RANDOM MAP GENERATION

Personal Research - Maria Garrigolas Ledo

TABLE OF CONTENTS

01 INTRODUCTION

Introduction to the topic.

02 MARKET STUDY

What games use this type of algorithm and why.

03 SELECTED APPROACH

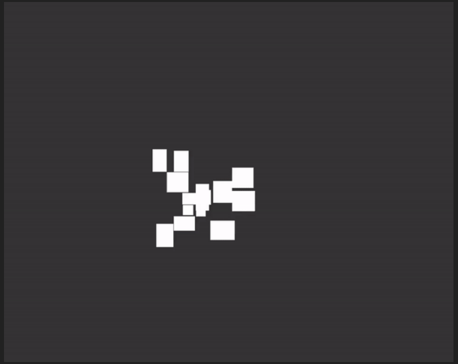
Perlin Noise.

04 CODE

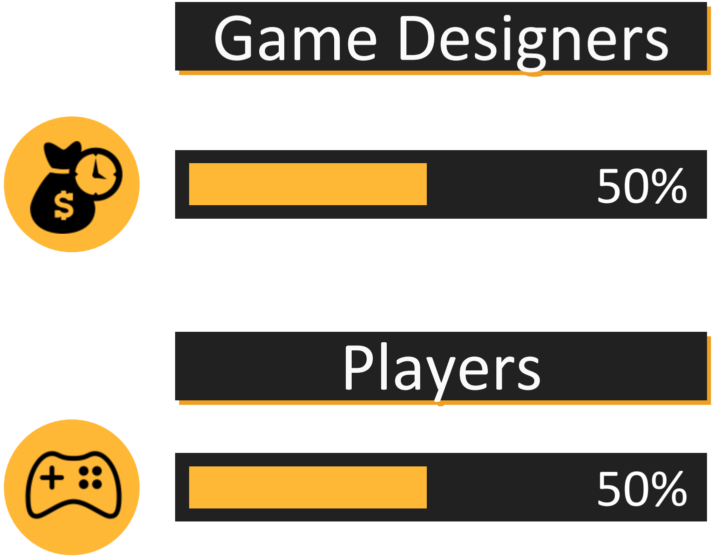
Explanation of the code.
Exercises with solutions.

WHAT ARE RANDOM GENERATED MAPS?

Process to generate content algorithmically



Why do we want random generated maps?



Game Designers

50%

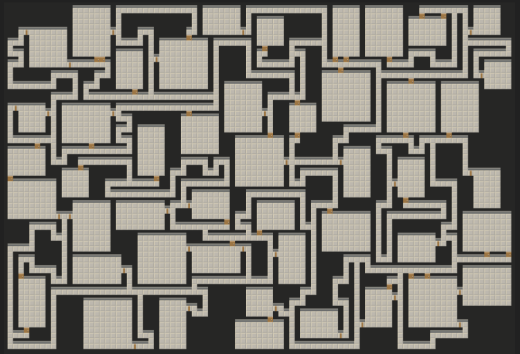
Players

50%

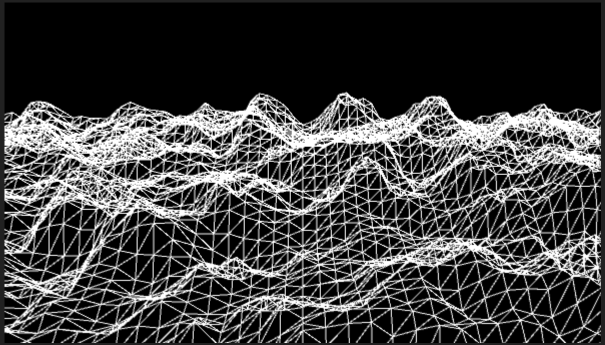
The pros and cons of random generated maps?

