**МИНОБРНАУКИ РОССИИ**

**ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМ. Р.Е.АЛЕКСЕЕВА»**

**(НГТУ)**

Институт радиоэлектроники и информационных технологий

Кафедра «Графические информационные системы»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к курсовой работе

по дисциплине «Графические информационные технологии»

|  |  |
| --- | --- |
| РУКОВОДИТЕЛЬ  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись)  СТУДЕНТ  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (дата)  Работа защищена  С оценкой | \_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О.)  \_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О.)  \_\_\_\_\_\_\_\_\_\_\_  (группа)  \_\_\_\_\_\_\_\_\_\_\_  (дата)  \_\_\_\_\_\_\_\_\_\_\_ |

|  |
| --- |
|  |

Нижний Новгород, 2020

**НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ**

**УНИВЕРСИТЕТ ИМ. Р.Е.АЛЕКСЕЕВА**

**(НГТУ)**

Кафедра «Графические системы и технологии»

ЗАДАНИЕ

на курсовое проектирование

студенту \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Тема курсового проекта: Моделирование роста травы.

2. Исходные данные к проекту: изображение травы.

3. Срок сдачи студентов законченной курсовой работы: 22.05.2020

4. Содержание графического материала:

файл в формате:

1. travaForWork.blend – заготовочный файл для создания кадров с моделью травы.

5. Содержание пояснительной записки:

1. титульный лист,
2. список использованных источников

6. Основная рекомендуемая литература:

1. Руководство по использованию blender API.   
   <https://docs.blender.org/api/current/index.html>
2. Дата выдачи задания: 22.04.2020

|  |
| --- |
| Руководитель: \_\_\_\_\_\_\_\_ |
| (подпись) |
| Задание принял к исполнению: \_\_\_\_\_\_\_\_\_ |
| (подпись) |

**СОДЕРЖАНИЕ**

[1. Создание модели травы. 2](#_Toc40790354)

[Моделирование 2](#_Toc40790355)

[Материалы 9](#_Toc40790356)

[Моделирование 11](#_Toc40790357)

[2. Настройка заготовочного файла. 14](#_Toc40790358)

[Моделирование 15](#_Toc40790359)

[Программирование 21](#_Toc40790360)

[3. Написание скрипта отрисовки травы (использование Blender API). 24](#_Toc40790361)

[Программирование 24](#_Toc40790362)

[4. Написание программы, которая определяет, где и какая трава будет отрисована. (будет усовершенствована) 27](#_Toc40790363)

[Программирование 27](#_Toc40790364)

[5. Запуск всей работы. 30](#_Toc40790365)

[Программирование 30](#_Toc40790366)

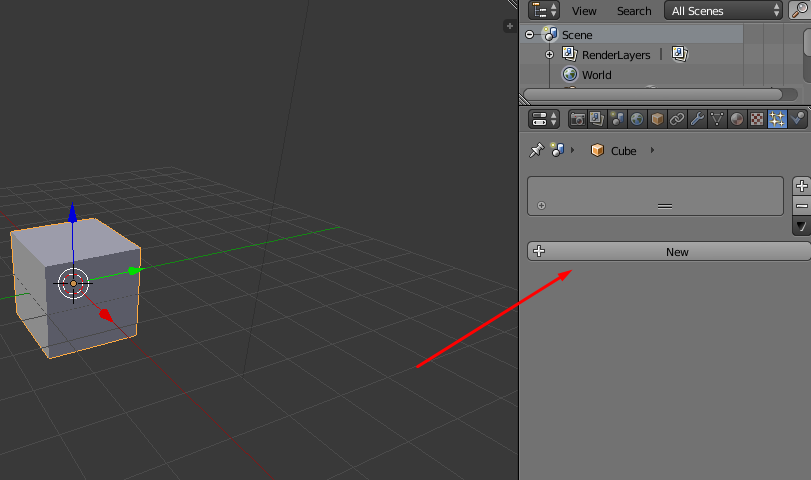
# 1. Создание модели травы.

Трава создана по видео <https://youtu.be/K0hoV6ZSUEI>.

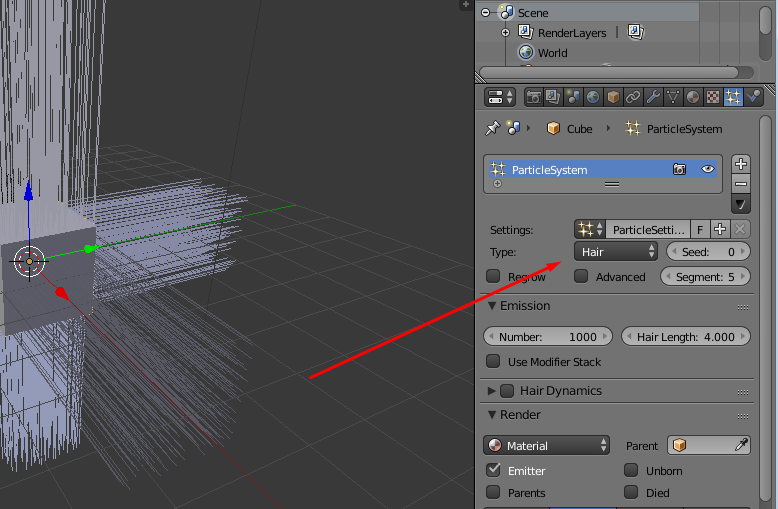
Создание файла trava.blend.

Порядок работы:

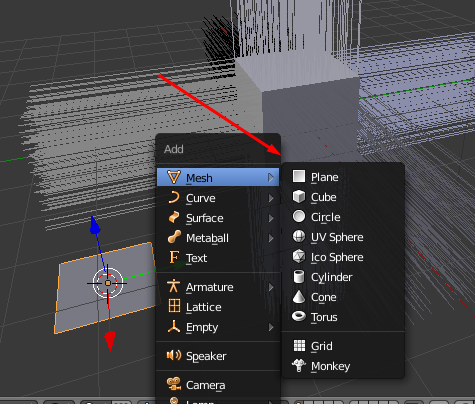
## Моделирование

1. Создать новый файл.
2. Добавим дефолтному кубу систему частиц: Вкладка Particle system (Система частиц) -> New  
   

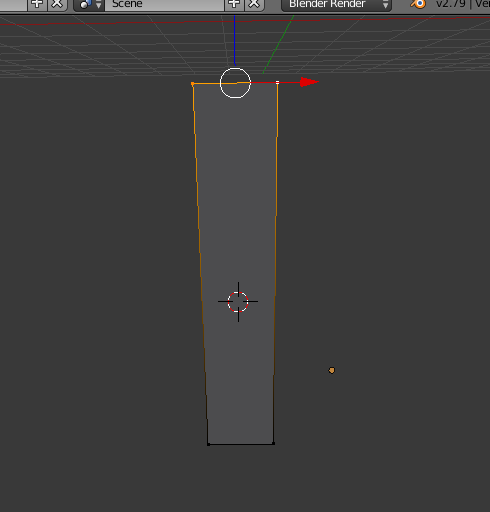
Рисунок

1. Вместо Emitter выберем Hair. (Type – Hair)

