

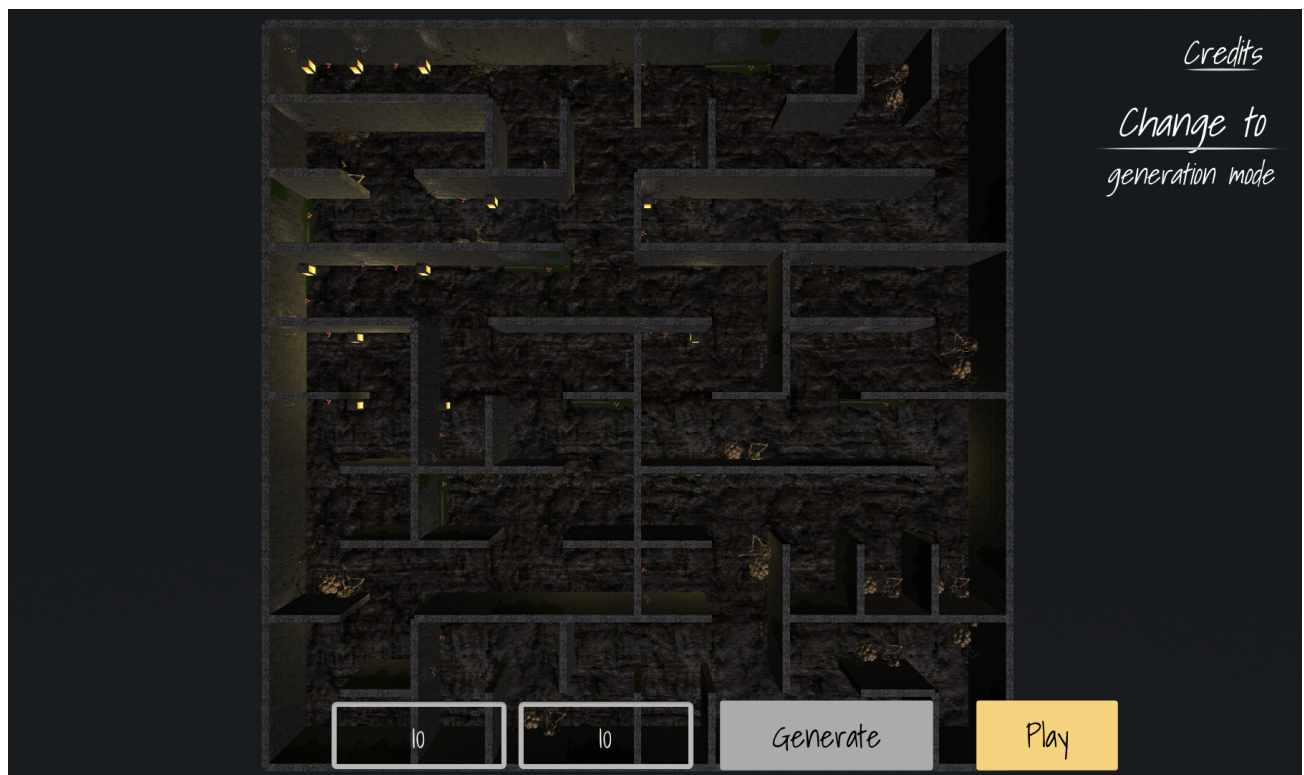


Test project

Report and project by
Kristiine Pärna

MAZE GENERATION WITH ALDOUS-BRODER ALGORITHM

Report



INTRODUCTION

In this report I am briefly going to explain my creation process, problems and results of the maze project. There is also Play mode, which allows you to just go through the maze. Play mode is only possible on the desktop and in limited size.