PROS	CONS
LOW BUDGET	REPETITIVE WORLDS
INCREASE GAMEPLAY VARIETY	UNPLAYABLE WORLD
REPLAYABILITY	
LARGER MAPS	
SAVES TIME	

DIFFERENT TYPE OF MAPS

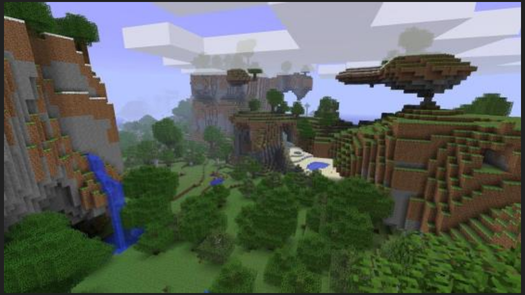


Dungeon maps

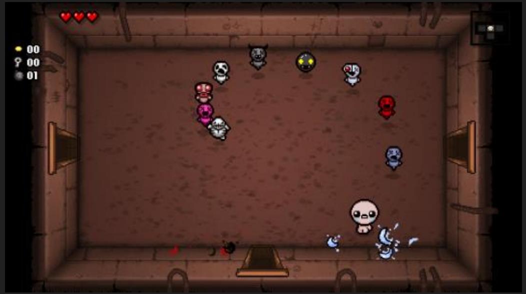


Outdoor maps

GAMES THAT USE RANDOM GENERATED MAPS



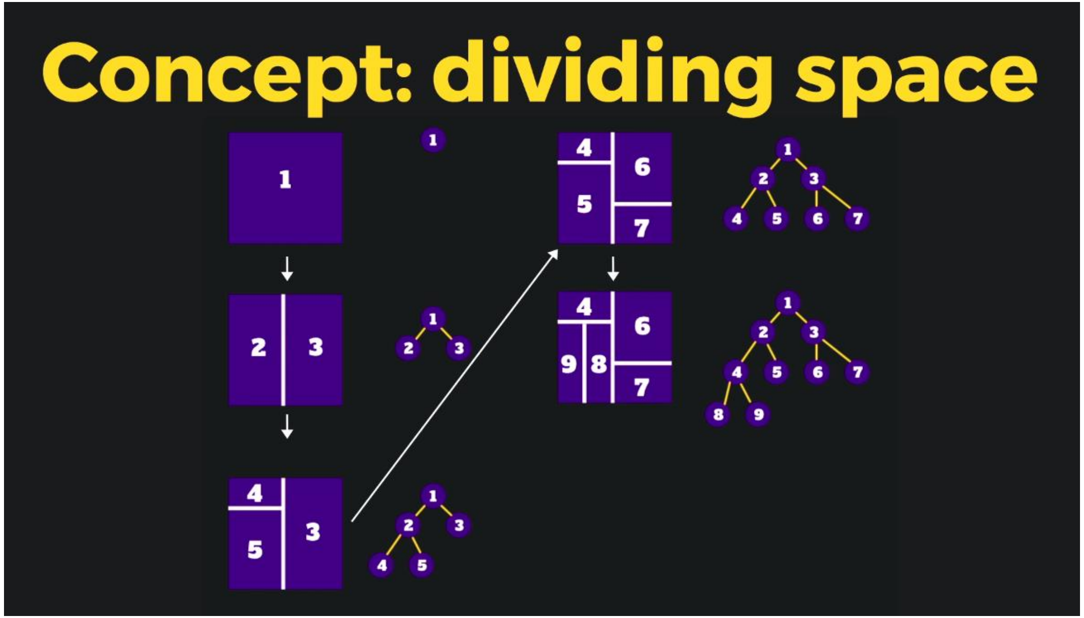
MINECRAFT



THE BINDING OF ISAAC

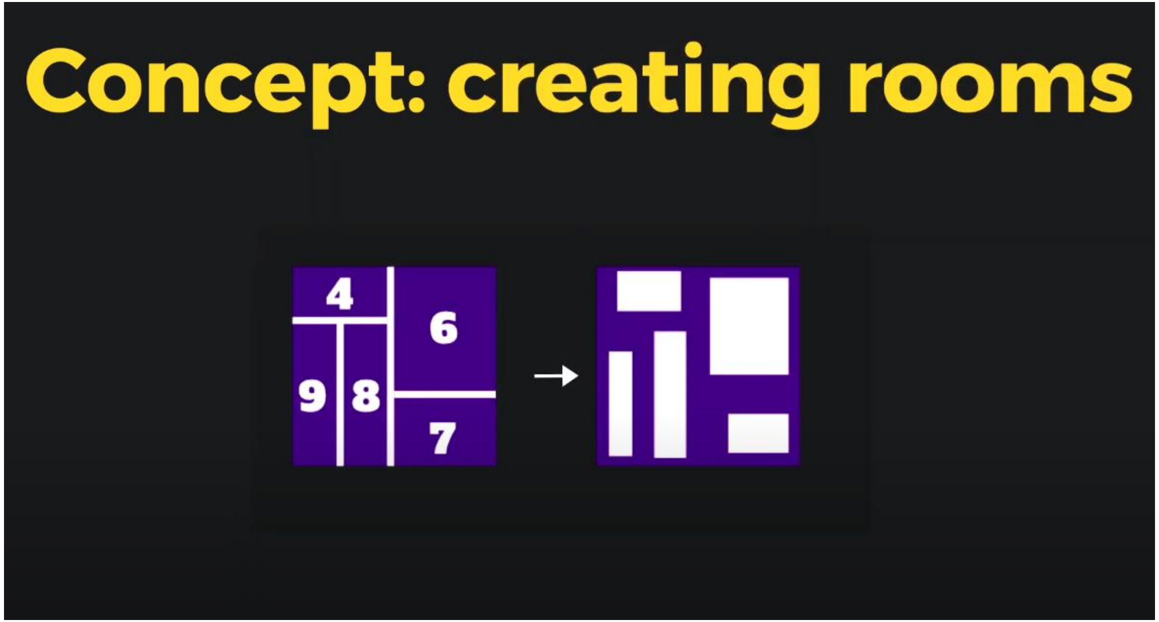
INDOOR MAP ALGORITHM

