Appendix A

Extra Information

A.1 Tables & Test Cases

This section contains tables and test cases mentioned in the Evaluation section of this report. They are included here as they remain peripheral to the main body of this report, and would break up the theme and flow of the text if it appeared in the body, as would the source code listings featured in chapter C and the detailed user guide featured in chapter B.

	Noise type				
Cellular Distance Function	Simplex Smooth	Simplex	Perlin	Value	Value Cubic
Euclidean	81ms	83ms	80ms	82ms	92ms
	84ms	88ms	74ms	84ms	105ms
	83ms	83ms	74ms	83ms	74ms
	AVG: 83ms	AVG: 85ms	AVG: 76ms	AVG: 83ms	AVG: 90ms
Euclidean Squared	81ms	74ms	81ms	90ms	76ms
	77ms	91ms	73ms	119ms	109ms
	84ms	79ms	80ms	93ms	82ms
	AVG: 81ms	AVG: 81ms	AVG: 78ms	AVG: 101ms	AVG: 89ms
Manhattan	83ms	93ms	82ms	80ms	81ms
	107ms	101ms	72ms	81ms	72ms
	82ms	87ms	80ms	94ms	101ms
	AVG: 91ms	AVG: 94ms	AVG: 78ms	AVG: 85ms	AVG: 85ms
Hybrid (Euclidean & Manhattan)	77ms	87ms	85ms	85ms	76ms
	96ms	122ms	74ms	85ms	77ms
	82ms	87ms	83ms	87ms	77ms
	AVG: 85ms	AVG: 99ms	AVG: 81ms	AVG: 86ms	AVG: 77ms

Figure A.1: A table denoting some performance tests done with comparing the noise algorithms, the cellular distance functions and the combination pairs between them. This was done on the Noise implementation of the game; the "tests" were simply checking how long it took to create levels on the author of this report's computer, and all the other script variables were assigned to their default values as described in chapter 4. Each noise type and cellular distance function pair was run 3 times, with the mean time (including potential outliers) calculated afterwards *to the nearest integer*. Be advised that the author of this report did these tests on his computer, so on different computers, results can, and likely *will*, vary.

	Noise type				
Fractal Type	Simplex Smooth	Simplex	Perlin	Value	Value Cubic
None	92ms 86ms 123ms AVG: 100ms	89ms 152ms 99ms AVG: 113ms	84ms 98ms 95ms AVG: 92ms	101ms 86ms 97ms AVG: 95ms	77ms 88ms 86ms AVG: 84ms
FBM (Fractional Brownian Motion)	77ms 93ms 87ms AVG: 86ms	81ms 87ms 93ms AVG: 87ms	73ms 79ms 73ms AVG: 75ms	78ms 137ms 82ms AVG: 99ms	68ms 64ms 87ms AVG: 73ms
Ridged	23ms 25ms 23ms AVG: 24ms	74ms 69ms 70ms AVG: 71ms	15ms 16ms 16ms AVG: 16ms	27ms 28ms 26ms AVG: 27ms	14ms 9ms 11ms AVG: 11ms
Ping Pong	59ms 54ms 58ms AVG: 57ms	67ms 77ms 64ms AVG: 69ms	108ms 105ms 111ms AVG: 108ms	128ms 71ms 72ms AVG: 90ms	163ms 172ms 164ms AVG: 166ms

Figure A.2: A table denoting some performance tests done with comparing the noise algorithms, the fractal types and the combination pairs between them. This was done on the Noise implementation of the game; the "tests" were simply checking how long it took to create levels on the author of this report's computer, and all the other script variables were assigned to their default values as described in chapter 4. Each noise type and fractal type pair was run 3 times, with the mean time (including potential outliers) calculated afterwards *to the nearest integer*. Be advised that the author of this report did these tests on his computer, so on different computers, results can, and likely *will*, vary.

use_custom_axiom = false axiom = "OWB"	use_custom_axiom = true upper_limit = 3	use_custom_axiom = true upper_limit = 10	use_custom_axiom = true upper_limit = 25
21ms	25ms (length = 2)	20ms (length = 4)	9ms (length = 25)
17ms	13ms (length = 2)	11ms (length = 9)	14ms (length = 2)
21ms	21ms (length = 1)	21ms (length = 8)	21ms (length = 21)
20ms	16ms (length = 1)	18ms (length = 5)	20ms (length = 24)
20ms	11ms (length = 2)	11ms (length = 4)	15ms (length = 14)
AVG: 20ms	AVG: 17ms	AVG: 16ms	AVG: 16ms

Figure A.3: A table denoting some performance tests done with comparing the lengths of axioms used in L-Systems. Obviously, this was done on the L-System implementation of the game; the "tests" were simply checking how long it took to create levels on the author of this report's computer, as well as how long the randomly generated axioms were (where appropriate), and all the other script variables were assigned to their default values. Each of the shown settings were run 5 times, with the mean time (including potential outliers) calculated afterwards *to the nearest integer*. Be advised that the author of this report did these tests on his computer, so on different computers, results can, and likely *will*, vary.

rejection_samples	3	8	13	18
time	170ms	337ms	444ms	503ms
	103ms	224ms	392ms	505ms
	111ms	242ms	388ms	670ms
	AVG: 128ms	AVG: 268ms	AVG: 408ms	AVG: 559ms

Figure A.4: A table denoting some performance tests done with comparing the number of rejection samples used for Poisson Disk Sampling. Obviously, this was done on the Poisson Disk Sampling/Distribution implementation of the game; the "tests" were simply checking how long it took to create levels on the author of this report's computer, and all the other script variables were assigned to their default values. Each of the shown settings were run 3 times, with the mean time (including potential outliers) calculated afterwards *to the nearest integer*. The bottom cell of the rightmost column, with tests done with 18 rejection samples, is highlighted red because, while testing with 18 rejection samples, at one time the program hung without returning any cell points within 10 seconds. The test had to be retaken another time. Be advised that the author of this report did these tests on his computer, so on different computers, results can, and likely *will*, vary.

	Random Starting Points			
Distance type	15	20	30	40
Euclidean distance	393ms 385ms 362ms AVG: 380ms	496ms 504ms 497ms AVG: 499ms	775ms 744ms 723ms AVG: 747ms	968ms 970ms 967ms AVG: 968ms
Manhattan Distance	364ms 346ms 346ms AVG: 352ms	441ms 470ms 437ms AVG:449ms	645ms 630ms 650ms AVG: 642ms	843ms 835ms 908ms AVG: 862ms

Figure A.5: A table denoting some performance tests done with comparing the distance calculation algorithms, the number of random starting points and the combination pairs between them. This was done on the Voronoi Cells implementation of the game, and thus the number of random starting points corresponds directly to the number of unique Voronoi cells generated for each level. The "tests" were simply checking how long it took to create levels on the author of this report's computer. Each of the shown settings were run 3 times, with the mean time (including potential outliers) calculated afterwards *to the nearest integer*. Be advised that the author of this report did these tests on his computer, so on different computers, results can, and likely *will*, vary.

L-System	Perlin/Simplex Noise	Poisson Disk Sampling/Distribution	Voronoi Cells
24ms	78ms	176ms	502ms
10ms	78ms	190ms	425ms
14ms	82ms	210ms	455ms
14ms	91ms	216ms	449ms
17ms	75ms	193ms	442ms
AVG: 16ms	AVG: 81ms	AVG: 197ms	AVG: 455ms

Figure A.6: A table denoting some performance tests done with comparing all of the algorithms implemented with the chosen scenario. This was done on every single implementation implementation of the game; the "tests" were simply checking how long it took to create levels on the author of this report's computer, and every implementation was tested with its default variable values. Each of the implementations were run 5 times, with the mean time (including potential outliers) calculated afterwards *to the nearest integer*. Be advised that the author of this report did these tests on his computer, so on different computers, results can, and likely *will*, vary.

Appendix B

User Guide

B.1 Opening Godot

You will first need to download Godot 4 to run all the provided artefacts.

To run the projects in the .zip file, extract the projects into one folder. Then open Godot 4 (all projects in the source code listings folder are Godot 4 projects, **not** Godot 3 projects). When you start Godot for the first time, the project manager should be completely empty, without any projects, as described in Figure B.1. Projects have to be imported either one-by-one (by clicking "Import" and going to the relevant project and opening it) or by clicking "Scan", then going to a folder of Godot projects and selecting it. The projects can then be opened in the project manager and edited as needed in Godot. Click "Scan", then go to the folder where you extracted the projects, then click the "Select Current Folder" button, as shown in Figure B.2, and all the projects should show up in the editor, as shown in Figure B.3. You can then double click on any one project (or click on it once and click the "Edit" button) to open it in the Godot editor, an example of which is shown in Figure B.4. Alternatively, to run the project itself without opening the editor, using the currently saved values for exported script variables where appropriate, click on the project *once* and click the "Run" button.

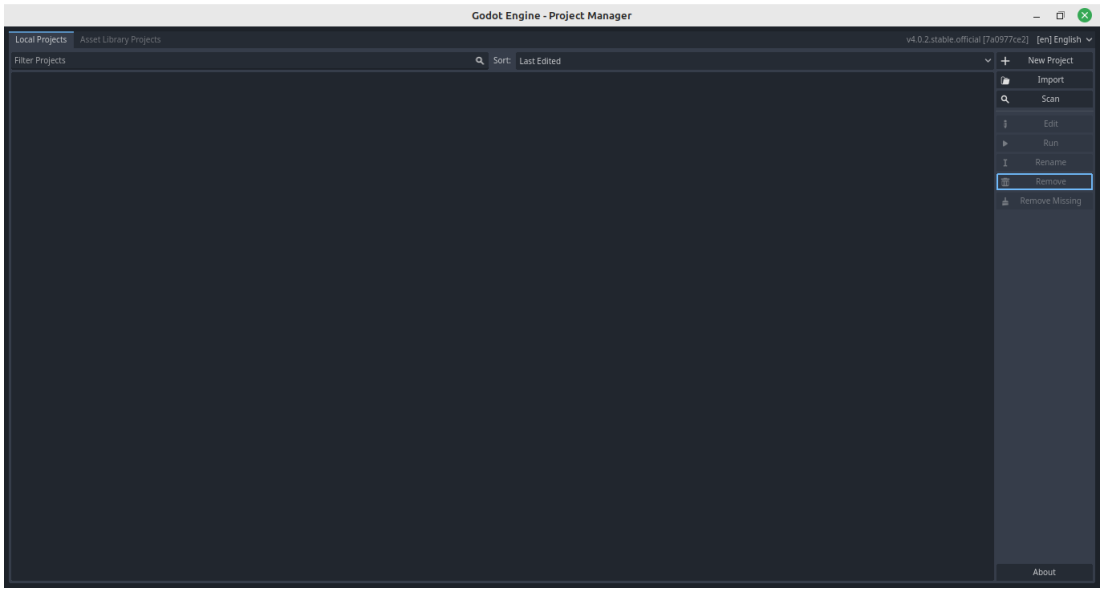

Figure B.1: The Godot editor, when it is opened for the first time, does not show any projects in the editor (the Steam version bundles several example projects). Projects need to be imported either one-by-one or by scanning a folder of Godot projects.