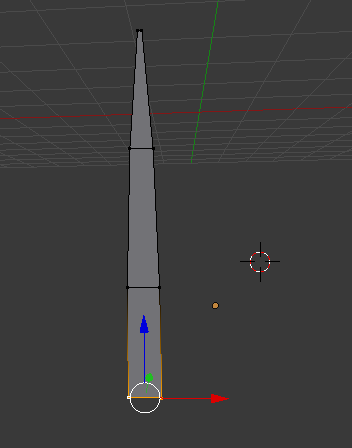
Рисунок

1. Создадим (c помощью Shift + A) Plane рядом с кубом: Mesh -> Plane

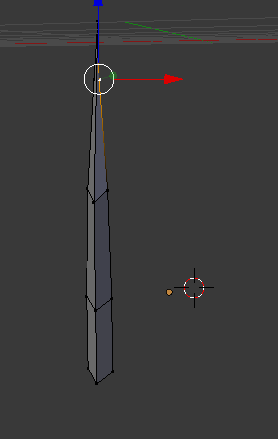
Рисунок

1. Перейдём в Edit mode, нажмём / для изолированного редактирования. Придаём plane форму травинки. Двигаем точки и получаем 

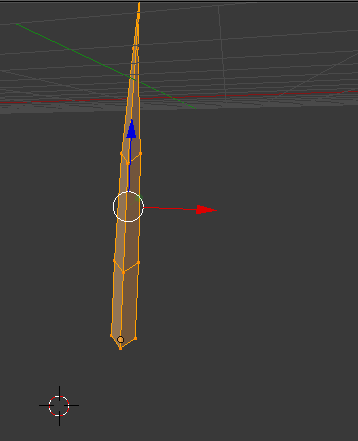
Рисунок

1. Дважды делим с помощью Crtl + R, двигаем точки, масштабируем получившиеся части plane после деления Ctrl + R и получаем   
   

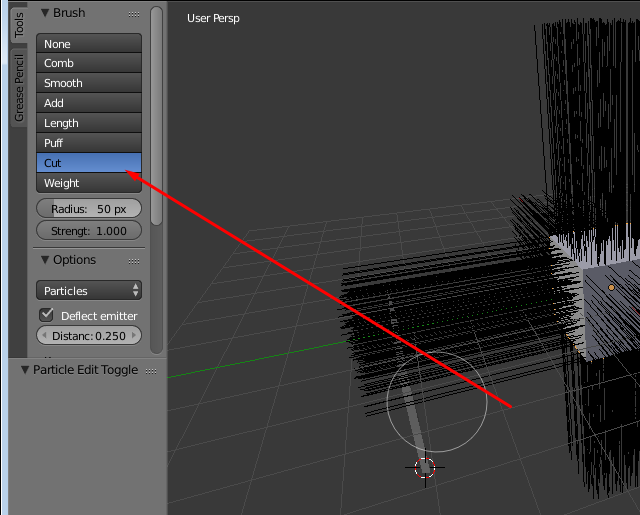
Рисунок

1. Делим посередине с помощью Ctrl + R и получаем ребро, двигаем его по Y. Можно также скорректировать вершины   
   

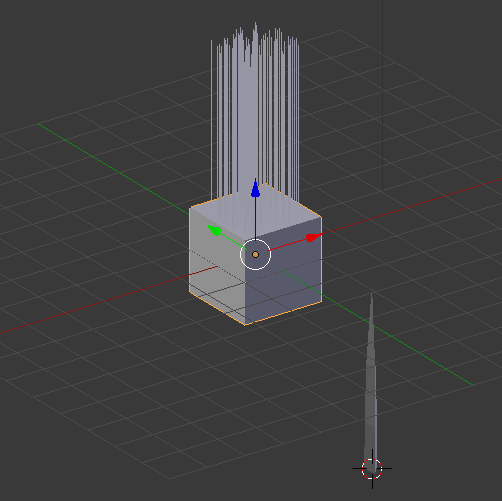
Рисунок

1. Перемещаем объект в оранжевой точке – логическому центру объекта, чтобы отснование травинки и точка примерно совпадали. Весь объект можно выделить с помощью A. 

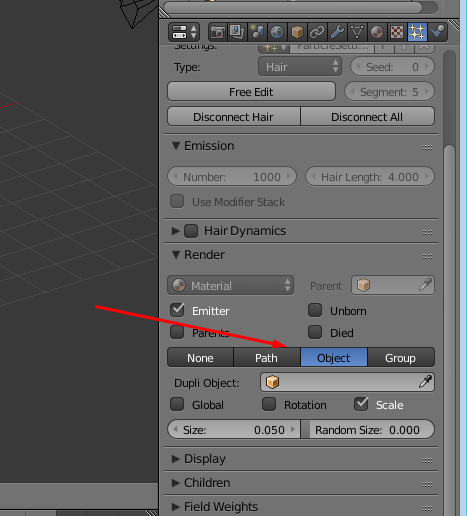
Рисунок

1. Переходим обратно в object mode и нажимаем /, чтобы увидеть куб. Выделяем куб и где Object mode выбираем Particle edit. И выбираем инструмент Cut  
   

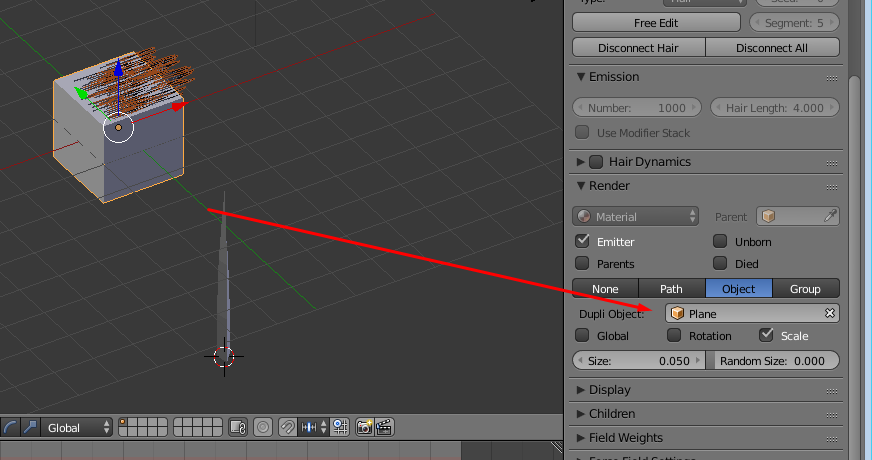
Рисунок

1. Удаляем лишние “волосы”, учитывая, что они будут травинками. Нажмём Z, чтобы стирать насквозь. Тщательно убрать со всех граней кроме верхней, даже можно убртать немного с краёв верхней. Если стёрли лишнее, то можно вернуть их с помощью инструмента Add. Возвращаемся в Object mode.  
   