Binary space partitioning for Dungeons



INDOOR MAP ALGORITHM

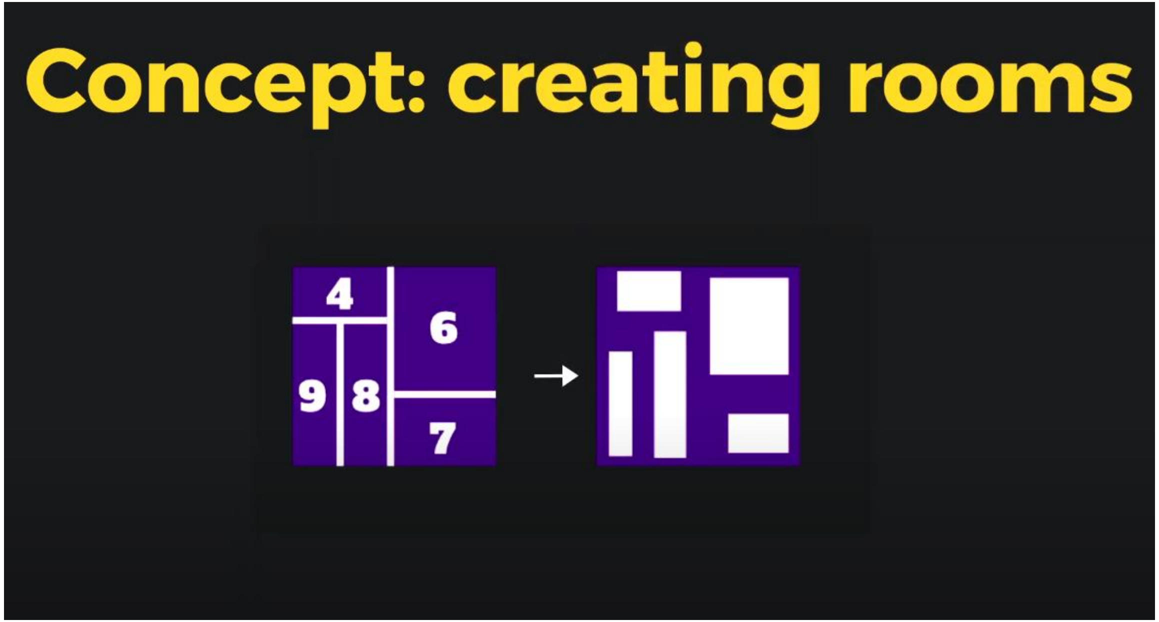
Binary space partitioning for Dungeons



Concept: creating rooms

INDOOR MAP ALGORITHM

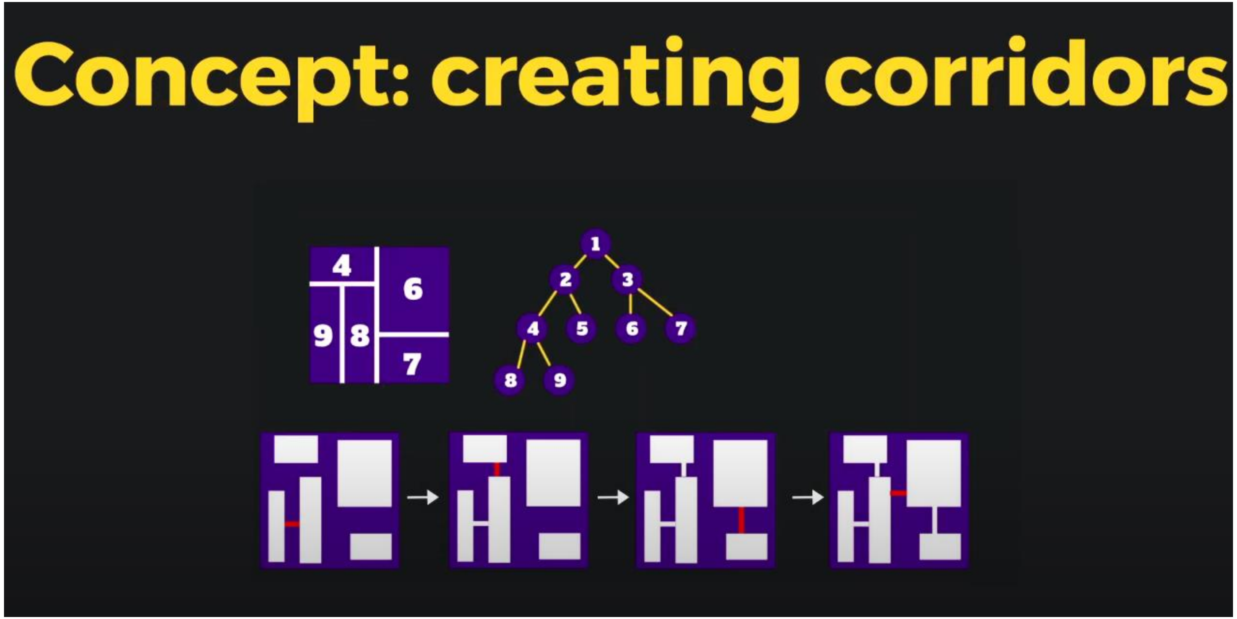
Binary space partitioning for Dungeons



Concept: creating rooms

INDOOR MAP ALGORITHM

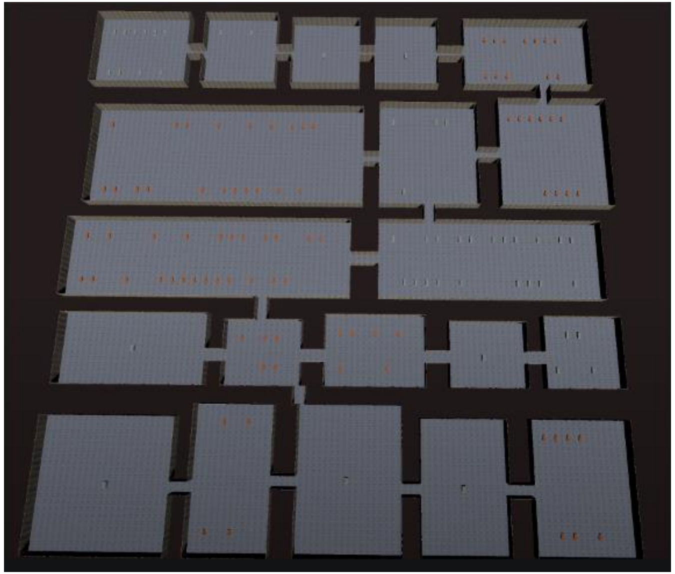
Binary space partitioning for Dungeons



Concept: creating corridors