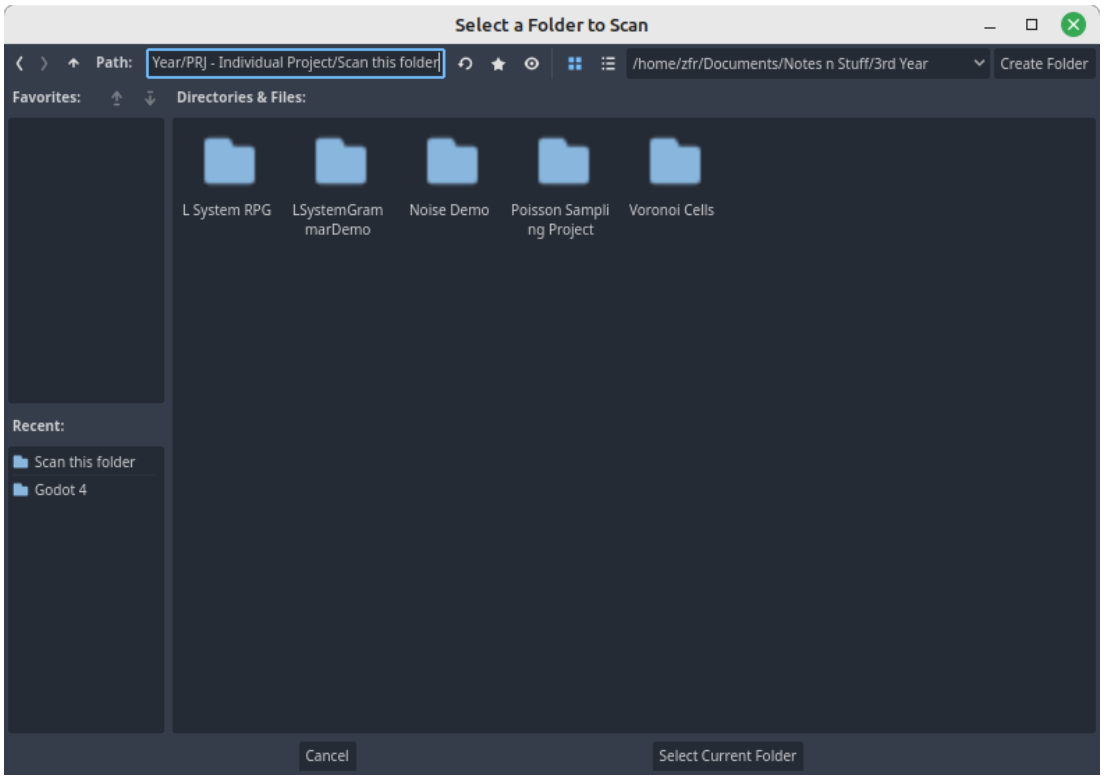

Figure B.2: You can click the ”Scan” button in the project manager (in Figure B.1), then go to the relevant folder where your project are in Godot’s built-in file explorer. Here, the artefacts behind this project have been exported into a separate folder called “Scan this folder” as an example.

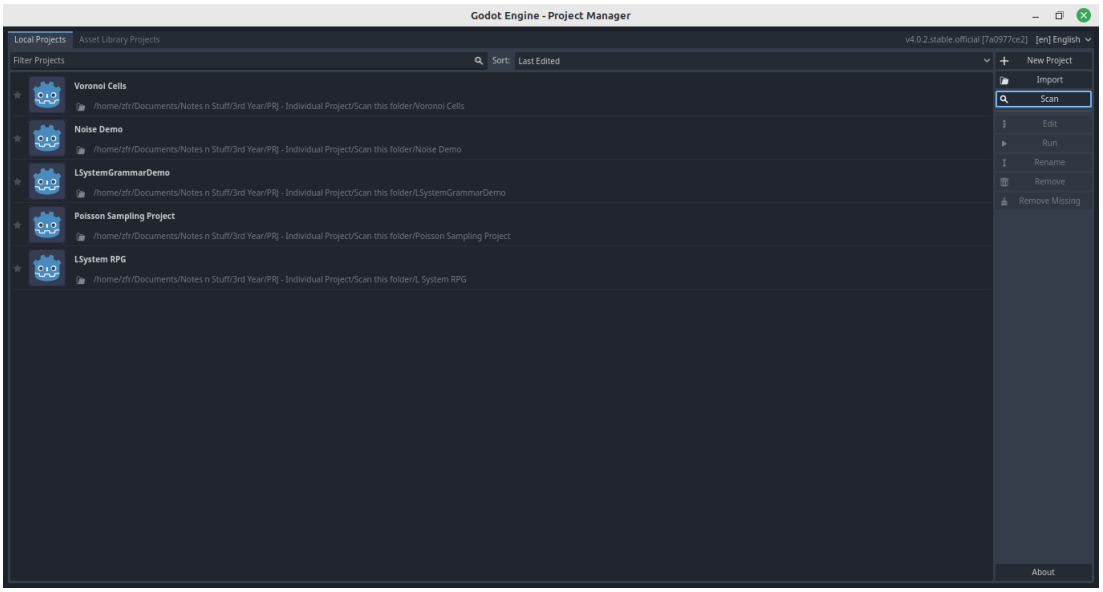

Figure B.3: Once some Godot projects have been imported into the project manager, you should be able to easily view the list and double-click on any one of the projects to edit them, which will open the editor after closing the project manager. You could also click the "Edit" button, or click "Run" to run the game without having to open the editor itself.

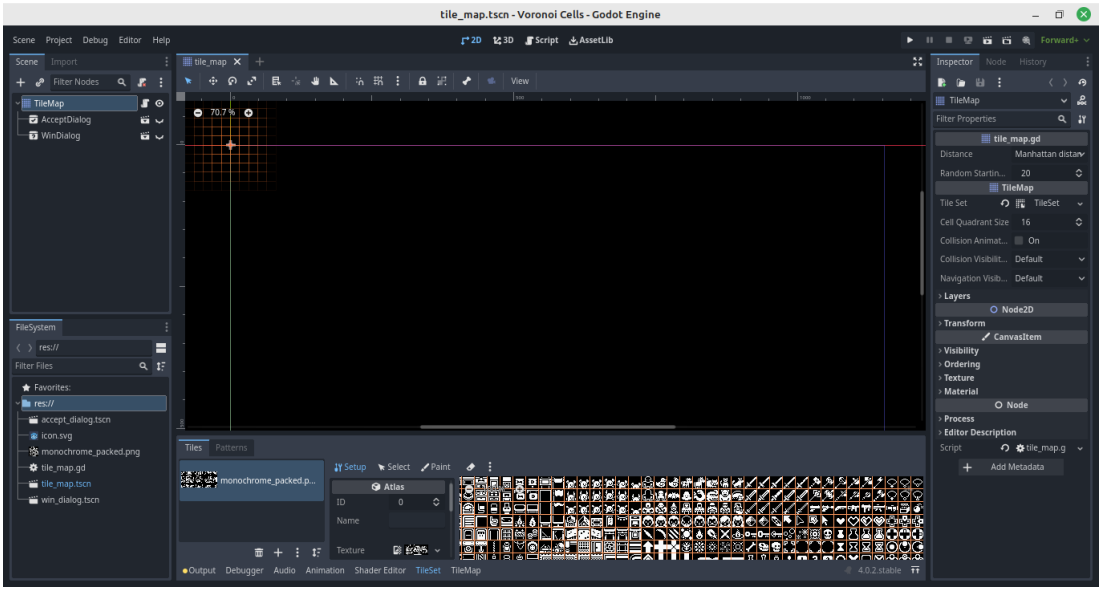

Figure B.4: The Godot editor open with the Voronoi cells project as an example. A visual description of the editor's contents is in chapter B.2.

B.2 The Godot Editor

As you open up the Godot editor, you will see the main scene view in the center, as shown in Figure B.4, using the Voronoi cells implementation as an example. The left-hand side shows

the scene tree at the top, and the file system (from the root folder of the project) at the bottom. Meanwhile, the right hand side shows the currently selected node's export variables, *including the custom export variables* defined in the node's script file, and two other tabs, "Node" (which shows a list of signals for the scene that could be called in a script) and "History" (which shows the sequence of recent actions performed on the scene during the current session). Above this is also a set of buttons which can be used for playing the project and/or the current scene. We go over how to run the current project in chapter B.4.

B.3 Custom Export Variables

When you click on some of the scenes in the projects, there may be some "exported" variables from scripts that are visible to you in the editor (examples include the "Distance" and "Random Starting Points" variables in the Voronoi Cells project). You can hover over the variable names in the editor and it will show a brief description of what the variable correlates to in the script. We will now go over the different export variables across all of the provided artefacts in this section.

B.3.1 Lindenmayer System

All of the custom export variables defined in scene scripts for your use are in the child node "LSystem" (it is saved into its own scene file, but it is cheifly a child node of the root node "TileMap"). Open the "TileMap" scene (if it is not already opened for you when you launch the Godot editor) and click on the "LSystem" node in the scene tree to edit it.

1. axiom
    - **Type:** String
    - **Documentation:** The starting string from which the grammar starts applying its rules. Here it may be self defined, or randomly defined when "use_random_axiom" is true.
    - **Default value:** "OWB"
2. use_random_axiom
    - **Type:** Boolean (bool)
    - **Documentation:** Uses a random axiom with the currently set grammar, computed upon runtime, with a length up to (but not strictly) the value of upper_limit. For

example, if upper_limit is set to 15, the generated axiom can be 15 characters, or it can be just 5 characters.

- **Default value:** true

3. upper_limit

   - **Type:** Integer (int)
   - **Documentation:** Defines how many characters a random axiom can have MAXIMUM. Only used when "use_random_axiom" is true.
   - **Default value:** 10

4. use_custom_ruleset

   - **Type:** Boolean (bool)
   - **Documentation:** Allows the use of a customly defined ruleset made through amending the rules array in the editor.
   - **Default value:** false

5. ruleset

   - **Type:** String enumeration of the following choices:
     (a) "Default"
     (b) "More Buildings (IMPOSSIBLE)"
     (c) "More Trees"
     (d) "More Space"
   - **Documentation:** Denotes a series of pre-defined rulesets for this L-System grammar, of alphabet O (blank space), W (trees and fauna) and B (buildings), that can be chosen and then used on runtime. Can choose between a default ruleset, a ruleset that produces more buildings, a ruleset that produces more trees and a ruleset that produces more empty space.
   - **Default value:** "Default"

6. rules

   - **Type:** Array of dictionaries

- **Documentation:** The set of rules that the L-System grammar uses. Shows the "default" ruleset in the Godot editor for the user to see. If "use_custom_ruleset" is set to true, this array can be edited with a custom defined ruleset that will be used on runtime, so long as it adheres to the alphabet of O (blank space), W (trees and fauna) and B (buildings).
- **Additional information:** The "_get_ruleset" function uses the String value in "ruleset" to set the value for "rules" on runtime, *if* "use_custom_ruleset" is false (which it *is* by default).
- **Default value:** The "Default" grammar, as shown in Figure B.5.

```
[
  {
     "from": "O",
     "to": "OWO"
  },
  {
     "from": "W",
     "to": "WB"
  },
  {
     "from": "B",
     "to": "BWO"
  }
]
```

Figure B.5: The "Default" grammar used for the "rules" export variable, stored in the constant "DEFAULT" in l_system.gd. See l_system.gd for the other grammars represented as arrays of dictionaries.