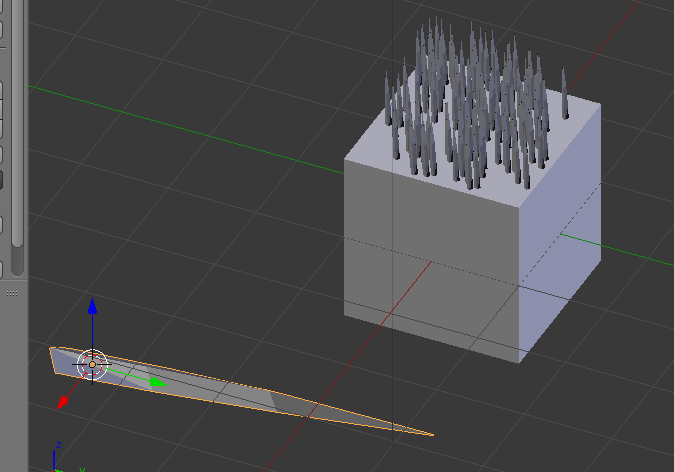
Рисунок

1. На вкладке Particles включаем пункт Object.  
   

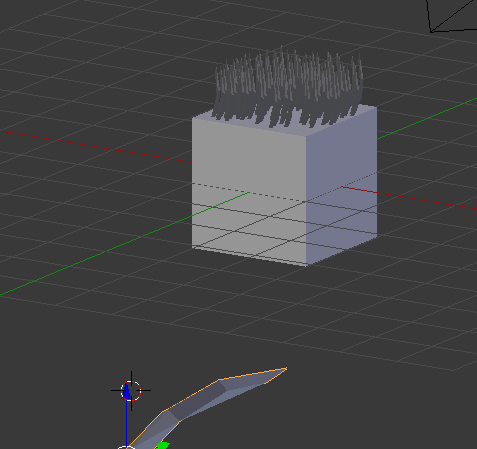
Рисунок

1. В Dupli Object ставим нашу травинку, то есть объект plane. ( Dupil object - Plane)  
   

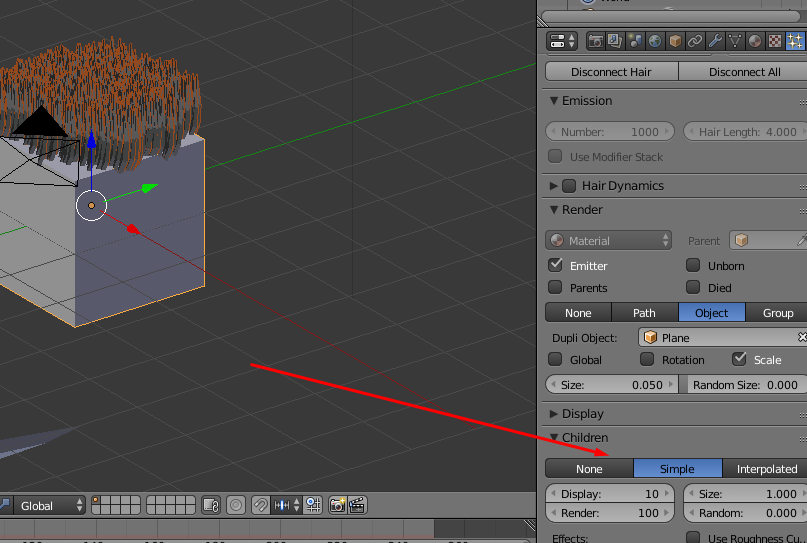
Рисунок

1. Чтобы выровнять травинки, выберем одну травинку plane и в Edit mod вращаем её и перемещаем.  
   

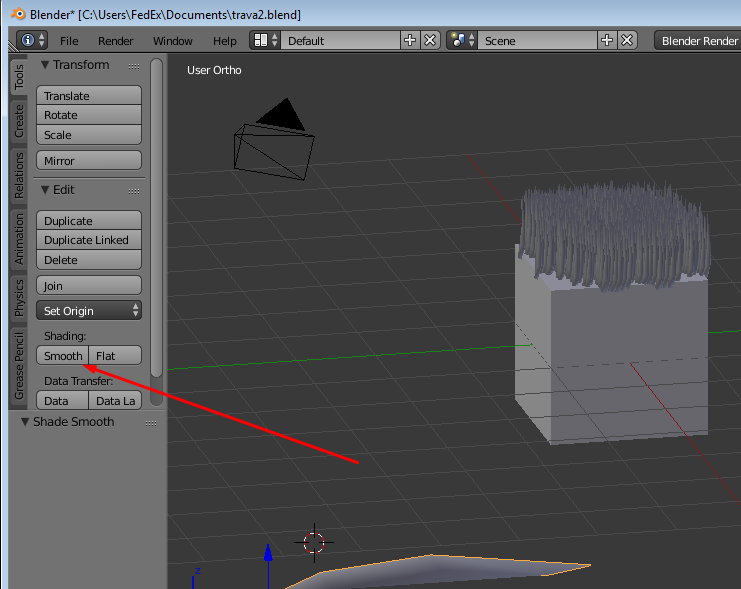
Рисунок

1. Чтобы придать траве более интересную форму выделим в Edit mode несколько вершин по середине травинки и перетащим пониже, чтобы изогнуть травинку. Снять выделение всех точек – А, прозрачное выделение - Z, выделить несколько вершин – клавиша B.  
   

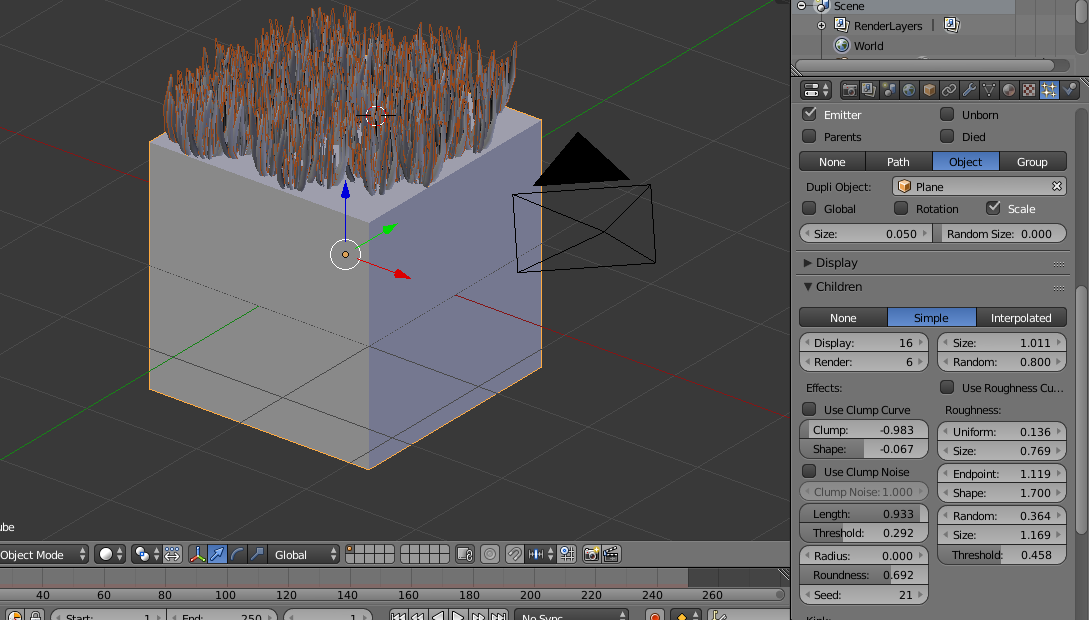
Рисунок

1. В Object mode выделяем куб, переходим во вкладку Children, нажимаем кнопку Simple.  
   