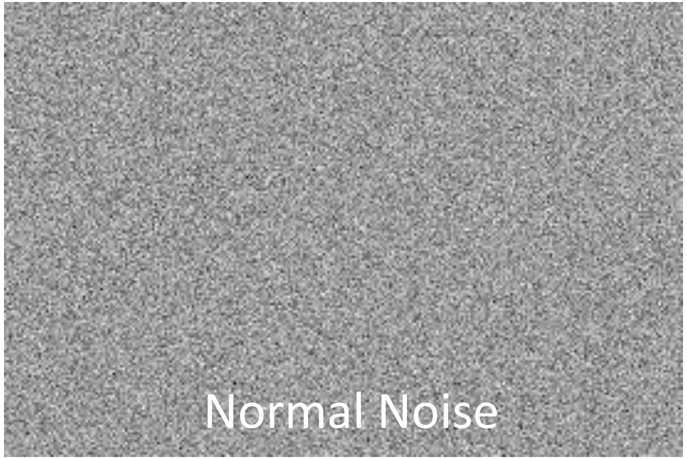
INDOOR MAP ALGORITHM

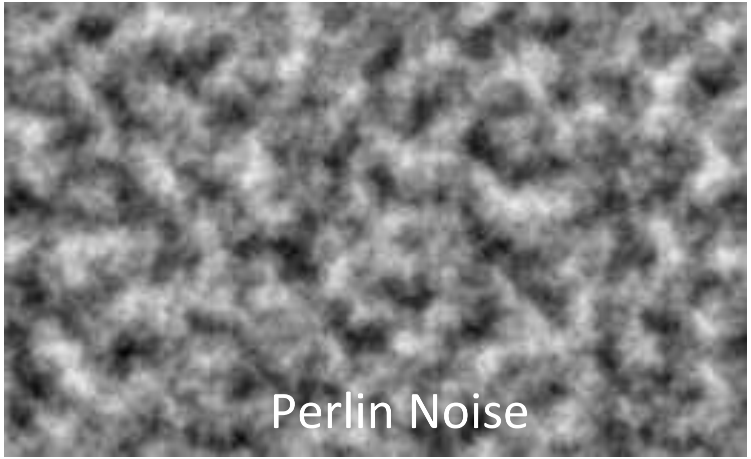
Binary space partitioning for Dungeons



OUTDOOR MAP ALGORITHM

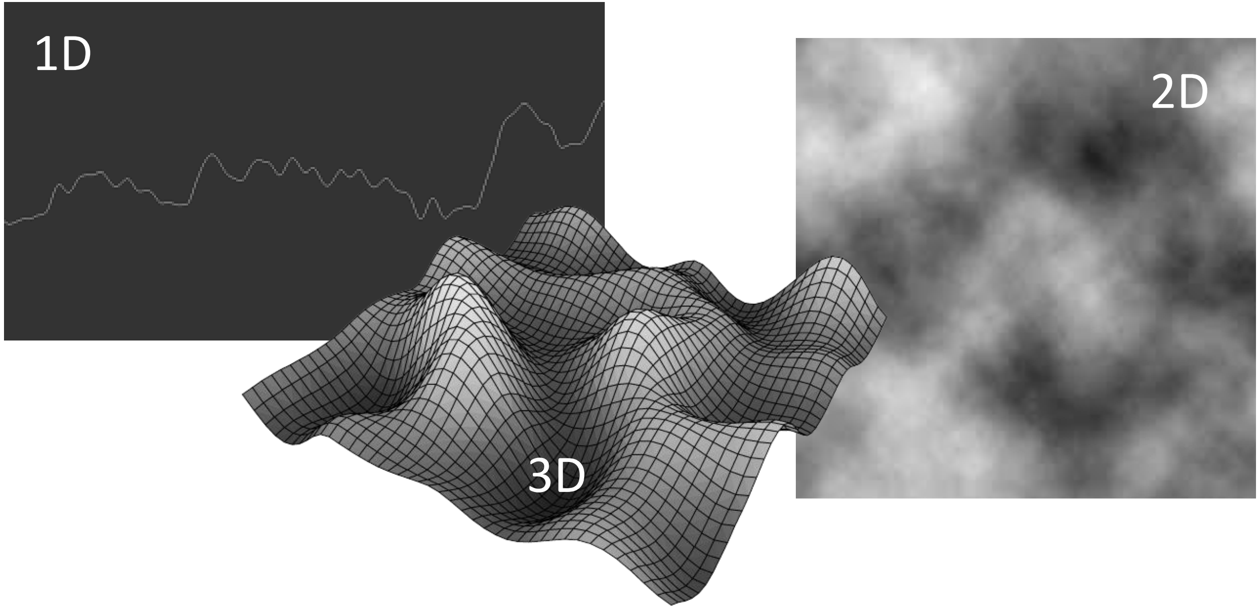
Perlin Noise





OUTDOOR MAP ALGORITHM

Perlin Noise



OUTDOOR MAP ALGORITHM

Perlin Noise

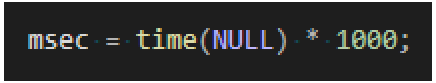
FASTNOISELITE by Jordan Peck

https://github.com/Auburn/FastNoiseLite

OUTDOOR MAP ALGORITHM

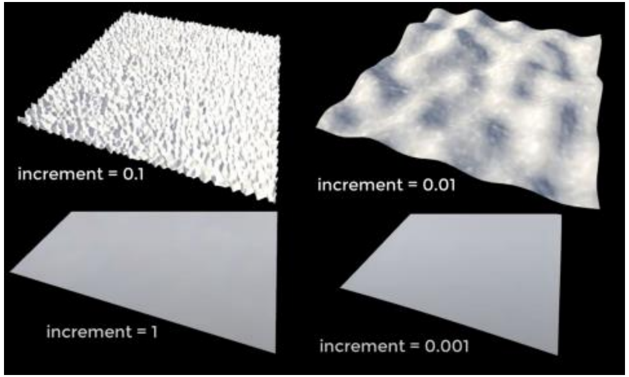