B.3.2 Perlin/Simplex Noise

1. noise_type

- **Type:** String enumeration of the following choices:

(a) “Perlin”

(b) “Simplex”

(c) “Simplex Smooth”

(d) “Value”

(e) “Value Cubic”

- **Documentation:** Defines the type of noise generation algorithm to use. Equates to the “noise_type” property in FastNoiseLite.
- **Default value:** “Value Cubic”

2. fractal_type

- **Type:** Enumeration of the following choices:

(a) “Fractal None”

(b) “Fractal FBM”

(c) “Fractal Ridged”

(d) “Fractal Ping Pong”

- **Documentation:** Defines the type of method used to combine octaves of a noise image into a fractal. Directly equates to the FractalType enumeration in FastNoiseLite.
- **Default value:** “Fractal None”

3. cellular_distance_type

- **Type:** Enumeration of the following choices:

(a) “Distance Euclidean”

(b) “Distance Euclidean Squared”

(c) “Distance Manhattan”

(d) “Distance Hybrid”

- **Documentation:** Defines the function used to calculate the distance between the nearest/second-nearest point(s). Directly equates to the CellularDistanceFunction enumeration in FastNoiseLite.
- **Default value:** “Distance Euclidean”

4. noise_frequency

- **Type:** Floating point number (float) between 0.0 and 1.0 inclusive

- **Documentation:** Defines the frequency of the generated noise, the higher the frequency, the rougher and more granular the noise.
- **Additional information:** The default value for "frequency" in the "FastNoiseClass" is 0.01, resulting in very smooth and distinct noise.
- **Default value:** 0.894

5. tree_cap
    - **Type:** Floating point number (float) between -1.0 and 1.0 inclusive
    - **Documentation:** Defines the upper limit to set for painting a tree tile on a specific noise pixel. If the value returned by the "get_noise_2d" method (in FastNoiseLite) is smaller than this, then it gets painted.
    - **Default value:** -0.048
6. building_cap
    - **Type:** Floating point number (float) between -1.0 and 1.0 inclusive
    - **Documentation:** Defines the upper limit to set for painting a building tile on a specific noise pixel. If the value returned by the "get_noise_2d" method (in FastNoiseLite) is smaller than this, then it gets painted. If the value of "building_cap" is smaller than "tree_cap," then decide whether or not to paint a building cell there with "building_overtakes_tree."
    - **Default value:** -0.252
7. building_overtakes_tree
    - **Type:** Floating point number (float) between 0.0 and 0.5 inclusive
    - **Documentation:** Only used when "building_cap" is smaller than "tree_cap." Determines the probability that a building tile would be painted in a cell where a tree tile was, or could be, also painted. Whether or not the cell actually is painted over is decided on computation time.
    - **Default value:** 0.12

B.3.3 Poisson Disk Sampling

1. paint_building_probability

- **Type:** Floating point number (float) between 0.0 and 1.0 inclusive
- **Documentation:** The probability that a building gets painted at a cell in lieu of a tree. The higher this probability, the more likely a building tile gets painted instead of a tree tile.
- **Default value:** 0.125

2. point_radius
    - **Type:** Floating point number (float) between 0.5 and 2.5 inclusive
    - **Documentation:** The radius value used to measure distances between points for the algorithm. The longer the radius, the further apart points are during the algorithm's processing, and the further apart painted cells are in the game.
    - **Default value:** 1.0
3. region_size
    - **Type:** Vector2
    - **Documentation:** The size of the region in which the algorithm is performed. Set to the "default" tile map size (72, 40) in the script, shown as (0, 0) in the Godot editor. Can be changed to use a smaller region for the algorithm itself, of course resulting in less cells covered within the boundaries set for this game, though the algorithm will perform faster due to less cells being checked.
    - **Default value:**
        - **x**: The value in "x_tile_range" (72)
        - **y**: The value in "y_tile_range" (40)
4. rejection_samples
    - **Type:** Integer (int) between 1 andc 50 inclusive
    - **Documentation:** The maximum number of times a cell is checked before it is ignored. A cell can be accepted and painted on before this maximum number is reached. The higher this value, the more times a cell is checked, therefore the higher the algorithm's processing time.
    - **Default value:** 8

B.3.4 Voronoi Cells

1. distance

    - **Type:** String enumeration of the following choices:

        (a) “Euclidean distance”

        (b) “Manhattan distance”

    - **Documentation:** Determines whether or not the Euclidean or Manhattan distance formula is used for calculation of the deltas between points within Voronoi cells.

    - **Default value:** “Manhattan distance”

2. random_starting_points

    - **Type:** Integer (int) between 15 and 40 inclusive

    - **Documentation:** Determines the number of points randomly picked from at the start. Therefore, it also determines the number of cells in our Voronoi tesselation.

    - **Default value:** 20

B.3.5 The Basic L-System Demo Used to Create the Screenshots in Chapter 3.1.1

There is only one export variable for this: “choices”, which allows you to choose which one of the three provided rulesets to use. “choices” is the default ruleset, and either “deterministic” or “basic” can be chosen. It is in the “DemoNode” scene, the only scene of this Godot project. Quoting the documentation comment, this export variable “Allows you to decide which ruleset to use. See the script file for the sources of said rulesets.” The ruleset is assigned with the “set_values” function.

B.4 Running the Godot Projects

To *run* the current project in the Godot editor, go to the bar above the Inspector, Node and History tabs on the right-hand side. You will find a ▶ button which will play the main scene of the project (in the artefacts, the main scenes have already been set; if it were not already set, you would have been asked to set one). If closing the window to stop the project does not work, hit the ■ button to end it. If you want to replay the current project without having to stop it, hit the ▶ button again. Although **both** the icon **and** the colour of the ▶ button will have changed by then, it will be in the same position as before.

As described in section 3.2.1, only the close button of both the popup dialogs in the game seems to work properly for the moment, but this does not adversely affect the game functioning properly, nor does it adversely affect this project, so this is trivial.

Appendix C

Source Code

Contents

C.1 Instructions

If needed, use the table of contents provided to browse through the code listings in this section. Each listing folder will have a short description, a link to its public GitHub repository, and a listing for each readable source file. Use the .zip folder containing the project artefacts to edit and run them in Godot.

*"**I verify that I am the sole author of the programs contained in this folder, except where explicitly stated to the contrary.**" - Zishan Rahman, 21st April 2023*

C.2 GD4LSystemRPG

These are the source code listings for the L-System implementation of the 2D tile-map RPG scenario in the Godot game engine. The link to the publicly available source code repository is here: https://github.com/Zishan-Rahman/GD4LSystemRPG

C.2.1 .gitattributes

```
# Normalize EOL for all files that Git considers text files.
* text=auto eol=lf
```

C.2.2 .gitignore

```
# Godot 4+ specific ignores
.godot/
```

C.2.3 project.godot

```
; Engine configuration file.
; It's best edited using the editor UI and not directly,
; since the parameters that go here are not all obvious.
;
```

```
; Format:
;   [section] ; section goes between []
;   param=value ; assign values to parameters

config_version=5

[application]

config/name="LSystem RPG"
run/main_scene="res://tile_map.tscn"
config/features=PackedStringArray("4.0", "Forward Plus")
config/icon="res://icon.svg"

[display]

window/size/viewport_height=640

[rendering]

environment/defaults/default_clear_color=Color(0, 0, 0, 1)
```

C.2.4 l_system.tscn

```
[gd_scene load_steps=2 format=3 uid="uid://d0v18e7ms571f"]

[ext_resource type="Script" path="res://l_system.gd" id="1_elydp"]

[node name="LSystem" type="Node"]
script = ExtResource("1_elydp")
```

C.2.5 l_system.gd

```
1   extends Node
2
3   class_name LSystem
4
5   @onready var tile_map: TileMap = get_parent()
6   ## The starting string from which the grammar starts applying its
        rules. Here it may be self defined, or randomly defined when
        use_random_axiom is true.
7   @export var axiom: String = "OWB"
8   ## Uses a random axiom with the currently set grammar, computed
        upon runtime, with a length up to (but not strictly) the value
        of upper_limit. For example, if upper_limit is set to 15, the
        generated axiom can be 15 characters, or it can be just 5
        characters.
9   @export var use_random_axiom: bool = true
10  ## Defines how many characters a random axiom can have MAXIMUM.
        Only used when use_random_axiom is true.
11  @export var upper_limit: int = 10
12  ## Allows the use of a customly defined ruleset made through
        amending the rules array in the editor.
13  @export var use_custom_ruleset: bool = false
14  @onready var string: String = axiom
15  ## Denotes a series of pre-defined rulesets for this L-System
        grammar, of alphabet O (blank space), W (trees and fauna) and B
         (buildings), that can be chosen and then used on runtime. Can
        choose between a default ruleset, a ruleset that produces more
        buildings, a ruleset that produces more trees and a ruleset
        that produces more empty space.
16  @export_enum("Default", "More Buildings (IMPOSSIBLE)", "More Trees"
        , "More Space") var ruleset: String = "Default"
17  ## The set of rules that the L-System grammar uses. Shows the "
        default" ruleset in the Godot editor for the user to see. If
        use_custom_ruleset is set to true, this array can be edited
```

```
        with a custom defined ruleset that will be used on runtime, so
        long as it adheres to the alphabet of O (blank space), W (trees
         and fauna) and B (buildings).
18  @export var rules: Array[Dictionary] = DEFAULT
19
20  const DEFAULT: Array[Dictionary] = [
21      {
22          "from": "O",
23          "to": "OWO"
24      },
25      {
26          "from": "W",
27          "to": "WB"
28      },
29      {
30          "from": "B",
31          "to": "BWO"
32      }
33  ]
34  const MORE_BUILDINGS: Array[Dictionary] = [
35      {
36          "from": "O",
37          "to": "BWOB"
38      },
39      {
40          "from": "W",
41          "to": "WBOBO"
42      },
43      {
44          "from": "B",
45          "to": "BB"
46      }
47  ]
```

```
const MORE_TREES: Array[Dictionary] = [
    {
        "from": "O",
        "to": "WWO"
    },
    {
        "from": "W",
        "to": "WBWO"
    },
    {
        "from": "B",
        "to": "BWWO"
    }
]
const MORE_SPACE: Array[Dictionary] = [
    {
        "from": "O",
        "to": "OOBWO"
    },
    {
        "from": "W",
        "to": "OB"
    },
    {
        "from": "B",
        "to": "OW"
    }
]

const buildings: Array[Vector2i] = [
    Vector2i(0, 19),
    Vector2i(1, 19),
    Vector2i(2, 19),
```

```
81      Vector2i(3, 19),
82      Vector2i(4, 19),
83      Vector2i(5, 19),
84      Vector2i(6, 19),
85      Vector2i(7, 19),
86      Vector2i(8, 20),
87      Vector2i(0, 20),
88      Vector2i(1, 20),
89      Vector2i(2, 20),
90      Vector2i(3, 20),
91      Vector2i(4, 20),
92      Vector2i(5, 20),
93      Vector2i(6, 20),
94      Vector2i(7, 20),
95      Vector2i(8, 20),
96      Vector2i(0, 21),
97      Vector2i(1, 21),
98      Vector2i(2, 21),
99      Vector2i(3, 21),
100     Vector2i(4, 21),
101     Vector2i(5, 21),
102     Vector2i(6, 21),
103     Vector2i(7, 21),
104     Vector2i(8, 21)
105 ]
106 const trees: Array[Vector2i] = [
107     Vector2i(0,1),
108     Vector2i(1,1),
109     Vector2i(2,1),
110     Vector2i(3,1),
111     Vector2i(4,1),
112     Vector2i(5,1),
113     Vector2i(6,1),
```

```
    Vector2i(7,1),
    Vector2i(0,2),
    Vector2i(1,2),
    Vector2i(2,2),
    Vector2i(3,2),
    Vector2i(4,2)
]

func _get_ruleset() -> Array[Dictionary]:
    match ruleset:
        "More Buildings (IMPOSSIBLE)": return MORE_BUILDINGS
        "More Trees": return MORE_TREES
        "More Space": return MORE_SPACE
        _: return DEFAULT

func get_new_replacement(character: String) -> String:
    for rule in rules:
        if rule["from"] == character:
            return rule["to"]
    return character

func _size() -> int:
    return tile_map.x_tile_range * tile_map.y_tile_range

func rand_axiom() -> String:
    var string_buffer: String = ""
    var limit: int = randi_range(1, upper_limit)
    for i in range(limit):
        string_buffer += ["O", "W", "B"].pick_random()
    return string_buffer

func parse() -> String:
    if use_random_axiom:
```

```
147         axiom = rand_axiom()
148         string = axiom
149     if not use_custom_ruleset or ruleset != "Default":
150         rules = _get_ruleset()
151     print("Axiom length: " + str(len(axiom)))
152     var size: int = _size()
153     while len(string) <= size:
154         var new_string = ""
155         for character in string:
156             new_string += get_new_replacement(character)
157         string = new_string
158     string = string.substr(0, size)
159     return string
160 
161 func paint() -> void:
162     string = parse()
163     var i: int = -1
164     for x in range(tile_map.x_tile_range):
165         for y in range(tile_map.y_tile_range):
166             i += 1
167             if string[i] == "O": # "O" = BLANK
168                 pass # Do not paint any cell.
169             elif string[i] == "W": # "W" = TREE
170                 tile_map.set_cell(0, Vector2i(x, y), 0, trees.
                        pick_random())
171             elif string[i] == "B": # "B" = BUILDING
172                 tile_map.set_cell(0, Vector2i(x, y), 0, buildings.
                        pick_random())
```