Рисунок

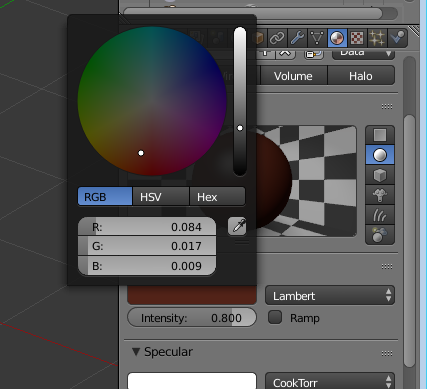
1. Используем ползунки Size – для настройкаи размера травы, Random – для настройки рандомности размера, Display – для количества отображаемых в blender, Render - для количества отображаемых после рендера на картинке. EndPoint – чтобы поворачивать. Можно также вернуться в Edit mode и опять повернуть травинку для поворота все травы. Чтобы травинка была более гладкой можно в Object mode включить Shading Smooth  
   

Рисунок

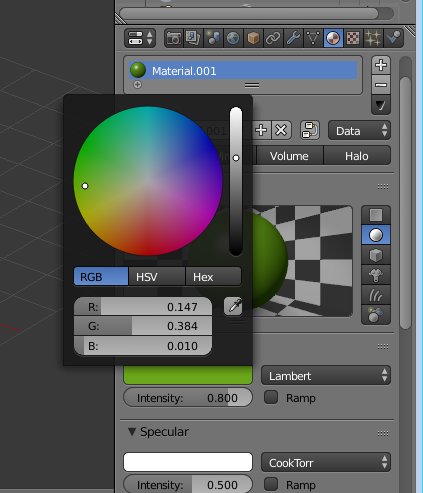
1. Получем что-то вроде этого  
   

Рисунок

## Материалы

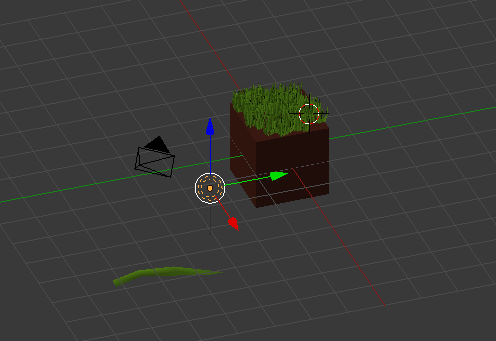
1. Придадим материл кубу похожий на землю.  
   

Рисунок

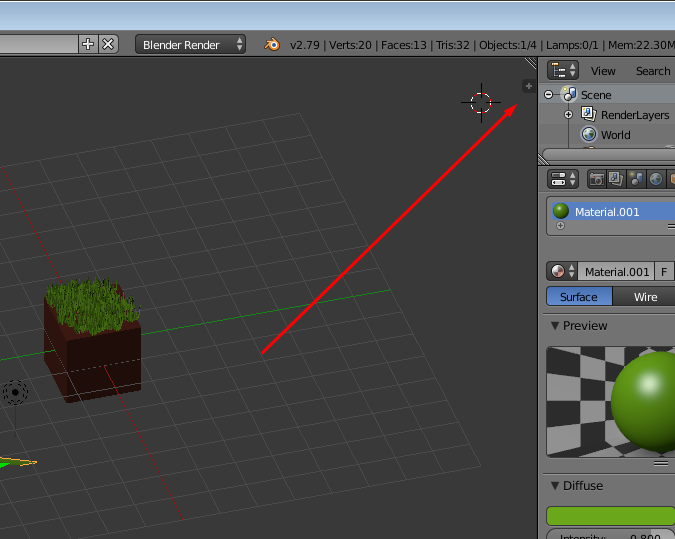
1. Травинке Plane –новый матераил - зелёный.  
   

## Моделирование

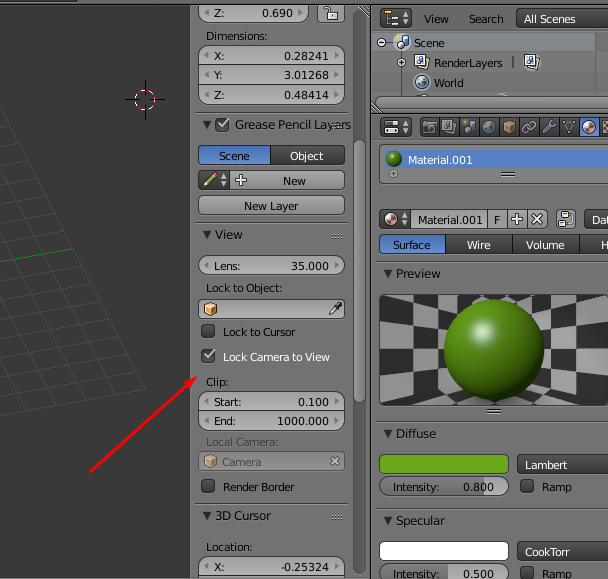
Рисунок

1. Поставим свет, передвинув источник  
   

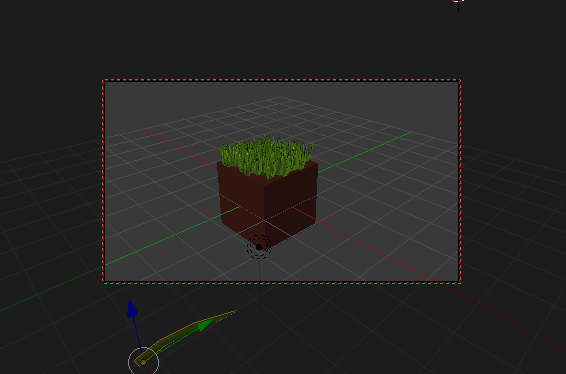
Рисунок

1. Поставим камеру с помощью Lock Camera to View  
   

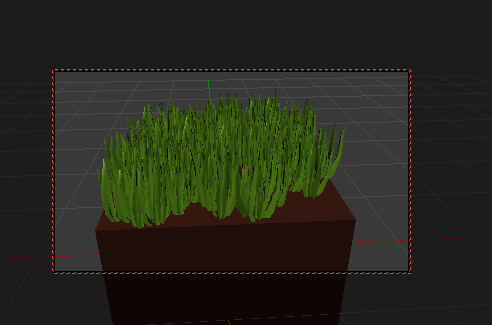
Рисунок

1. 

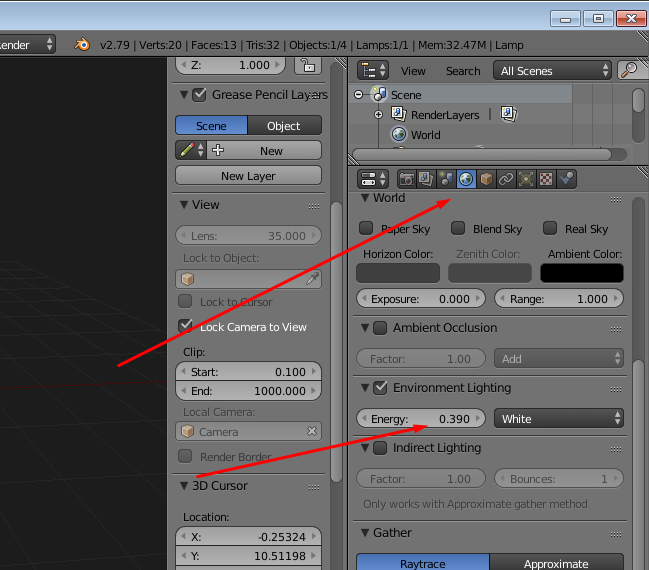
Рисунок

1. Потом NumPad 0.  
   