Perlin Noise

Seed

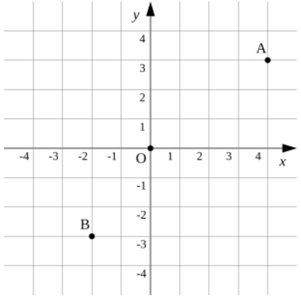


```
msec = time(NULL) * 1000;
```

Frequency



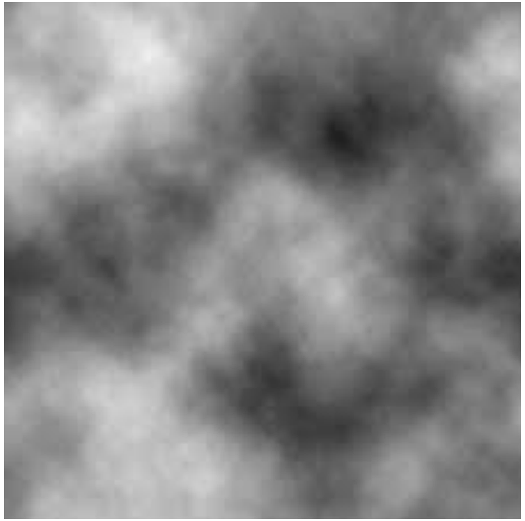
Coords X, Y

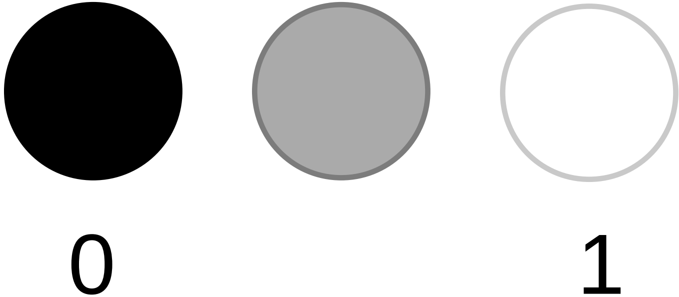


Formula: (Noise + 1) *0.5

OUTDOOR MAP ALGORITHM

Perlin Noise





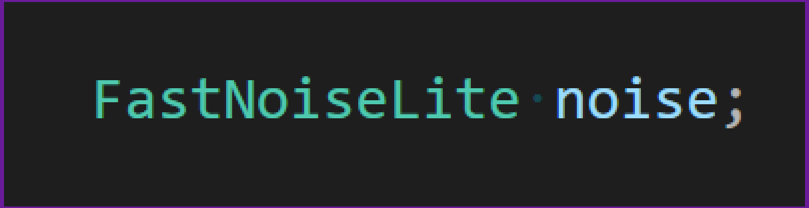
EXERCISES

TODO 1

Create a FastNoiseLite object

Solution

TODO 1



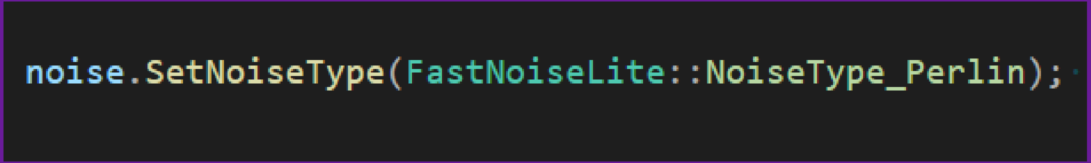
```