WORK PLAN

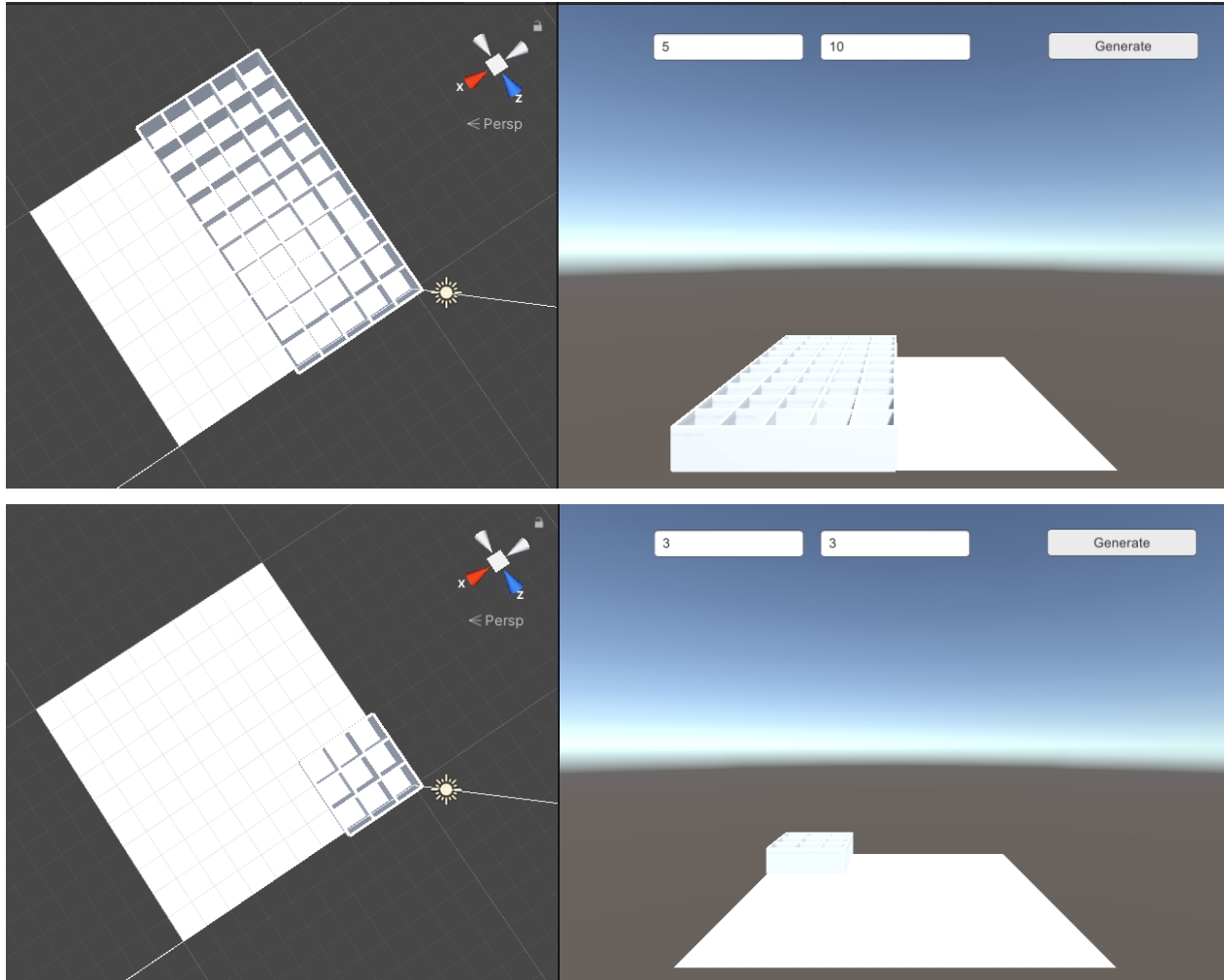
Here is a short plan that I created in the beginning after taking a look at different algorithms and finding the one that I'm gonna use.

- Make a decision between 2D or 3D
- Create grid with walls
- Make UI for user to insert square amount for height and for width with maze generation button
- Apply algorithm
- Look through code and things in Unity to make sure that everything is named correctly and code is written efficient way with enough comments
- Maze floor size is same as maze size
- Make sure that the maze is always fully seen on the screen, with every maze size
- Add design to maze
- Add UI design
- Look how it looks in ipad and iphone
- Extras

GENERATING MAZE GRID

In this paragraph I am going to try to explain what my idea was and what I exactly did. First, I wanted to create a simple grid. For that I wrote two functions, one which creates the walls in one direction and one which creates them in another direction.

After that I applied the algorithm, which allowed me to start removing the walls inside the grid in order to create a maze.



ALGORITHM

I decided to use the Aldous-Broder algorithm. I chose this one because I am not that familiar with the algorithms and so decided to start with something easier. Implementing the algorithm was not too complicated. As with the research on the algorithm, it was said that its generating is slow and that is also a thing that is a little disturbing with my maze generation.

Now my next step was to test it - take time and see how much it takes to generate bigger sizes. For 100x100 it takes around 45 sec (plain designed wall) to 1min - 1min 30sec (different designed walls). But after that the time really quickly starts to rise exponentially and for a 200x200 maze it already takes around 10 minutes (plain designed wall) to 17-18 minutes (different designed walls). I spent 2-3h trying different techniques in order to make it more

efficient but ended up making the maze not so perfect or these were not really helping to improve the generation time. STILL, the code consists of a function which starts to look for the unvisited square from the beginning when the algorithm has been going too much around in the grid without finding one.