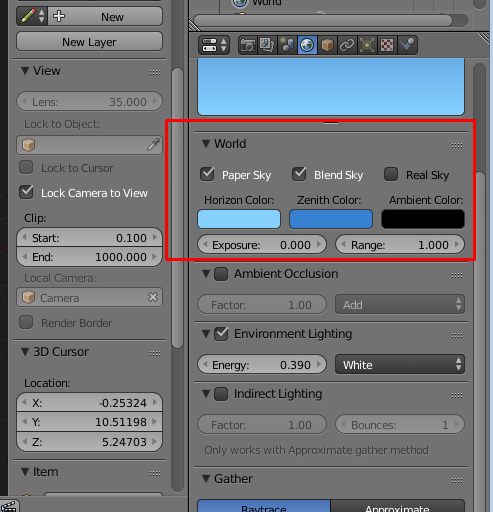
Рисунок

1. И перемещаемся в помощью мыши, как при обычном виде.  
   

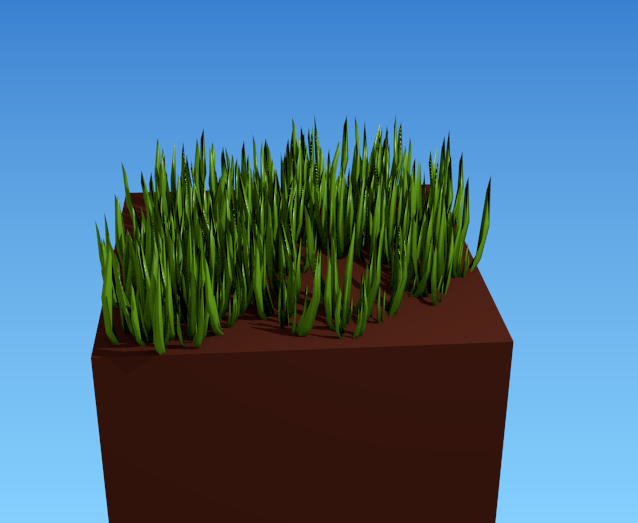
Рисунок

1. Чтобы улучшить картинку во вкладке World включим Environment lighting (Освещение среды) с Energy - 0.4 примерно.  
   

Рисунок

1. Также можно включить небо в соответствии и настройками на скрине.  
   

Рисунок

1. Отрендерим на F12 и готово. Можно сохранить в файл trava.blend.  
   

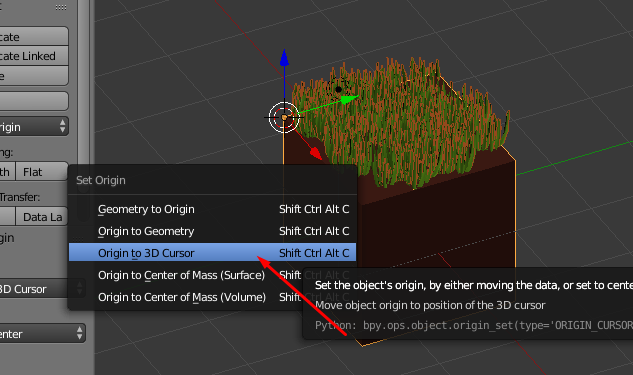
Рисунок

# 2. Настройка заготовочного файла.

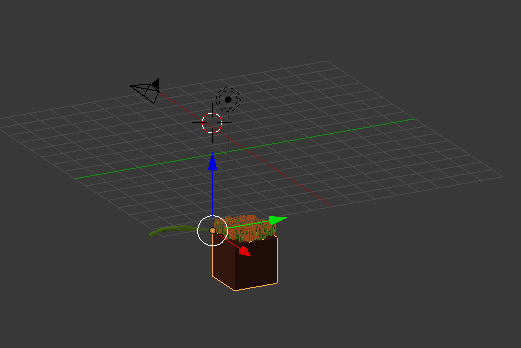
Порядок создания файла travaForWork.blend.

Порядок работы:

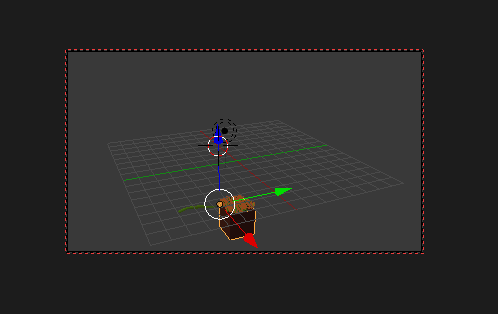
## Моделирование

1. Открыли файл trava.
2. Установили 3Д курсор на левом верхней вершине куба, нажали ctrl+shift+alt+C, потом Origin to 3D Cursor.  
   

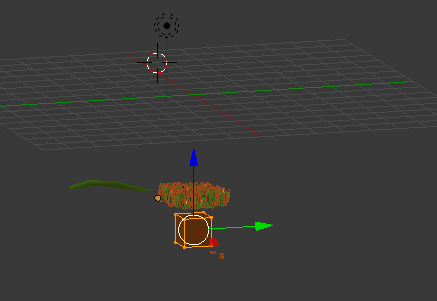
Рисунок

1. Переместили куб вниз по Z.  
   

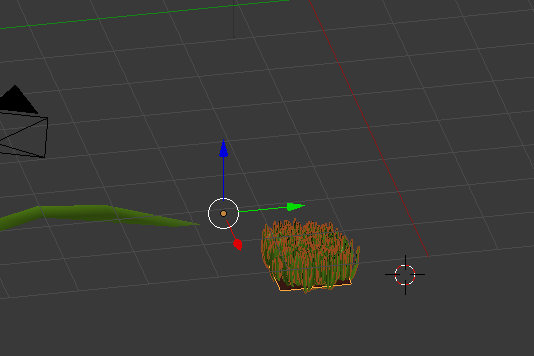
Рисунок

1. Настроили камеру согласно скрину по пункту 20 прошлой части.  
   

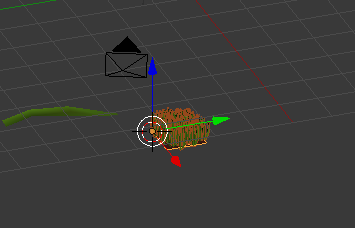
Рисунок

1. Перешли в Edit Mode, нажмём A два раза, чтобы выделить все вершины, нажали S, с зажатым Ctrl для точности, уменьшили куб в 2 раза (Scale 0.5 по всем осям), чтобы куб соответствовал одной клетки координат.  
   