FastNoiseLite noise;
```

TODO 1.2

Set its noise type to Perlin noise

TODO 1.2

Solution



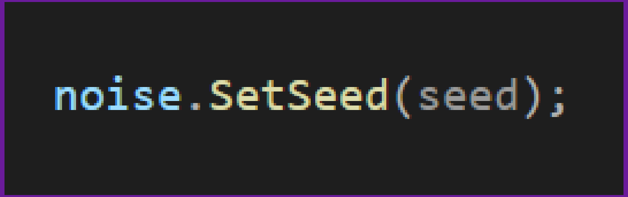
```
noise.SetNoiseType(FastNoiseLite::NoiseType_Perlin);
```

TODO 1.3

Set the seed with SetSeed function

Solution

TODO 1.3



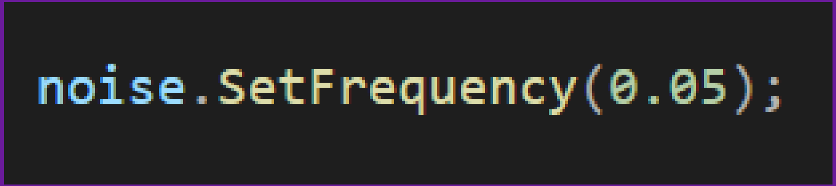
```
noise.SetSeed(seed);
```

TODO 1.4

Set the frequency to 0.05

TODO 1.4

Solution



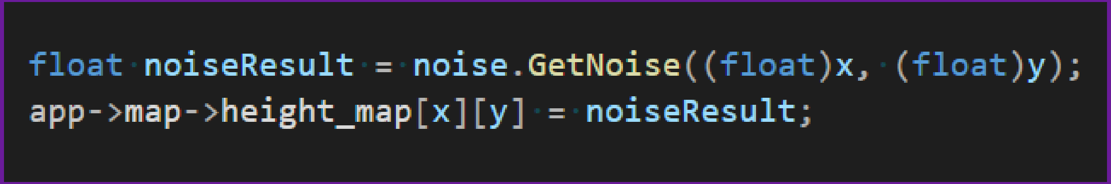
```