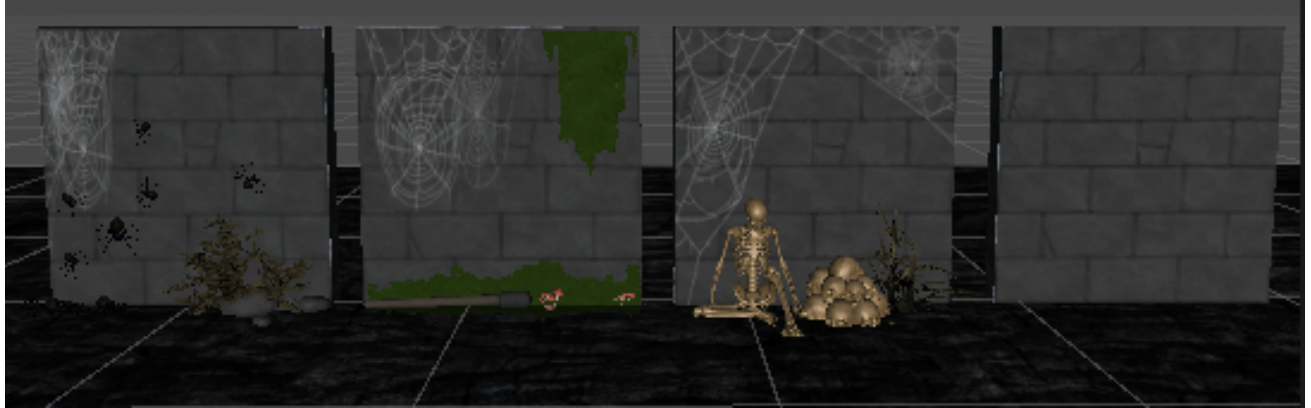
In conclusion to the algorithm, I would say that even though it can be a little bit small, it is still able to create really different and interesting mazes every time it is run. Also, I think it works perfectly with making the mazes max to size 100x100.

MAZE DESIGN

My maze design is inspired by a horror theme. I created 8 different designs for the walls. In the grid generator all of these walls are used on a random basis.

Making the wall design was one of the most interesting parts for me. For the wall base I had a random cube with a material. On both of the sides I added a floor actually, because only that way I was satisfied with the wall overall look and how it looked in the maze. Next, I tried to make unique wall designs. For that I used the clean wall base I just made and then added different prefabs to give it an unique look. Later on added some spiders and bones together with the light for the lanterns. Sadly the bigger the maze, the more effort it takes for the computer to run normally.





TWO MODES - GENERATION AND PLAY

In generation mode the user is able to generate a simple maze without big design in order to make the generating process faster. But there is also added gameplay mode, where the player is able to move around in the 3D environment. Generation and “playing” works fine with smaller mazes, but if the maze gets bigger, it gets too big to run all the assets. Therefore the gameplay mode is lowered down to 10x10 squares max currently.

FUTURE POSSIBILITIES AND IDEAS

How can my project be reused:

- It is really easy to change the whole maze design by adding different designed walls. It is just needed to change the wall prefabs in unity. So like that it is easy to create different themed mazes. This can even be a part of generation, where a person can choose between certain pre-made styles.
- Grid base - looking through different algorithms I found that there are few more that use as a grid for the base of generation. Hopefully this grid is easily usable for those kinds of algorithms.
- Making it to a game. My current maze seems good enough to make it easily into a 3D maze game. I even had a few ideas, like adding items and monsters in the maze for the player.
- Figure out how to make bigger sizes generate faster so that the algorithm works more efficiently.

REFERENCES

In this list you can find everything (hopefully) that I used for my project.

Algorithm:

https://en.wikipedia.org/wiki/Maze_generation_algorithm

<https://weblog.jamisbuck.org/2011/1/17/maze-generation-aldous-broder-algorithm>

Font:

<https://fonts.google.com/specimen/Shadows+Into+Light>

Programming help:

<https://answers.unity.com/questions/1022669/random-range-to-pick-from-two-specific-numbers-onl.html>

<https://forum.unity.com/threads/get-aspect-ratio.211255/>

Assets:

<https://assetstore.unity.com/packages/3d/characters/animals/insects/animated-spider-22986#description>

<https://assetstore.unity.com/packages/3d/characters/animals/insects/animated-spiders-pack-9864#description>

<https://assetstore.unity.com/packages/2d/textures-materials/sky/skybox-series-free-103633#description>

<https://assetstore.unity.com/packages/3d/environments/dungeons/lowpoly-dungeon-modules-108997#description>

<https://assetstore.unity.com/packages/3d/props/free-rigged-skeleton-and-bone-collection-166829>

<https://assetstore.unity.com/packages/3d/vegetation/free-desert-plants-32410>