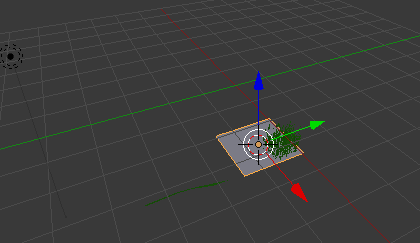
Рисунок

1. Уменьшили блок земли до максимально плоского варианта. В Edit Mode сняли выделение с помощью A, выделили нижнюю грань и переместили наверх.  
   

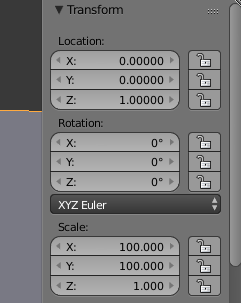
Рисунок

1. Поправим центр как в пункте 2 этой части.  
   

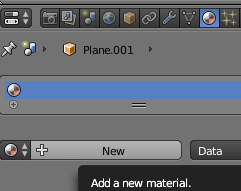
Рисунок

1. Создаём плоскость. Shift + A -> Mesh -> Plane  
   

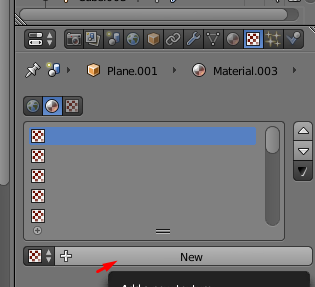
Рисунок

1. Задаём параметры. Location: X: 0 Y: 0 Z:1. Scale: X: 100 Y:100 Z: 1.  
   

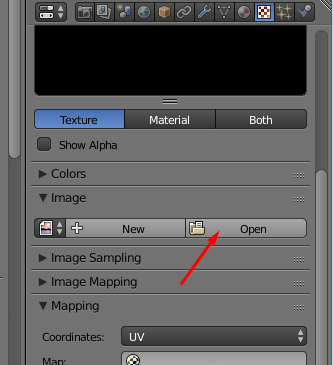
Рисунок

1. Добавим материал, нажав на кнопку New вкладки Materials.  
   

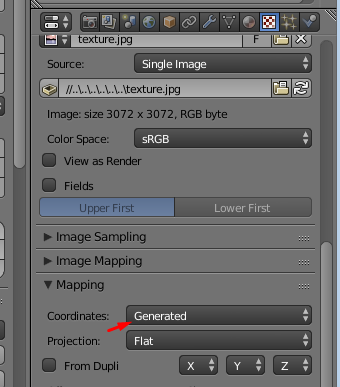
Рисунок

1. Добавим текстуру, нажав на New вкладки Textures.  
   

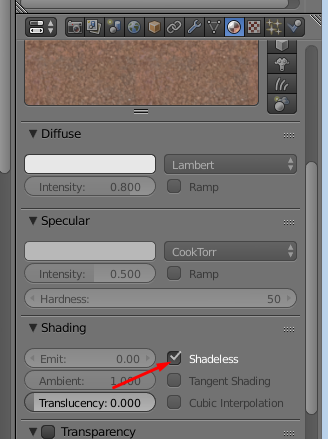
Рисунок

1. Добавим скачанную текстуру, нажав Open в свитке Image.  
   

Рисунок

1. В свитке Mapping параметр Coordinates: Generated.  
   

Рисунок

1. Включим параметр Shadeness.  
   

Рисунок

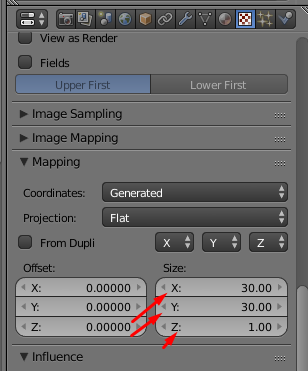
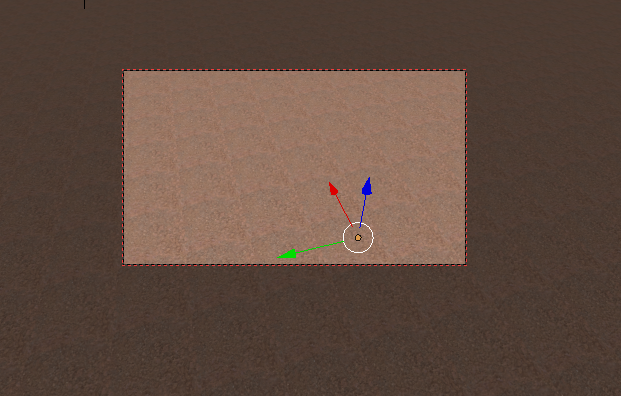
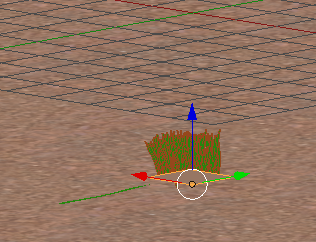
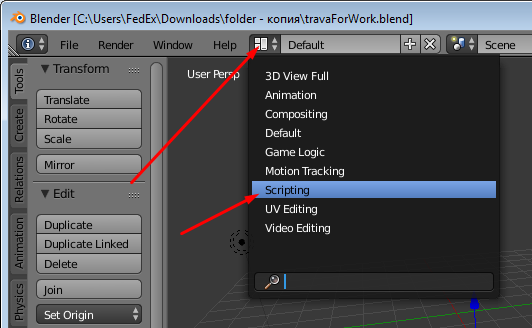
1. Настроим параметры Size во вкладке Textures: X:30 Y:30 Z:1.  
   

Рисунок   


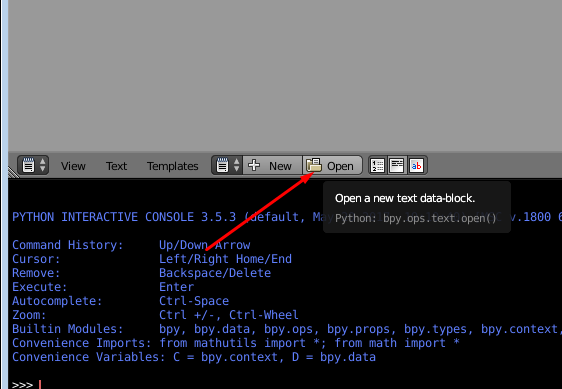
1. Выделим модель травы.  
   