noise.SetFrequency(0.05);
```

TODO 2

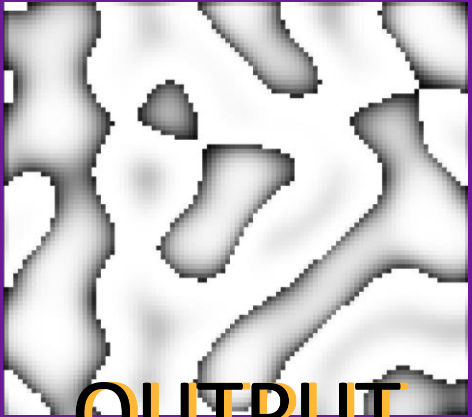
Store the values generated by Perlin Noise in app->map->height_map

TODO 2

Solution



```
float noiseResult = noise.GetNoise((float)x, (float)y);
app->map->height_map[x][y] = noiseResult;
```



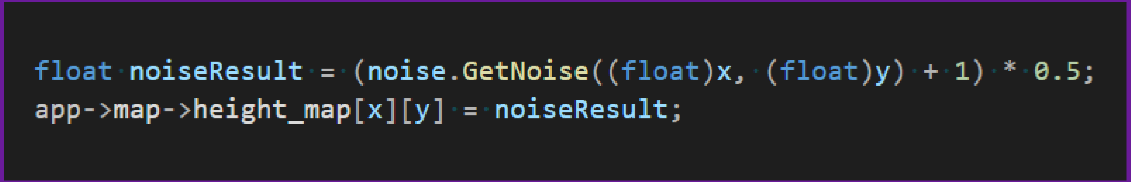
OUTPUT

TODO 2.1

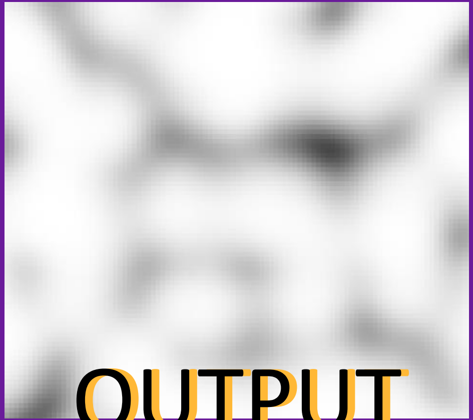
Noise must be always between 1 and 0. Use the following formula: (Noise + 1) *0.5.