C.2.6 tile_map.tscn

```
1 [gd_scene load_steps=6 format=3 uid="uid://bwhvtqld3yo8m"]
```

```
2
3   [ext_resource type="TileSet" uid="uid://c168x78r0tful" path="res://
        Tiles.tres" id="1_l3nwg"]
4   [ext_resource type="Script" path="res://tile_map.gd" id="2_wrxl8"]
5   [ext_resource type="PackedScene" uid="uid://d0v18e7ms571f" path="
        res://l_system.tscn" id="3_ktw1n"]
6   [ext_resource type="PackedScene" uid="uid://cau5jgogdnf53" path="
        res://accept_dialog.tscn" id="4_060oh"]
7   [ext_resource type="PackedScene" uid="uid://b5q8ovcigrvyr" path="
        res://win_dialog.tscn" id="5_3s48a"]
8
9   [node name="TileMap" type="TileMap"]
10  tile_set = ExtResource("1_l3nwg")
11  format = 2
12  layer_0/name = "Things"
13  script = ExtResource("2_wrxl8")
14
15  [node name="LSystem" parent="." instance=ExtResource("3_ktw1n")]
16
17  [node name="AcceptDialog" parent="." instance=ExtResource("4_060oh"
        )]
18
19  [node name="WinDialog" parent="." instance=ExtResource("5_3s48a")]
```

C.2.7 tile_map.gd

```
1   extends TileMap
2
3   @onready var l_system: LSystem = $LSystem
4
5   var x_tile_range: int = ProjectSettings.get_setting("display/window
        /size/viewport_width") / tile_set.tile_size.x
```

```
6   var y_tile_range: int = ProjectSettings.get_setting("display/window
        /size/viewport_height") / tile_set.tile_size.y
7
8   const PLAYER_SPRITE: Vector2i = Vector2i(24, 7)
9   var player_placement_cell: Vector2i
10  const rings: Array[Vector2i] = [
11      Vector2i(43, 6),
12      Vector2i(44, 6),
13      Vector2i(45, 6),
14      Vector2i(46, 6)
15  ]
16  var ring_placement_cell: Vector2i
17
18  # Called when the node enters the scene tree for the first time.
19  func _ready() -> void:
20      randomize()
21      var start_time: float = Time.get_ticks_msec()
22      l_system.paint()
23      place_player()
24      place_ring()
25      var new_time: float = Time.get_ticks_msec() - start_time
26      print("Time taken: " + str(new_time) + "ms")
27      $AcceptDialog.dialog_text = "You're a hollow Golem who seeks the
            ultimate treasure; a ring that's got something on top of it
            . It's somewhere in this large village and barely visible to
            your naked eyes, which took us " + str(new_time) + "
            milliseconds to generate (" + str(new_time / 1000.0) + "
            seconds), but you'll stop at nothing to get what you want.
            You can chow down every tree and fauna that stands in your
            way of the ring, but your Achilles heel is any bricks and
            mortar, which WILL make you stop at your tracks. Since it's
            easy to get lost in here, we'll tell you that you're in
            position " + str(player_placement_cell) + " in this big
```

```
        village of size " + str(Vector2i(x_tile_range, y_tile_range)
        ) + ". The ring is in position " + str(ring_placement_cell)
        + ", but it is YOUR job to traverse the village, chow down
        the trees and get it for yourself, so are you ready to
        attain the treasure that is rightfully yours?!"
28      $AcceptDialog.visible = true
29      $AcceptDialog.confirmed.connect(_on_AcceptDialog_closed)
30      $AcceptDialog.canceled.connect(_on_AcceptDialog_closed)
31      $WinDialog.confirmed.connect(_on_WinDialog_confirmed)
32      $WinDialog.canceled.connect(_on_WinDialog_canceled)
33      get_tree().paused = true
34
35  func _on_WinDialog_confirmed() -> void:
36      get_tree().reload_current_scene()
37
38  func _on_WinDialog_canceled() -> void:
39      get_tree().quit()
40
41  func _on_AcceptDialog_closed() -> void:
42      $AcceptDialog.visible = false
43      get_tree().paused = false
44
45  func _get_random_placement_cell() -> Vector2i:
46      return Vector2i(randi() % x_tile_range, randi() % y_tile_range)
47
48  func place_player() -> void:
49      player_placement_cell = _get_random_placement_cell()
50      while get_used_cells(0).has(player_placement_cell):
51          player_placement_cell = _get_random_placement_cell()
52      set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
53
54  func place_ring() -> void:
55      ring_placement_cell = _get_random_placement_cell()
```

```
56      while get_used_cells(0).has(ring_placement_cell):
57          ring_placement_cell = _get_random_placement_cell()
58      set_cell(0, ring_placement_cell, 0, rings.pick_random())
59
60  func _is_not_out_of_bounds(cell: Vector2i) -> bool:
61      return cell.x >= 0 and cell.x < x_tile_range and cell.y >= 0 and
            cell.y < y_tile_range
62
63  func _physics_process(_delta: float) -> void:
64      var previous_cell: Vector2i = player_placement_cell
65      var direction: Vector2i = Vector2i.ZERO
66      if Input.is_action_pressed("ui_up"): direction = Vector2i.UP
67      elif Input.is_action_pressed("ui_down"): direction = Vector2i.
            DOWN
68      elif Input.is_action_pressed("ui_left"): direction = Vector2i.
            LEFT
69      elif Input.is_action_pressed("ui_right"): direction = Vector2i.
            RIGHT
70      var new_placement_cell: Vector2i = player_placement_cell +
            direction
71      if (not get_used_cells(0).has(new_placement_cell) or l_system.
            trees.has(get_cell_atlas_coords(0, new_placement_cell)) or
            new_placement_cell == ring_placement_cell) and
            _is_not_out_of_bounds(new_placement_cell):
72          player_placement_cell = new_placement_cell
73          set_cell(0, previous_cell, 0) # deletes contents of previous
                cell (atlas_coords = Vector2i(-1, -1))
74          set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
75          if player_placement_cell == ring_placement_cell:
76              $WinDialog.visible = true
77              get_tree().paused = true
78
79  # ALGORITHM AND CUSTOM EXPORT VARIABLES ARE IN LSYSTEM NODE
```

C.2.8 accept_dialog.tscn

```
[gd_scene format=3 uid="uid://cau5jgogdnf53"]

[node name="AcceptDialog" type="AcceptDialog"]
title = "Tree-Munching Time!"
position = Vector2i(326, 100)
size = Vector2i(500, 421)
mouse_passthrough = true
ok_button_text = "Bring it on!"
dialog_text = "You're a hollow Golem who seeks the ultimate
    treasure; a ring that's got something on top of it. It's
    somewhere in this large village and barely visible to your
    naked eyes, but you'll stop at nothing to get what you want.
    You can chow down every tree and fauna that stands in your way
    of the ring, but your Achilles heel is any bricks and mortar,
    which will make you stop at your tracks. Are you ready to
    attain your treasure?w Golem in a black-and-white world, in
    search for your most desired treasure. It's a ring with
    something on top of it. And you'll stop at nothing to get what
    you want. You can chow down every tree and fauna that stands in
     your way of the ring, but your Achilles heel is any bricks and
     mortar, which will make you stop at your tracks. Are you ready
     to attain the treasure that is rightfully yours?!"
dialog_autowrap = true
```

C.2.9 win_dialog.tscn

```
[gd_scene format=3 uid="uid://b5q8ovcigrvyr"]

[node name="WinDialog" type="ConfirmationDialog"]
title = "You Found the Treasure!"
```

```
position = Vector2i(326, 100)
size = Vector2i(500, 421)
mouse_passthrough = true
ok_button_text = "Get Me a New Village"
dialog_text = "You found your treasure! Well done, you!

Would you like to travel to a new village in the hopes of finding another ring? Or would you like to take your treasure home now?"
dialog_autowrap = true
cancel_button_text = "Get Me Out of Here"
```

C.2.10 icon.svg.import

```
[remap]

importer="texture"
type="CompressedTexture2D"
uid="uid://b45qexb3wmhym"
path="res://.godot/imported/icon.svg-218a8f2b3041327d8a5756f3a245f83b.ctex"
metadata={
"vram_texture": false
}

[deps]

source_file="res://icon.svg"
dest_files=["res://.godot/imported/icon.svg-218a8f2b3041327d8a5756f3a245f83b.ctex"]

[params]
```

```
17
18  compress/mode=0
19  compress/high_quality=false
20  compress/lossy_quality=0.7
21  compress/hdr_compression=1
22  compress/normal_map=0
23  compress/channel_pack=0
24  mipmaps/generate=false
25  mipmaps/limit=-1
26  roughness/mode=0
27  roughness/src_normal=""
28  process/fix_alpha_border=true
29  process/premult_alpha=false
30  process/normal_map_invert_y=false
31  process/hdr_as_srgb=false
32  process/hdr_clamp_exposure=false
33  process/size_limit=0
34  detect_3d/compress_to=1
35  svg/scale=1.0
36  editor/scale_with_editor_scale=false
37  editor/convert_colors_with_editor_theme=false
```

C.2.11 monochrome_packed.png.import

```
1  [remap]
2
3  importer="texture"
4  type="CompressedTexture2D"
5  uid="uid://dic8oic1ybjyq"
6  path="res://.godot/imported/monochrome_packed.png-6
       b9bd1c64dd50f72acd3afd14d1ac34f.ctex"
7  metadata={
```

```
"vram_texture": false
}

[deps]

source_file="res://monochrome_packed.png"
dest_files=["res://.godot/imported/monochrome_packed.png-6
    b9bd1c64dd50f72acd3afd14d1ac34f.ctex"]

[params]

compress/mode=0
compress/high_quality=false
compress/lossy_quality=0.7
compress/hdr_compression=1
compress/normal_map=0
compress/channel_pack=0
mipmaps/generate=false
mipmaps/limit=-1
roughness/mode=0
roughness/src_normal=""
process/fix_alpha_border=true
process/premult_alpha=false
process/normal_map_invert_y=false
process/hdr_as_srgb=false
process/hdr_clamp_exposure=false
process/size_limit=0
detect_3d/compress_to=1
```