Рисунок

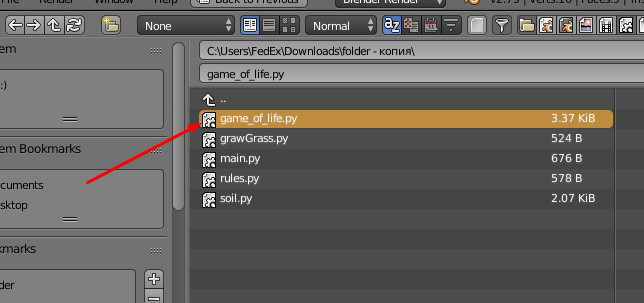
## Программирование

1. Добавим все исходный файлы скрипты в этот blender файл, чтобы потом выполнить всю программу.
2. Меняем среду на с Default на Scripting  
   

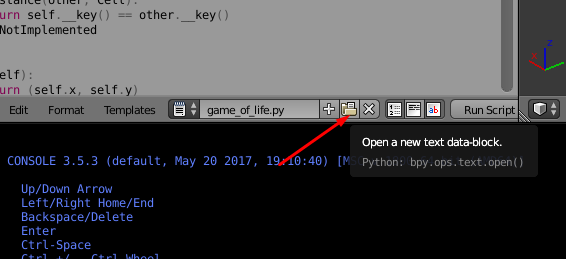
Рисунок

1. Нажимаем на значок добавления файла  
   

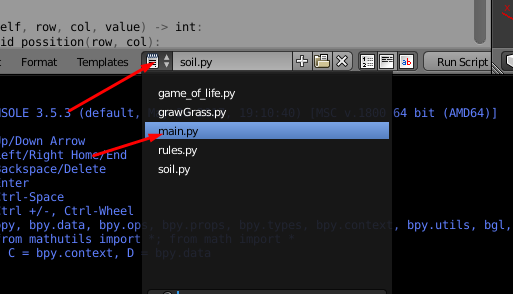
Рисунок

1. Двойной клик по game\_of\_life.py  
   

Рисунок

1. Потом опять значок добавить файл  
   

Рисунок

1. И так с каждым файлом .py в папке.
2. Открываем файл main.  
   

Рисунок

1. Всё. Сохраняем файл. Файл готов для запуска в нём скрипта.

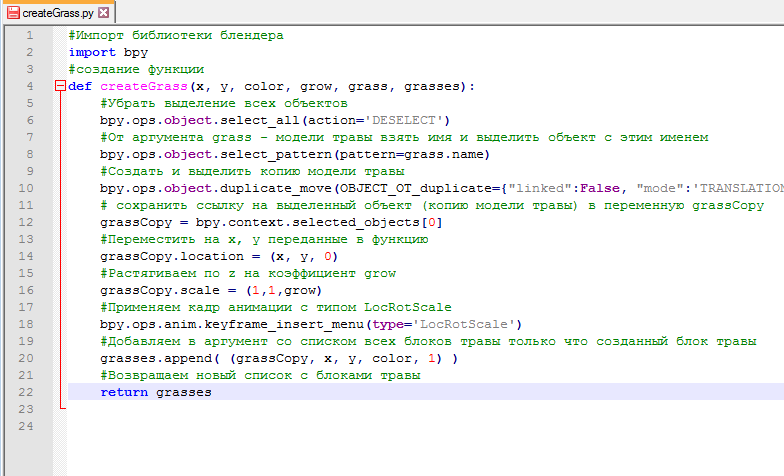
# 3. Написание скрипта отрисовки травы (использование Blender API).

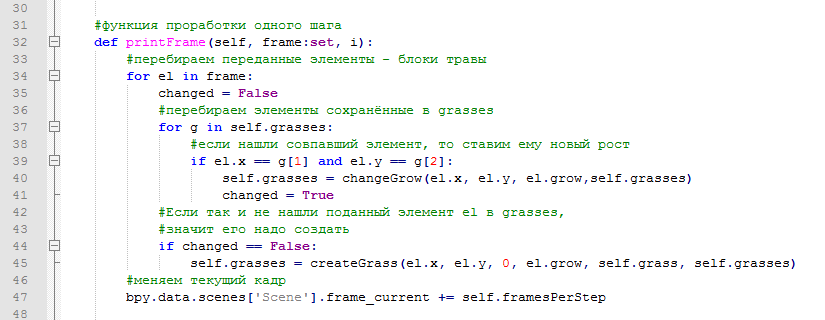
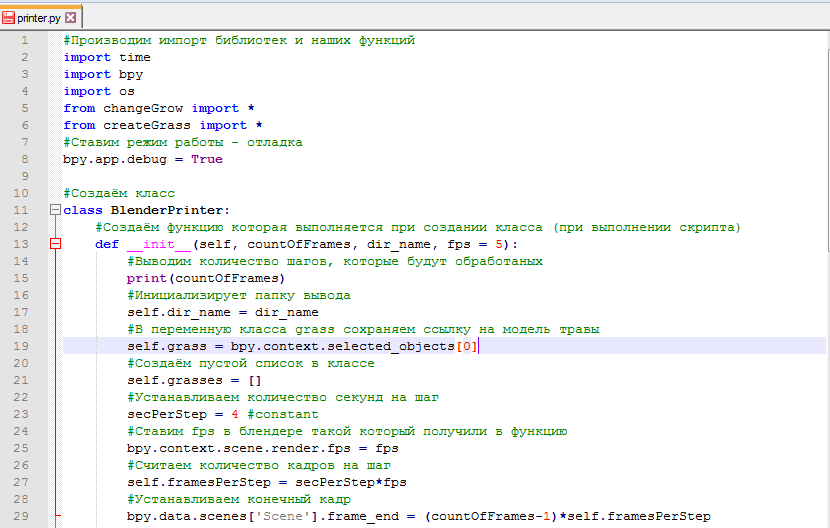
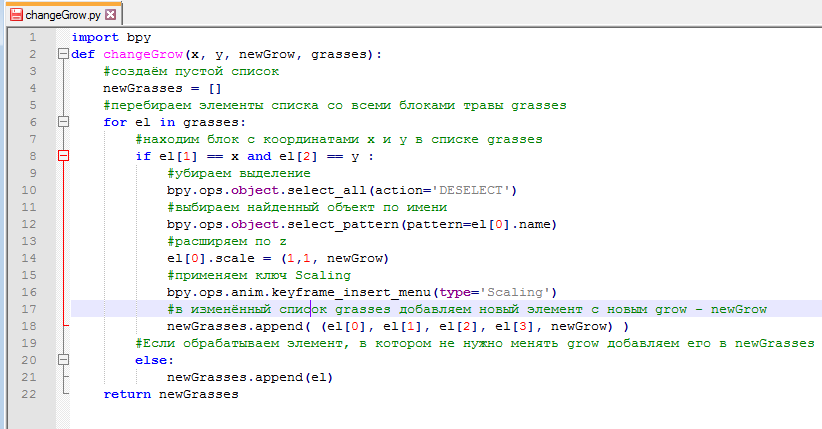
Создание файлов createGrass.py и changeGrow.py и blenderPrinter.py

Порядок работы:

## Программирование

Изучили Blender API и написали Python – код:





# 4. Написание программы, которая определяет, где и какая трава будет отрисована. (будет усовершенствована)

## Программирование

Модель поведения травы основана на алгоритмах клеточных автоматов. В этой работе за основу взят «*Conway's Game of Life»* правила, которой приведены ниже.

*Правила*

Место действия этой игры — «вселенная» — это размеченная на клетки поверхность или плоскость — безграничная, ограниченная, или замкнутая (в пределе — бесконечная плоскость). Каждая клетка на этой поверхности может находиться в двух состояниях: быть «живой» (заполненной) или быть «мёртвой» (пустой). Клетка имеет восемь соседей, окружающих её.