TODO 2.1

Solution



```
float noiseResult = (noise.GetNoise((float)x, (float)y) + 1) * 0.5;
app->map->height_map[x][y] = noiseResult;
```



OUTPUT

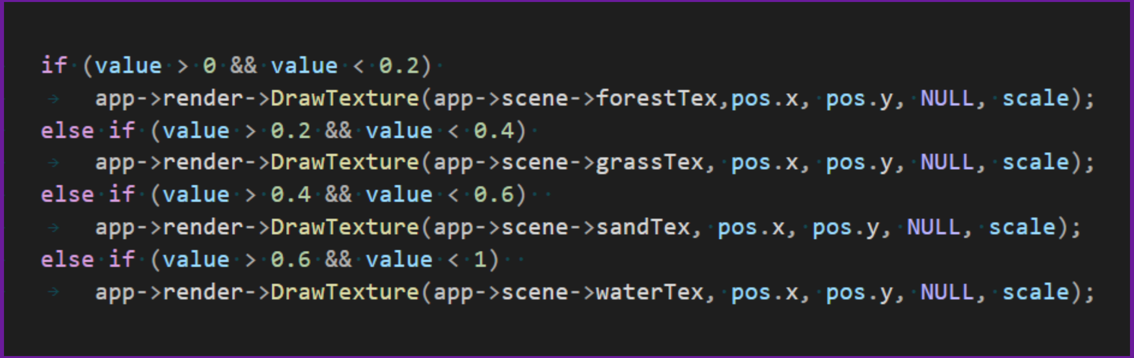
TODO 3

Draw the different textures into the map.

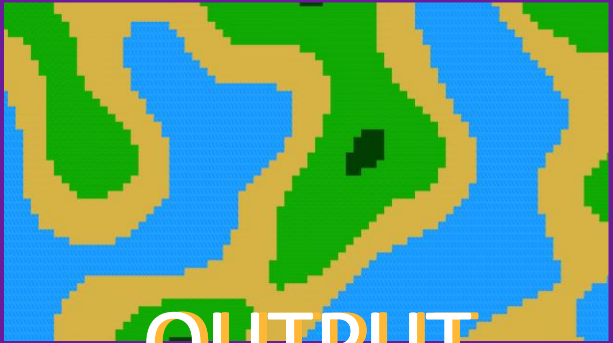
forestTex	0 - 0.2
grassTex	0.2 - 0.4
sandTex	0.4 - 0.6
waterTex	0.6 - 1

TODO 3

Solution



```
if (value > 0 && value < 0.2)
	app->render->DrawTexture(app->scene->forestTex,pos.x, pos.y, NULL, scale);
else if (value > 0.2 && value < 0.4)
	app->render->DrawTexture(app->scene->grassTex, pos.x, pos.y, NULL, scale);
else if (value > 0.4 && value < 0.6)
	app->render->DrawTexture(app->scene->sandTex, pos.x, pos.y, NULL, scale);
else if (value > 0.6 && value < 1)
	app->render->DrawTexture(app->scene->waterTex, pos.x, pos.y, NULL, scale);
```



OUTPUT

BIBLIOGRAPHY

https://github.com/Auburn/FastNoiseLite

https://www.youtube.com/watch?v=VnqN0v95jtU

https://www.youtube.com/watch?v=jv6YT9pPIHw

https://subscription.packtpub.com/book/game_development/9781785886713/1/ch01lvl1sec16/how-we-ll-implement-procedural-generation

https://www.redblobgames.com/maps/terrain-from-noise/

https://www.redblobgames.com/articles/noise/introduction.html

http://eastfarthing.com/blog/2015-04-21-noise/

https://upcommons.upc.edu/bitstream/handle/2099.1/26632/109439.pdf?sequence=1&isAllowed=y

THANKS!

DO YOU HAVE ANY QUESTIONS?

Maria.garrigolas@estudiantat.upc.edu
https://meeeri08.github.io/PersonalResearch-Random-Map-Generator/

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik**

Please keep this slide for attribution