C.2.12 Tiles.tres

```
[gd_resource type="TileSet" load_steps=3 format=3 uid="uid://
```

```
    c168x78r0tful"]

[ext_resource type="Texture2D" uid="uid://dic8oic1ybjyq" path="res
    ://monochrome_packed.png" id="1_fqi6r"]

[sub_resource type="TileSetAtlasSource" id="
    TileSetAtlasSource_qer06"]
texture = ExtResource("1_fqi6r")
0:0/0 = 0
1:0/0 = 0
2:0/0 = 0
3:0/0 = 0
4:0/0 = 0
5:0/0 = 0
6:0/0 = 0
7:0/0 = 0
8:0/0 = 0
9:0/0 = 0
10:0/0 = 0
11:0/0 = 0
12:0/0 = 0
13:0/0 = 0
14:0/0 = 0
15:0/0 = 0
16:0/0 = 0
17:0/0 = 0
18:0/0 = 0
19:0/0 = 0
20:0/0 = 0
21:0/0 = 0
22:0/0 = 0
23:0/0 = 0
24:0/0 = 0
```

```
25:0/0 = 0
26:0/0 = 0
27:0/0 = 0
28:0/0 = 0
29:0/0 = 0
30:0/0 = 0
31:0/0 = 0
32:0/0 = 0
33:0/0 = 0
34:0/0 = 0
35:0/0 = 0
36:0/0 = 0
37:0/0 = 0
38:0/0 = 0
39:0/0 = 0
40:0/0 = 0
41:0/0 = 0
42:0/0 = 0
43:0/0 = 0
44:0/0 = 0
45:0/0 = 0
46:0/0 = 0
47:0/0 = 0
48:0/0 = 0
0:1/0 = 0
1:1/0 = 0
2:1/0 = 0
3:1/0 = 0
4:1/0 = 0
5:1/0 = 0
6:1/0 = 0
7:1/0 = 0
8:1/0 = 0
```

```
9:1/0 = 0
10:1/0 = 0
11:1/0 = 0
12:1/0 = 0
13:1/0 = 0
14:1/0 = 0
15:1/0 = 0
16:1/0 = 0
17:1/0 = 0
18:1/0 = 0
19:1/0 = 0
20:1/0 = 0
21:1/0 = 0
22:1/0 = 0
23:1/0 = 0
24:1/0 = 0
25:1/0 = 0
26:1/0 = 0
27:1/0 = 0
28:1/0 = 0
29:1/0 = 0
30:1/0 = 0
31:1/0 = 0
32:1/0 = 0
33:1/0 = 0
34:1/0 = 0
35:1/0 = 0
36:1/0 = 0
37:1/0 = 0
38:1/0 = 0
39:1/0 = 0
40:1/0 = 0
41:1/0 = 0
```

```
42:1/0 = 0
43:1/0 = 0
44:1/0 = 0
45:1/0 = 0
46:1/0 = 0
47:1/0 = 0
48:1/0 = 0
0:2/0 = 0
1:2/0 = 0
2:2/0 = 0
3:2/0 = 0
4:2/0 = 0
5:2/0 = 0
6:2/0 = 0
7:2/0 = 0
8:2/0 = 0
9:2/0 = 0
10:2/0 = 0
11:2/0 = 0
12:2/0 = 0
13:2/0 = 0
14:2/0 = 0
15:2/0 = 0
16:2/0 = 0
17:2/0 = 0
18:2/0 = 0
19:2/0 = 0
20:2/0 = 0
21:2/0 = 0
22:2/0 = 0
23:2/0 = 0
24:2/0 = 0
25:2/0 = 0
```

```
26:2/0 = 0
27:2/0 = 0
28:2/0 = 0
29:2/0 = 0
30:2/0 = 0
31:2/0 = 0
32:2/0 = 0
33:2/0 = 0
34:2/0 = 0
35:2/0 = 0
36:2/0 = 0
37:2/0 = 0
38:2/0 = 0
39:2/0 = 0
40:2/0 = 0
41:2/0 = 0
42:2/0 = 0
43:2/0 = 0
44:2/0 = 0
45:2/0 = 0
46:2/0 = 0
47:2/0 = 0
48:2/0 = 0
0:3/0 = 0
1:3/0 = 0
2:3/0 = 0
3:3/0 = 0
4:3/0 = 0
5:3/0 = 0
6:3/0 = 0
7:3/0 = 0
8:3/0 = 0
9:3/0 = 0
```

```
10:3/0 = 0
11:3/0 = 0
12:3/0 = 0
13:3/0 = 0
14:3/0 = 0
15:3/0 = 0
16:3/0 = 0
17:3/0 = 0
18:3/0 = 0
19:3/0 = 0
20:3/0 = 0
21:3/0 = 0
22:3/0 = 0
23:3/0 = 0
24:3/0 = 0
25:3/0 = 0
26:3/0 = 0
27:3/0 = 0
28:3/0 = 0
29:3/0 = 0
30:3/0 = 0
31:3/0 = 0
32:3/0 = 0
33:3/0 = 0
34:3/0 = 0
35:3/0 = 0
36:3/0 = 0
37:3/0 = 0
38:3/0 = 0
39:3/0 = 0
40:3/0 = 0
41:3/0 = 0
42:3/0 = 0
```

```
43:3/0 = 0
44:3/0 = 0
45:3/0 = 0
46:3/0 = 0
47:3/0 = 0
48:3/0 = 0
0:4/0 = 0
1:4/0 = 0
2:4/0 = 0
3:4/0 = 0
4:4/0 = 0
5:4/0 = 0
6:4/0 = 0
7:4/0 = 0
8:4/0 = 0
9:4/0 = 0
10:4/0 = 0
11:4/0 = 0
12:4/0 = 0
13:4/0 = 0
14:4/0 = 0
15:4/0 = 0
16:4/0 = 0
17:4/0 = 0
18:4/0 = 0
19:4/0 = 0
20:4/0 = 0
21:4/0 = 0
22:4/0 = 0
23:4/0 = 0
24:4/0 = 0
25:4/0 = 0
26:4/0 = 0
```

```
27:4/0 = 0
28:4/0 = 0
29:4/0 = 0
30:4/0 = 0
31:4/0 = 0
32:4/0 = 0
33:4/0 = 0
34:4/0 = 0
35:4/0 = 0
36:4/0 = 0
37:4/0 = 0
38:4/0 = 0
39:4/0 = 0
40:4/0 = 0
41:4/0 = 0
42:4/0 = 0
43:4/0 = 0
44:4/0 = 0
45:4/0 = 0
46:4/0 = 0
47:4/0 = 0
48:4/0 = 0
0:5/0 = 0
1:5/0 = 0
2:5/0 = 0
3:5/0 = 0
4:5/0 = 0
5:5/0 = 0
6:5/0 = 0
7:5/0 = 0
8:5/0 = 0
9:5/0 = 0
10:5/0 = 0
```

```
11:5/0 = 0
12:5/0 = 0
13:5/0 = 0
14:5/0 = 0
15:5/0 = 0
16:5/0 = 0
17:5/0 = 0
18:5/0 = 0
19:5/0 = 0
20:5/0 = 0
21:5/0 = 0
22:5/0 = 0
23:5/0 = 0
24:5/0 = 0
25:5/0 = 0
26:5/0 = 0
27:5/0 = 0
28:5/0 = 0
29:5/0 = 0
30:5/0 = 0
31:5/0 = 0
32:5/0 = 0
33:5/0 = 0
34:5/0 = 0
35:5/0 = 0
36:5/0 = 0
37:5/0 = 0
38:5/0 = 0
39:5/0 = 0
40:5/0 = 0
41:5/0 = 0
42:5/0 = 0
43:5/0 = 0
```

```
44:5/0 = 0
45:5/0 = 0
46:5/0 = 0
47:5/0 = 0
48:5/0 = 0
0:6/0 = 0
1:6/0 = 0
2:6/0 = 0
3:6/0 = 0
4:6/0 = 0
5:6/0 = 0
6:6/0 = 0
7:6/0 = 0
8:6/0 = 0
9:6/0 = 0
10:6/0 = 0
11:6/0 = 0
12:6/0 = 0
13:6/0 = 0
14:6/0 = 0
15:6/0 = 0
16:6/0 = 0
17:6/0 = 0
18:6/0 = 0
19:6/0 = 0
20:6/0 = 0
21:6/0 = 0
22:6/0 = 0
23:6/0 = 0
24:6/0 = 0
25:6/0 = 0
26:6/0 = 0
27:6/0 = 0
```

```
28:6/0 = 0
29:6/0 = 0
30:6/0 = 0
31:6/0 = 0
32:6/0 = 0
33:6/0 = 0
34:6/0 = 0
35:6/0 = 0
36:6/0 = 0
37:6/0 = 0
38:6/0 = 0
39:6/0 = 0
40:6/0 = 0
41:6/0 = 0
42:6/0 = 0
43:6/0 = 0
44:6/0 = 0
45:6/0 = 0
46:6/0 = 0
47:6/0 = 0
48:6/0 = 0
0:7/0 = 0
1:7/0 = 0
2:7/0 = 0
3:7/0 = 0
4:7/0 = 0
5:7/0 = 0
6:7/0 = 0
7:7/0 = 0
8:7/0 = 0
9:7/0 = 0
10:7/0 = 0
11:7/0 = 0
```

```
12:7/0 = 0
13:7/0 = 0
14:7/0 = 0
15:7/0 = 0
16:7/0 = 0
17:7/0 = 0
18:7/0 = 0
19:7/0 = 0
20:7/0 = 0
21:7/0 = 0
22:7/0 = 0
23:7/0 = 0
24:7/0 = 0
25:7/0 = 0
26:7/0 = 0
27:7/0 = 0
28:7/0 = 0
29:7/0 = 0
30:7/0 = 0
31:7/0 = 0
32:7/0 = 0
33:7/0 = 0
34:7/0 = 0
35:7/0 = 0
36:7/0 = 0
37:7/0 = 0
38:7/0 = 0
39:7/0 = 0
40:7/0 = 0
41:7/0 = 0
42:7/0 = 0
43:7/0 = 0
44:7/0 = 0
```

```
45:7/0 = 0
46:7/0 = 0
47:7/0 = 0
48:7/0 = 0
0:8/0 = 0
1:8/0 = 0
2:8/0 = 0
3:8/0 = 0
4:8/0 = 0
5:8/0 = 0
6:8/0 = 0
7:8/0 = 0
8:8/0 = 0
9:8/0 = 0
10:8/0 = 0
11:8/0 = 0
12:8/0 = 0
13:8/0 = 0
14:8/0 = 0
15:8/0 = 0
16:8/0 = 0
17:8/0 = 0
18:8/0 = 0
19:8/0 = 0
20:8/0 = 0
21:8/0 = 0
22:8/0 = 0
23:8/0 = 0
24:8/0 = 0
25:8/0 = 0
26:8/0 = 0
27:8/0 = 0
28:8/0 = 0
```

```
29:8/0 = 0
30:8/0 = 0
31:8/0 = 0
32:8/0 = 0
33:8/0 = 0
34:8/0 = 0
35:8/0 = 0
36:8/0 = 0
37:8/0 = 0
38:8/0 = 0
39:8/0 = 0
40:8/0 = 0
41:8/0 = 0
42:8/0 = 0
43:8/0 = 0
44:8/0 = 0
45:8/0 = 0
46:8/0 = 0
47:8/0 = 0
48:8/0 = 0
0:9/0 = 0
1:9/0 = 0
2:9/0 = 0
3:9/0 = 0
4:9/0 = 0
5:9/0 = 0
6:9/0 = 0
7:9/0 = 0
8:9/0 = 0
9:9/0 = 0
10:9/0 = 0
11:9/0 = 0
12:9/0 = 0
```

```
13:9/0 = 0
14:9/0 = 0
15:9/0 = 0
16:9/0 = 0
17:9/0 = 0
18:9/0 = 0
19:9/0 = 0
20:9/0 = 0
21:9/0 = 0
22:9/0 = 0
23:9/0 = 0
24:9/0 = 0
25:9/0 = 0
26:9/0 = 0
27:9/0 = 0
28:9/0 = 0
29:9/0 = 0
30:9/0 = 0
31:9/0 = 0
32:9/0 = 0
33:9/0 = 0
34:9/0 = 0
35:9/0 = 0
36:9/0 = 0
37:9/0 = 0
38:9/0 = 0
39:9/0 = 0
40:9/0 = 0
41:9/0 = 0
42:9/0 = 0
43:9/0 = 0
44:9/0 = 0
45:9/0 = 0
```

```
46:9/0 = 0
47:9/0 = 0
48:9/0 = 0
0:10/0 = 0
1:10/0 = 0
2:10/0 = 0
3:10/0 = 0
4:10/0 = 0
5:10/0 = 0
6:10/0 = 0
7:10/0 = 0
8:10/0 = 0
9:10/0 = 0
10:10/0 = 0
11:10/0 = 0
12:10/0 = 0
13:10/0 = 0
14:10/0 = 0
15:10/0 = 0
16:10/0 = 0
17:10/0 = 0
18:10/0 = 0
19:10/0 = 0
20:10/0 = 0
21:10/0 = 0
22:10/0 = 0
23:10/0 = 0
24:10/0 = 0
25:10/0 = 0
26:10/0 = 0
27:10/0 = 0
28:10/0 = 0
29:10/0 = 0
```

```
30:10/0 = 0
31:10/0 = 0
32:10/0 = 0
33:10/0 = 0
34:10/0 = 0
35:10/0 = 0
36:10/0 = 0
37:10/0 = 0
38:10/0 = 0
39:10/0 = 0
40:10/0 = 0
41:10/0 = 0
42:10/0 = 0
43:10/0 = 0
44:10/0 = 0
45:10/0 = 0
46:10/0 = 0
47:10/0 = 0
48:10/0 = 0
0:11/0 = 0
1:11/0 = 0
2:11/0 = 0
3:11/0 = 0
4:11/0 = 0
5:11/0 = 0
6:11/0 = 0
7:11/0 = 0
8:11/0 = 0
9:11/0 = 0
