

## РЕАЛИЗАЦИИ ПРОЦЕДУРНОЙ ГЕНЕРАЦИИ ЛАНДШАФТА И РЕСУРСОВ В UNITY

С точки зрения программиста, для создания любого предмета есть два пути: сделать его руками или написать код, для его генерации.

Если необходимо небольшое количество контента, то, конечно же, самым простым способом будет создать его руками (расставить по карте деревья или нарисовать текстуру). Однако если говорить про действительно большие объёмы материала, то процедурная генерация может стать более выгодным способом. Принцип довольно прост, когда вам необходимо что то, то используя ГСПЧ, порождающий элемент, дополнительные входные данные и какой-либо алгоритм создания необходимой вещи вы получаете уникальную единицу контента. Если после генерации происходят какие-либо изменения, то вы их сохраняете отдельно. Впоследствии, если необходимо повторить сгенерированный объект, вы, зная порождающий элемент, входные данные и внесённые изменения можете повторить его генерацию и получить данный объект. Одним из универсальных способов описания земной поверхности является применение карты высот. Она представляет собой двумерный массив с данными о высоте точек на местности [1].

Обычный генератор псевдослучайных чисел не очень подходит для генерации карты, т. к. белый шум не предполагает зависимости от близлежащих точек, а, следовательно, мы получим большие перепады, не сильно похожие на реальную поверхность. Для генерации связанных данных воспользуемся ГСПЧ под названием шум Перлина. Обычный шум Перлина имеет очень простую структуру. Для решения этой проблемы складывают несколько шумов разного масштаба и интенсивности, за счёт чего появляется больше деталей. Эти повторения называются октавами (рисунок 1).

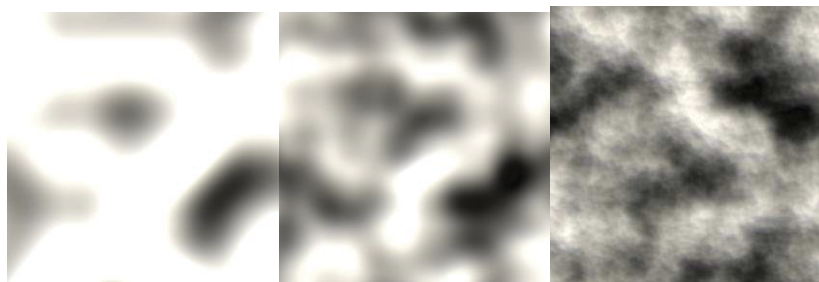


Рисунок 1 –Шум Перлина с 1-ой, 2-мя и 4-мя октавами

Полученный результат является основой для генерации поверхности. Более светлые участки отвечают за более высокие области на поверхности. Далее могут быть применены различные виды искажений и различные кривые.

Для создания 3d модели в Unity существует класс Mesh, служащий для создания трёхмерной сетки прямо в коде. Для генерации поверхности предварительно создаётся двумерная квадратная сетка. Координаты x и z не изменяются, а координата y устанавливается в соответствии с имеющейся картой высот.

```
mesh = new Mesh();
GetComponent<MeshFilter>().mesh = mesh;
vertices = new Vector3[(borderSize + 1) * (borderSize + 1)];
triangles = new int[borderSize * borderSize * 6];
uv = new Vector2[vertices.Length];
tangents = new Vector4[vertices.Length];

mesh.Clear();
mesh.vertices = vertices;
mesh.triangles = triangles;
mesh.tangents = tangents;
mesh.RecalculateNormals();
GetComponent<MeshCollider>().sharedMesh = mesh;
```

Рисунок 2 – Создание меша через скрипт

Созданный таким способом мир разбивается на участки и загружается, если рядом находится игрок. Это позволяет создать мир огромного размера, а также улучшает производительность.

Полученный ландшафт можно наполнить различными объектами, такими как деревья, камни и другими деталями. Для их размещения можно пройти по вершинам сетки, выполнить рейкаст, и, если луч столкнулся с поверхностью, с некоторой вероятностью разместить там объект. Для того чтобы объекты не разбрасывались равномерно, а группировались, имеет смысл также использовать шум Перлина и изменять вероятность в соответствии с ним.

## ЛИТЕРАТУРА

1. Процедурная генерация и Шум Перлина(PerlinNoise) в Unity [Электронный ресурс] / Kihontekina. – Режим доступа: [https://ru.kihontekina.dev/posts/perlin\\_noise\\_part\\_one/](https://ru.kihontekina.dev/posts/perlin_noise_part_one/). – Дата доступа: 03.04.2022.