* Распределение живых клеток в начале игры называется первым поколением. Каждое следующее поколение рассчитывается на основе предыдущего по таким правилам:
  + в пустой (мёртвой) клетке, рядом с которой ровно три живые клетки, зарождается жизнь;
  + если у живой клетки есть две или три живые соседки, то эта клетка продолжает жить; в противном случае, если соседей меньше двух или больше трёх, клетка умирает («от одиночества» или «от перенаселённости»).
* Игра прекращается, если

на поле не останется ни одной «живой» клетки,

конфигурация на очередном шаге в точности (без сдвигов и поворотов) повторит себя же на одном из более ранних шагов (складывается периодическая конфигурация).

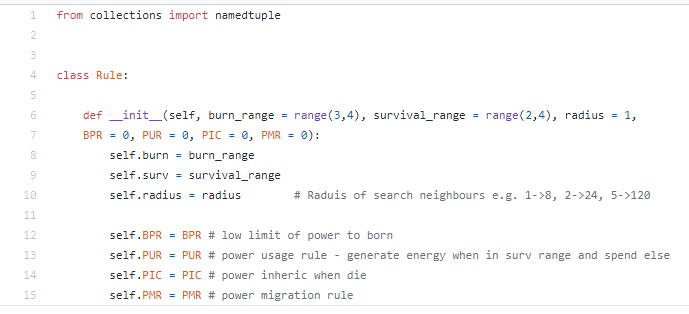
* + при очередном шаге ни одна из клеток не меняет своего состояния (складывается стабильная конфигурация; предыдущее правило, вырожденное до одного шага назад).

Эти простые правила приводят к огромному разнообразию форм, которые могут возникнуть в игре.

Игрок не принимает прямого участия в игре, а лишь расставляет или генерирует начальную конфигурацию «живых» клеток, которые затем взаимодействуют согласно правилам уже без его участия (он является наблюдателем).

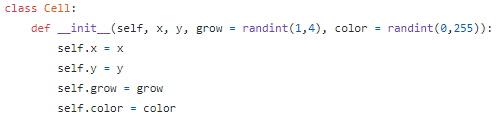
Теперь необходимо написать программу, которая будет не только симулировать данные правила, но и легко модифицироваться.

1. Для начала создадим модуль с правилами, чтобы иметь возможность менять количество соседей для рождения и смерти клетки, радиус, и добавим свои правила.



Рисунок

Первые три параметра отвечают за классический набор правил, а остальные отвечают за почву. Но о почве в следующей работе.

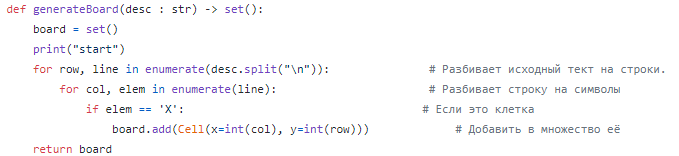
2. Потом нужна клетка травы, для которой будут применены вышеупомянутые правила.   


Рисунок

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

Клетка содержит в себе позицию, рост и цвет.

3. Теперь добавим матрицу для нашей симуляции.



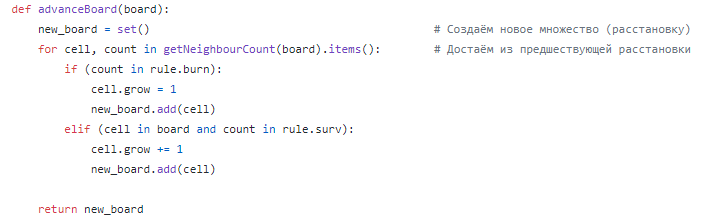
Рисунок

Матрица – это множество клеток

4. Имея правила, элементарный участок и матрицу, перейдём к описанию шагов.

- Сгенерировать матрицу (generateBoard())

- Обновлять матрицу (advanceBoard())



Рисунок

Этот шаг мы делаем сколько угодно раз. И тем самым происходит симуляция. Пока что мы игнорируем симуляцию роста и цвета, поэтому они будут генерироваться произвольно.

Весь код, включая модель почвы, будет приведён в приложении.

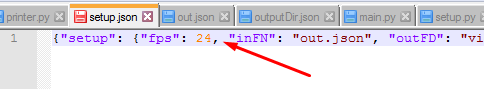
Итоги: в этой версии мы пытались повторить классические правила с возможностью их модификации. Эта версия рабочая. Далее будут добавлены: почва и правила связанные с ней. А именно трава не будет погибать, если под ней плодородная почва. Плодородность имеет диффузию, и растёт, когда травы не слишком много или мало и убывает, когда над ней трава, которая по классическим правилам должна была погибнуть.

# 5. Запуск всей работы.

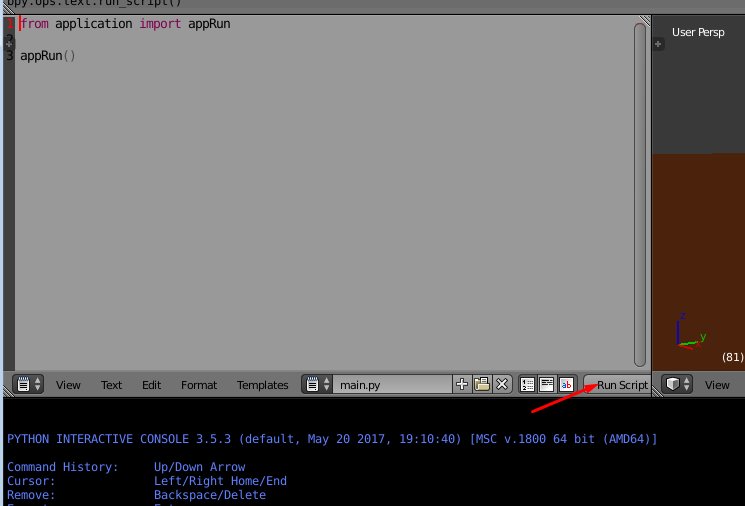
Добавление всех файлов в блендер и запуск скриптов для создания папки с видео.

Порядок работы:

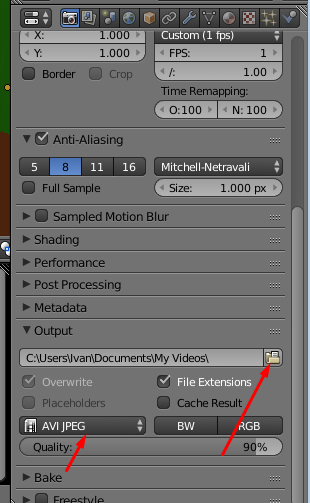
## Программирование

1. Открыть файл setup.json через блокнот и настроить количество кадров в секунду fps и сохранить файл.  
   

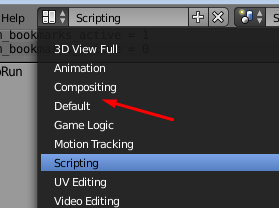
Рисунок

1. Открыть файл travaForWork.blend и нажать Run Script.  
   

Рисунок

1. Во вкладке Render, свитке Out настроить тип файла вывода и директорию вывода.  
   

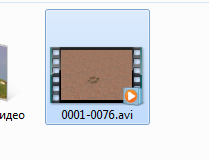
Рисунок

1. Перейти в вид Default.  
   

Рисунок

1. Во вкладке Render нажать animation.  
   

Рисунок

1. Готово. В заданной директории появится видео.  
   

Рисунок