10:11/0 = 0
11:11/0 = 0
12:11/0 = 0
13:11/0 = 0
```

```
14:11/0 = 0
15:11/0 = 0
16:11/0 = 0
17:11/0 = 0
18:11/0 = 0
19:11/0 = 0
20:11/0 = 0
21:11/0 = 0
22:11/0 = 0
23:11/0 = 0
24:11/0 = 0
25:11/0 = 0
26:11/0 = 0
27:11/0 = 0
28:11/0 = 0
29:11/0 = 0
30:11/0 = 0
31:11/0 = 0
32:11/0 = 0
33:11/0 = 0
34:11/0 = 0
35:11/0 = 0
36:11/0 = 0
37:11/0 = 0
38:11/0 = 0
39:11/0 = 0
40:11/0 = 0
41:11/0 = 0
42:11/0 = 0
43:11/0 = 0
44:11/0 = 0
45:11/0 = 0
46:11/0 = 0
```

```
47:11/0 = 0
48:11/0 = 0
0:12/0 = 0
1:12/0 = 0
2:12/0 = 0
3:12/0 = 0
4:12/0 = 0
5:12/0 = 0
6:12/0 = 0
7:12/0 = 0
8:12/0 = 0
9:12/0 = 0
10:12/0 = 0
11:12/0 = 0
12:12/0 = 0
13:12/0 = 0
14:12/0 = 0
15:12/0 = 0
16:12/0 = 0
17:12/0 = 0
18:12/0 = 0
19:12/0 = 0
20:12/0 = 0
21:12/0 = 0
22:12/0 = 0
23:12/0 = 0
24:12/0 = 0
25:12/0 = 0
26:12/0 = 0
27:12/0 = 0
28:12/0 = 0
29:12/0 = 0
30:12/0 = 0
```

```
31:12/0 = 0
32:12/0 = 0
33:12/0 = 0
34:12/0 = 0
35:12/0 = 0
36:12/0 = 0
37:12/0 = 0
38:12/0 = 0
39:12/0 = 0
40:12/0 = 0
41:12/0 = 0
42:12/0 = 0
43:12/0 = 0
44:12/0 = 0
45:12/0 = 0
46:12/0 = 0
47:12/0 = 0
48:12/0 = 0
0:13/0 = 0
1:13/0 = 0
2:13/0 = 0
3:13/0 = 0
4:13/0 = 0
5:13/0 = 0
6:13/0 = 0
7:13/0 = 0
8:13/0 = 0
9:13/0 = 0
10:13/0 = 0
11:13/0 = 0
12:13/0 = 0
13:13/0 = 0
14:13/0 = 0
```

```
15:13/0 = 0
16:13/0 = 0
17:13/0 = 0
18:13/0 = 0
19:13/0 = 0
20:13/0 = 0
21:13/0 = 0
22:13/0 = 0
23:13/0 = 0
24:13/0 = 0
25:13/0 = 0
26:13/0 = 0
27:13/0 = 0
28:13/0 = 0
29:13/0 = 0
30:13/0 = 0
31:13/0 = 0
32:13/0 = 0
33:13/0 = 0
34:13/0 = 0
35:13/0 = 0
36:13/0 = 0
37:13/0 = 0
38:13/0 = 0
39:13/0 = 0
40:13/0 = 0
41:13/0 = 0
42:13/0 = 0
43:13/0 = 0
44:13/0 = 0
45:13/0 = 0
46:13/0 = 0
47:13/0 = 0
```

```
48:13/0 = 0
0:14/0 = 0
1:14/0 = 0
2:14/0 = 0
3:14/0 = 0
4:14/0 = 0
5:14/0 = 0
6:14/0 = 0
7:14/0 = 0
8:14/0 = 0
9:14/0 = 0
10:14/0 = 0
11:14/0 = 0
12:14/0 = 0
13:14/0 = 0
14:14/0 = 0
15:14/0 = 0
16:14/0 = 0
17:14/0 = 0
18:14/0 = 0
19:14/0 = 0
20:14/0 = 0
21:14/0 = 0
22:14/0 = 0
23:14/0 = 0
24:14/0 = 0
25:14/0 = 0
26:14/0 = 0
27:14/0 = 0
28:14/0 = 0
29:14/0 = 0
30:14/0 = 0
31:14/0 = 0
```

```
32:14/0 = 0
33:14/0 = 0
34:14/0 = 0
35:14/0 = 0
36:14/0 = 0
37:14/0 = 0
38:14/0 = 0
39:14/0 = 0
40:14/0 = 0
41:14/0 = 0
42:14/0 = 0
43:14/0 = 0
44:14/0 = 0
45:14/0 = 0
46:14/0 = 0
47:14/0 = 0
48:14/0 = 0
0:15/0 = 0
1:15/0 = 0
2:15/0 = 0
3:15/0 = 0
4:15/0 = 0
5:15/0 = 0
6:15/0 = 0
7:15/0 = 0
8:15/0 = 0
9:15/0 = 0
10:15/0 = 0
11:15/0 = 0
12:15/0 = 0
13:15/0 = 0
14:15/0 = 0
15:15/0 = 0
```

```
16:15/0 = 0
17:15/0 = 0
18:15/0 = 0
19:15/0 = 0
20:15/0 = 0
21:15/0 = 0
22:15/0 = 0
23:15/0 = 0
24:15/0 = 0
25:15/0 = 0
26:15/0 = 0
27:15/0 = 0
28:15/0 = 0
29:15/0 = 0
30:15/0 = 0
31:15/0 = 0
32:15/0 = 0
33:15/0 = 0
34:15/0 = 0
35:15/0 = 0
36:15/0 = 0
37:15/0 = 0
38:15/0 = 0
39:15/0 = 0
40:15/0 = 0
41:15/0 = 0
42:15/0 = 0
43:15/0 = 0
44:15/0 = 0
45:15/0 = 0
46:15/0 = 0
47:15/0 = 0
48:15/0 = 0
```

```
0:16/0 = 0
1:16/0 = 0
2:16/0 = 0
3:16/0 = 0
4:16/0 = 0
5:16/0 = 0
6:16/0 = 0
7:16/0 = 0
8:16/0 = 0
9:16/0 = 0
10:16/0 = 0
11:16/0 = 0
12:16/0 = 0
13:16/0 = 0
14:16/0 = 0
15:16/0 = 0
16:16/0 = 0
17:16/0 = 0
18:16/0 = 0
19:16/0 = 0
20:16/0 = 0
21:16/0 = 0
22:16/0 = 0
23:16/0 = 0
24:16/0 = 0
25:16/0 = 0
26:16/0 = 0
27:16/0 = 0
28:16/0 = 0
29:16/0 = 0
30:16/0 = 0
31:16/0 = 0
32:16/0 = 0
```

```
33:16/0 = 0
34:16/0 = 0
35:16/0 = 0
36:16/0 = 0
37:16/0 = 0
38:16/0 = 0
39:16/0 = 0
40:16/0 = 0
41:16/0 = 0
42:16/0 = 0
43:16/0 = 0
44:16/0 = 0
45:16/0 = 0
46:16/0 = 0
47:16/0 = 0
48:16/0 = 0
0:17/0 = 0
1:17/0 = 0
2:17/0 = 0
3:17/0 = 0
4:17/0 = 0
5:17/0 = 0
6:17/0 = 0
7:17/0 = 0
8:17/0 = 0
9:17/0 = 0
10:17/0 = 0
11:17/0 = 0
12:17/0 = 0
13:17/0 = 0
14:17/0 = 0
15:17/0 = 0
16:17/0 = 0
```

```
17:17/0 = 0
18:17/0 = 0
19:17/0 = 0
20:17/0 = 0
21:17/0 = 0
22:17/0 = 0
23:17/0 = 0
24:17/0 = 0
25:17/0 = 0
26:17/0 = 0
27:17/0 = 0
28:17/0 = 0
29:17/0 = 0
30:17/0 = 0
31:17/0 = 0
32:17/0 = 0
33:17/0 = 0
34:17/0 = 0
35:17/0 = 0
36:17/0 = 0
37:17/0 = 0
38:17/0 = 0
39:17/0 = 0
40:17/0 = 0
41:17/0 = 0
42:17/0 = 0
43:17/0 = 0
44:17/0 = 0
45:17/0 = 0
46:17/0 = 0
47:17/0 = 0
48:17/0 = 0
0:18/0 = 0
```

```
1:18/0 = 0
2:18/0 = 0
3:18/0 = 0
4:18/0 = 0
5:18/0 = 0
6:18/0 = 0
7:18/0 = 0
8:18/0 = 0
9:18/0 = 0
10:18/0 = 0
11:18/0 = 0
12:18/0 = 0
13:18/0 = 0
14:18/0 = 0
15:18/0 = 0
16:18/0 = 0
17:18/0 = 0
18:18/0 = 0
19:18/0 = 0
20:18/0 = 0
21:18/0 = 0
22:18/0 = 0
23:18/0 = 0
24:18/0 = 0
25:18/0 = 0
26:18/0 = 0
27:18/0 = 0
28:18/0 = 0
29:18/0 = 0
30:18/0 = 0
31:18/0 = 0
32:18/0 = 0
33:18/0 = 0
```

```
34:18/0 = 0
35:18/0 = 0
36:18/0 = 0
37:18/0 = 0
38:18/0 = 0
39:18/0 = 0
40:18/0 = 0
41:18/0 = 0
42:18/0 = 0
43:18/0 = 0
44:18/0 = 0
45:18/0 = 0
46:18/0 = 0
47:18/0 = 0
48:18/0 = 0
0:19/0 = 0
1:19/0 = 0
2:19/0 = 0
3:19/0 = 0
4:19/0 = 0
5:19/0 = 0
6:19/0 = 0
7:19/0 = 0
8:19/0 = 0
9:19/0 = 0
10:19/0 = 0
11:19/0 = 0
12:19/0 = 0
13:19/0 = 0
14:19/0 = 0
15:19/0 = 0
16:19/0 = 0
17:19/0 = 0
```

```
18:19/0 = 0
19:19/0 = 0
20:19/0 = 0
21:19/0 = 0
22:19/0 = 0
23:19/0 = 0
24:19/0 = 0
25:19/0 = 0
26:19/0 = 0
27:19/0 = 0
28:19/0 = 0
29:19/0 = 0
30:19/0 = 0
31:19/0 = 0
32:19/0 = 0
33:19/0 = 0
34:19/0 = 0
35:19/0 = 0
36:19/0 = 0
37:19/0 = 0
38:19/0 = 0
39:19/0 = 0
40:19/0 = 0
41:19/0 = 0
42:19/0 = 0
43:19/0 = 0
44:19/0 = 0
45:19/0 = 0
46:19/0 = 0
47:19/0 = 0
48:19/0 = 0
0:20/0 = 0
1:20/0 = 0
```

```
2:20/0 = 0
3:20/0 = 0
4:20/0 = 0
5:20/0 = 0
6:20/0 = 0
7:20/0 = 0
8:20/0 = 0
9:20/0 = 0
10:20/0 = 0
11:20/0 = 0
12:20/0 = 0
13:20/0 = 0
14:20/0 = 0
15:20/0 = 0
16:20/0 = 0
17:20/0 = 0
18:20/0 = 0
19:20/0 = 0
20:20/0 = 0
21:20/0 = 0
22:20/0 = 0
23:20/0 = 0
24:20/0 = 0
25:20/0 = 0
26:20/0 = 0
27:20/0 = 0
28:20/0 = 0
29:20/0 = 0
30:20/0 = 0
31:20/0 = 0
32:20/0 = 0
33:20/0 = 0
34:20/0 = 0
```

```
35:20/0 = 0
36:20/0 = 0
37:20/0 = 0
38:20/0 = 0
39:20/0 = 0
40:20/0 = 0
41:20/0 = 0
42:20/0 = 0
43:20/0 = 0
44:20/0 = 0
45:20/0 = 0
46:20/0 = 0
47:20/0 = 0
48:20/0 = 0
0:21/0 = 0
1:21/0 = 0
2:21/0 = 0
3:21/0 = 0
4:21/0 = 0
5:21/0 = 0
6:21/0 = 0
7:21/0 = 0
8:21/0 = 0
9:21/0 = 0
10:21/0 = 0
11:21/0 = 0
12:21/0 = 0
13:21/0 = 0
14:21/0 = 0
15:21/0 = 0
16:21/0 = 0
17:21/0 = 0
18:21/0 = 0
```

```
19:21/0 = 0
20:21/0 = 0
21:21/0 = 0
22:21/0 = 0
23:21/0 = 0
24:21/0 = 0
25:21/0 = 0
26:21/0 = 0
27:21/0 = 0
28:21/0 = 0
29:21/0 = 0
30:21/0 = 0
31:21/0 = 0
32:21/0 = 0
33:21/0 = 0
34:21/0 = 0
35:21/0 = 0
36:21/0 = 0
37:21/0 = 0
38:21/0 = 0
39:21/0 = 0
40:21/0 = 0
41:21/0 = 0
42:21/0 = 0
43:21/0 = 0
44:21/0 = 0
45:21/0 = 0
46:21/0 = 0
47:21/0 = 0
48:21/0 = 0

[resource]
sources/0 = SubResource("TileSetAtlasSource_qer06")
```

