когда программа рассчитает промежуточные кадры, при анимации получится, что между кадрами 38f 50f поезд будет двигаться быстрее, чем между предыдущей парой ключевых кадров.

Аналогичным образом мы можем разогнать ближний поезд, «протянуть» его через кадр и заставить его скрыться за левым краем кадра.