C.2.13 LICENSE

```
This work is licensed under the Creative Commons Attribution-
    ShareAlike 4.0 International License. To view a copy of this
    license, visit http://creativecommons.org/licenses/by-sa/4.0/
    or send a letter to Creative Commons, PO Box 1866, Mountain
    View, CA 94042, USA.
```

C.3 GD4VoronoiRPG

These are the source code listings for the Voronoi Cells implementation of the 2D tile-map RPG scenario in the Godot game engine. The link to the publicly available source code repository is here: https://github.com/Zishan-Rahman/GD4VoronoiRPG

C.3.1 .gitattributes

```
# Normalize EOL for all files that Git considers text files.
* text=auto eol=lf
```

C.3.2 .gitignore

```
# Godot 4+ specific ignores
.godot/
```

C.3.3 project.godot

```
; Engine configuration file.
; It's best edited using the editor UI and not directly,
; since the parameters that go here are not all obvious.
;
```

```
; Format:
;   [section] ; section goes between []
;   param=value ; assign values to parameters

config_version=5

[application]

config/name="Voronoi Cells"
run/main_scene="res://tile_map.tscn"
config/features=PackedStringArray("4.0", "Forward Plus")
config/icon="res://icon.svg"

[display]

window/size/viewport_height=640

[input]

reset_position={
"deadzone": 0.5,
"events": [Object(InputEventKey,"resource_local_to_scene":false,"
    resource_name":"","device":-1,"window_id":0,"alt_pressed":false
    ,"shift_pressed":false,"ctrl_pressed":false,"meta_pressed":
    false,"pressed":false,"keycode":71,"physical_keycode":0,"
    key_label":0,"unicode":103,"echo":false,"script":null)
, Object(InputEventMouseButton,"resource_local_to_scene":false,"
    resource_name":"","device":-1,"window_id":0,"alt_pressed":false
    ,"shift_pressed":false,"ctrl_pressed":false,"meta_pressed":
    false,"button_mask":2,"position":Vector2(75, 12),"
    global_position":Vector2(78, 44),"factor":1.0,"button_index
    ":2,"pressed":true,"double_click":false,"script":null)
]
```

```
}

[rendering]

environment/defaults/default_clear_color=Color(0, 0, 0, 1)
```

C.3.4 tile_map.tscn

```
[gd_scene load_steps=7 format=3 uid="uid://d6lxnr5bdh1w"]

[ext_resource type="Texture2D" uid="uid://cpign73sfbsrt" path="res
    ://monochrome_packed.png" id="1_o183d"]
[ext_resource type="Script" path="res://tile_map.gd" id="2_lf4lw"]
[ext_resource type="PackedScene" uid="uid://cau5jgogdnf53" path="
    res://accept_dialog.tscn" id="3_y08lj"]
[ext_resource type="PackedScene" uid="uid://b5q8ovcigrvyr" path="
    res://win_dialog.tscn" id="4_fkys0"]

[sub_resource type="TileSetAtlasSource" id="
    TileSetAtlasSource_6h0bd"]
texture = ExtResource("1_o183d")
0:0/0 = 0
1:0/0 = 0
2:0/0 = 0
3:0/0 = 0
4:0/0 = 0
5:0/0 = 0
6:0/0 = 0
7:0/0 = 0
8:0/0 = 0
9:0/0 = 0
10:0/0 = 0
```

```
11:0/0 = 0
12:0/0 = 0
13:0/0 = 0
14:0/0 = 0
15:0/0 = 0
16:0/0 = 0
17:0/0 = 0
18:0/0 = 0
19:0/0 = 0
20:0/0 = 0
21:0/0 = 0
22:0/0 = 0
23:0/0 = 0
24:0/0 = 0
25:0/0 = 0
26:0/0 = 0
27:0/0 = 0
28:0/0 = 0
29:0/0 = 0
30:0/0 = 0
31:0/0 = 0
32:0/0 = 0
33:0/0 = 0
34:0/0 = 0
35:0/0 = 0
36:0/0 = 0
37:0/0 = 0
38:0/0 = 0
39:0/0 = 0
40:0/0 = 0
41:0/0 = 0
42:0/0 = 0
43:0/0 = 0
```

```
44:0/0 = 0
45:0/0 = 0
46:0/0 = 0
47:0/0 = 0
48:0/0 = 0
0:1/0 = 0
1:1/0 = 0
2:1/0 = 0
3:1/0 = 0
4:1/0 = 0
5:1/0 = 0
6:1/0 = 0
7:1/0 = 0
8:1/0 = 0
9:1/0 = 0
10:1/0 = 0
11:1/0 = 0
12:1/0 = 0
13:1/0 = 0
14:1/0 = 0
15:1/0 = 0
16:1/0 = 0
17:1/0 = 0
18:1/0 = 0
19:1/0 = 0
20:1/0 = 0
21:1/0 = 0
22:1/0 = 0
23:1/0 = 0
24:1/0 = 0
25:1/0 = 0
26:1/0 = 0
27:1/0 = 0
```

```
28:1/0 = 0
29:1/0 = 0
30:1/0 = 0
31:1/0 = 0
32:1/0 = 0
33:1/0 = 0
34:1/0 = 0
35:1/0 = 0
36:1/0 = 0
37:1/0 = 0
38:1/0 = 0
39:1/0 = 0
40:1/0 = 0
41:1/0 = 0
42:1/0 = 0
43:1/0 = 0
44:1/0 = 0
45:1/0 = 0
46:1/0 = 0
47:1/0 = 0
48:1/0 = 0
0:2/0 = 0
1:2/0 = 0
2:2/0 = 0
3:2/0 = 0
4:2/0 = 0
5:2/0 = 0
6:2/0 = 0
7:2/0 = 0
8:2/0 = 0
9:2/0 = 0
10:2/0 = 0
11:2/0 = 0
```

```
12:2/0 = 0
13:2/0 = 0
14:2/0 = 0
15:2/0 = 0
16:2/0 = 0
17:2/0 = 0
18:2/0 = 0
19:2/0 = 0
20:2/0 = 0
21:2/0 = 0
22:2/0 = 0
23:2/0 = 0
24:2/0 = 0
25:2/0 = 0
26:2/0 = 0
27:2/0 = 0
28:2/0 = 0
29:2/0 = 0
30:2/0 = 0
31:2/0 = 0
32:2/0 = 0
33:2/0 = 0
34:2/0 = 0
35:2/0 = 0
36:2/0 = 0
37:2/0 = 0
38:2/0 = 0
39:2/0 = 0
40:2/0 = 0
41:2/0 = 0
42:2/0 = 0
43:2/0 = 0
44:2/0 = 0
```

```
45:2/0 = 0
46:2/0 = 0
47:2/0 = 0
48:2/0 = 0
0:3/0 = 0
1:3/0 = 0
2:3/0 = 0
3:3/0 = 0
4:3/0 = 0
5:3/0 = 0
6:3/0 = 0
7:3/0 = 0
8:3/0 = 0
9:3/0 = 0
10:3/0 = 0
11:3/0 = 0
12:3/0 = 0
13:3/0 = 0
14:3/0 = 0
15:3/0 = 0
16:3/0 = 0
17:3/0 = 0
18:3/0 = 0
19:3/0 = 0
20:3/0 = 0
21:3/0 = 0
22:3/0 = 0
23:3/0 = 0
24:3/0 = 0
25:3/0 = 0
26:3/0 = 0
27:3/0 = 0
28:3/0 = 0
```

```
29:3/0 = 0
30:3/0 = 0
31:3/0 = 0
32:3/0 = 0
33:3/0 = 0
34:3/0 = 0
35:3/0 = 0
36:3/0 = 0
37:3/0 = 0
38:3/0 = 0
39:3/0 = 0
40:3/0 = 0
41:3/0 = 0
42:3/0 = 0
43:3/0 = 0
44:3/0 = 0
45:3/0 = 0
46:3/0 = 0
47:3/0 = 0
48:3/0 = 0
0:4/0 = 0
1:4/0 = 0
2:4/0 = 0
3:4/0 = 0
4:4/0 = 0
5:4/0 = 0
6:4/0 = 0
7:4/0 = 0
8:4/0 = 0
9:4/0 = 0
10:4/0 = 0
11:4/0 = 0
12:4/0 = 0
```

```
13:4/0 = 0
14:4/0 = 0
15:4/0 = 0
16:4/0 = 0
17:4/0 = 0
18:4/0 = 0
19:4/0 = 0
20:4/0 = 0
21:4/0 = 0
22:4/0 = 0
23:4/0 = 0
24:4/0 = 0
25:4/0 = 0
26:4/0 = 0
27:4/0 = 0
28:4/0 = 0
29:4/0 = 0
30:4/0 = 0
31:4/0 = 0
32:4/0 = 0
33:4/0 = 0
34:4/0 = 0
35:4/0 = 0
36:4/0 = 0
37:4/0 = 0
38:4/0 = 0
39:4/0 = 0
40:4/0 = 0
41:4/0 = 0
42:4/0 = 0
43:4/0 = 0
44:4/0 = 0
45:4/0 = 0
```

```
46:4/0 = 0
47:4/0 = 0
48:4/0 = 0
0:5/0 = 0
1:5/0 = 0
2:5/0 = 0
3:5/0 = 0
4:5/0 = 0
5:5/0 = 0
6:5/0 = 0
7:5/0 = 0
8:5/0 = 0
9:5/0 = 0
10:5/0 = 0
11:5/0 = 0
12:5/0 = 0
13:5/0 = 0
14:5/0 = 0
15:5/0 = 0
16:5/0 = 0
17:5/0 = 0
18:5/0 = 0
19:5/0 = 0
20:5/0 = 0
21:5/0 = 0
22:5/0 = 0
23:5/0 = 0
24:5/0 = 0
25:5/0 = 0
26:5/0 = 0
27:5/0 = 0
28:5/0 = 0
29:5/0 = 0
```

```
30:5/0 = 0
31:5/0 = 0
32:5/0 = 0
33:5/0 = 0
34:5/0 = 0
35:5/0 = 0
36:5/0 = 0
37:5/0 = 0
38:5/0 = 0
39:5/0 = 0
40:5/0 = 0
41:5/0 = 0
42:5/0 = 0
43:5/0 = 0
44:5/0 = 0
45:5/0 = 0
46:5/0 = 0
47:5/0 = 0
48:5/0 = 0
0:6/0 = 0
1:6/0 = 0
2:6/0 = 0
3:6/0 = 0
4:6/0 = 0
5:6/0 = 0
6:6/0 = 0
7:6/0 = 0
8:6/0 = 0
9:6/0 = 0
10:6/0 = 0
11:6/0 = 0
12:6/0 = 0
13:6/0 = 0
```

```
14:6/0 = 0
15:6/0 = 0
16:6/0 = 0
17:6/0 = 0
18:6/0 = 0
19:6/0 = 0
20:6/0 = 0
21:6/0 = 0
22:6/0 = 0
23:6/0 = 0
24:6/0 = 0
25:6/0 = 0
26:6/0 = 0
27:6/0 = 0
28:6/0 = 0
29:6/0 = 0
30:6/0 = 0
31:6/0 = 0
32:6/0 = 0
33:6/0 = 0
34:6/0 = 0
35:6/0 = 0
36:6/0 = 0
37:6/0 = 0
38:6/0 = 0
39:6/0 = 0
40:6/0 = 0
41:6/0 = 0
42:6/0 = 0
43:6/0 = 0
44:6/0 = 0
45:6/0 = 0
46:6/0 = 0
```

```
47:6/0 = 0
48:6/0 = 0
0:7/0 = 0
1:7/0 = 0
2:7/0 = 0
3:7/0 = 0
4:7/0 = 0
5:7/0 = 0
6:7/0 = 0
7:7/0 = 0
8:7/0 = 0
9:7/0 = 0
10:7/0 = 0
11:7/0 = 0
12:7/0 = 0
13:7/0 = 0
14:7/0 = 0
15:7/0 = 0
16:7/0 = 0
17:7/0 = 0
18:7/0 = 0
19:7/0 = 0
20:7/0 = 0
21:7/0 = 0
22:7/0 = 0
23:7/0 = 0
24:7/0 = 0
25:7/0 = 0
26:7/0 = 0
27:7/0 = 0
28:7/0 = 0
29:7/0 = 0
30:7/0 = 0
```

```
31:7/0 = 0
32:7/0 = 0
33:7/0 = 0
34:7/0 = 0
35:7/0 = 0
36:7/0 = 0
37:7/0 = 0
38:7/0 = 0
39:7/0 = 0
40:7/0 = 0
41:7/0 = 0
42:7/0 = 0
43:7/0 = 0
44:7/0 = 0
45:7/0 = 0
46:7/0 = 0
47:7/0 = 0
48:7/0 = 0
0:8/0 = 0
1:8/0 = 0
2:8/0 = 0
3:8/0 = 0
4:8/0 = 0
5:8/0 = 0
6:8/0 = 0
7:8/0 = 0
8:8/0 = 0
9:8/0 = 0
10:8/0 = 0
11:8/0 = 0
12:8/0 = 0
13:8/0 = 0
14:8/0 = 0
```

```
15:8/0 = 0
16:8/0 = 0
17:8/0 = 0
18:8/0 = 0
19:8/0 = 0
20:8/0 = 0
21:8/0 = 0
22:8/0 = 0
23:8/0 = 0
24:8/0 = 0
25:8/0 = 0
26:8/0 = 0
27:8/0 = 0
28:8/0 = 0
29:8/0 = 0
30:8/0 = 0
31:8/0 = 0
32:8/0 = 0
33:8/0 = 0
34:8/0 = 0
35:8/0 = 0
36:8/0 = 0
37:8/0 = 0
38:8/0 = 0
39:8/0 = 0
40:8/0 = 0
41:8/0 = 0
42:8/0 = 0
43:8/0 = 0
44:8/0 = 0
45:8/0 = 0
46:8/0 = 0
47:8/0 = 0
```

```
48:8/0 = 0
0:9/0 = 0
1:9/0 = 0
2:9/0 = 0
3:9/0 = 0
4:9/0 = 0
5:9/0 = 0
6:9/0 = 0
7:9/0 = 0
8:9/0 = 0
9:9/0 = 0
10:9/0 = 0
11:9/0 = 0
12:9/0 = 0
13:9/0 = 0
14:9/0 = 0
15:9/0 = 0
16:9/0 = 0
17:9/0 = 0
18:9/0 = 0
19:9/0 = 0
20:9/0 = 0
21:9/0 = 0
22:9/0 = 0
23:9/0 = 0
24:9/0 = 0
25:9/0 = 0
26:9/0 = 0
27:9/0 = 0
28:9/0 = 0
29:9/0 = 0
30:9/0 = 0
31:9/0 = 0
```

```
32:9/0 = 0
33:9/0 = 0
34:9/0 = 0
35:9/0 = 0
36:9/0 = 0
37:9/0 = 0
38:9/0 = 0
39:9/0 = 0
40:9/0 = 0
41:9/0 = 0
42:9/0 = 0
43:9/0 = 0
44:9/0 = 0
45:9/0 = 0
46:9/0 = 0
47:9/0 = 0
48:9/0 = 0
0:10/0 = 0
1:10/0 = 0
2:10/0 = 0
3:10/0 = 0
4:10/0 = 0
5:10/0 = 0
6:10/0 = 0
7:10/0 = 0
8:10/0 = 0
9:10/0 = 0
10:10/0 = 0
11:10/0 = 0
12:10/0 = 0
13:10/0 = 0
14:10/0 = 0
15:10/0 = 0
```

```
16:10/0 = 0
17:10/0 = 0
18:10/0 = 0
19:10/0 = 0
20:10/0 = 0
21:10/0 = 0
22:10/0 = 0
23:10/0 = 0
24:10/0 = 0
25:10/0 = 0
26:10/0 = 0
27:10/0 = 0
28:10/0 = 0
29:10/0 = 0
30:10/0 = 0
31:10/0 = 0
32:10/0 = 0
33:10/0 = 0
34:10/0 = 0
35:10/0 = 0
36:10/0 = 0
37:10/0 = 0
38:10/0 = 0
39:10/0 = 0
40:10/0 = 0
41:10/0 = 0
42:10/0 = 0
43:10/0 = 0
44:10/0 = 0
45:10/0 = 0
46:10/0 = 0
47:10/0 = 0
48:10/0 = 0
```

```
0:11/0 = 0
1:11/0 = 0
2:11/0 = 0
3:11/0 = 0
4:11/0 = 0
5:11/0 = 0
6:11/0 = 0
7:11/0 = 0
8:11/0 = 0
9:11/0 = 0
10:11/0 = 0
11:11/0 = 0
12:11/0 = 0
13:11/0 = 0
14:11/0 = 0
15:11/0 = 0
16:11/0 = 0
17:11/0 = 0
18:11/0 = 0
19:11/0 = 0
20:11/0 = 0
21:11/0 = 0
22:11/0 = 0
23:11/0 = 0
24:11/0 = 0
25:11/0 = 0
26:11/0 = 0
27:11/0 = 0
28:11/0 = 0
29:11/0 = 0
30:11/0 = 0
31:11/0 = 0
32:11/0 = 0
```

```
33:11/0 = 0
34:11/0 = 0
35:11/0 = 0
36:11/0 = 0
37:11/0 = 0
38:11/0 = 0
39:11/0 = 0
40:11/0 = 0
41:11/0 = 0
42:11/0 = 0
43:11/0 = 0
44:11/0 = 0
45:11/0 = 0
46:11/0 = 0
47:11/0 = 0
48:11/0 = 0
0:12/0 = 0
1:12/0 = 0
2:12/0 = 0
3:12/0 = 0
4:12/0 = 0
5:12/0 = 0
6:12/0 = 0
7:12/0 = 0
8:12/0 = 0
9:12/0 = 0
10:12/0 = 0
11:12/0 = 0
12:12/0 = 0
13:12/0 = 0
14:12/0 = 0
15:12/0 = 0
16:12/0 = 0
```

```
615    17:12/0 = 0
616    18:12/0 = 0
617    19:12/0 = 0
618    20:12/0 = 0
619    21:12/0 = 0
620    22:12/0 = 0
621    23:12/0 = 0
622    24:12/0 = 0
623    25:12/0 = 0
624    26:12/0 = 0
625    27:12/0 = 0
626    28:12/0 = 0
627    29:12/0 = 0
628    30:12/0 = 0
629    31:12/0 = 0
630    32:12/0 = 0
631    33:12/0 = 0
632    34:12/0 = 0
633    35:12/0 = 0
634    36:12/0 = 0
635    37:12/0 = 0
636    38:12/0 = 0
637    39:12/0 = 0
638    40:12/0 = 0
639    41:12/0 = 0
640    42:12/0 = 0
641    43:12/0 = 0
642    44:12/0 = 0
643    45:12/0 = 0
644    46:12/0 = 0
645    47:12/0 = 0
646    48:12/0 = 0
647    0:13/0 = 0
```

```
1:13/0 = 0
2:13/0 = 0
3:13/0 = 0
4:13/0 = 0
5:13/0 = 0
6:13/0 = 0
7:13/0 = 0
8:13/0 = 0
9:13/0 = 0
10:13/0 = 0
11:13/0 = 0
12:13/0 = 0
13:13/0 = 0
14:13/0 = 0
15:13/0 = 0
16:13/0 = 0
17:13/0 = 0
18:13/0 = 0
19:13/0 = 0
20:13/0 = 0
21:13/0 = 0
22:13/0 = 0
23:13/0 = 0
24:13/0 = 0
25:13/0 = 0
26:13/0 = 0
27:13/0 = 0
28:13/0 = 0
29:13/0 = 0
30:13/0 = 0
31:13/0 = 0
32:13/0 = 0
33:13/0 = 0
```

```
34:13/0 = 0
35:13/0 = 0
36:13/0 = 0
37:13/0 = 0
38:13/0 = 0
39:13/0 = 0
40:13/0 = 0
41:13/0 = 0
42:13/0 = 0
43:13/0 = 0
44:13/0 = 0
45:13/0 = 0
46:13/0 = 0
47:13/0 = 0
48:13/0 = 0
0:14/0 = 0
1:14/0 = 0
2:14/0 = 0
3:14/0 = 0
4:14/0 = 0
5:14/0 = 0
6:14/0 = 0
7:14/0 = 0
8:14/0 = 0
9:14/0 = 0
10:14/0 = 0
11:14/0 = 0
12:14/0 = 0
13:14/0 = 0
14:14/0 = 0
15:14/0 = 0
16:14/0 = 0
17:14/0 = 0
```

```
18:14/0 = 0
19:14/0 = 0
20:14/0 = 0
21:14/0 = 0
22:14/0 = 0
23:14/0 = 0
24:14/0 = 0
25:14/0 = 0
26:14/0 = 0
27:14/0 = 0
28:14/0 = 0
29:14/0 = 0
30:14/0 = 0
31:14/0 = 0
32:14/0 = 0
33:14/0 = 0
34:14/0 = 0
35:14/0 = 0
36:14/0 = 0
37:14/0 = 0
38:14/0 = 0
39:14/0 = 0
40:14/0 = 0
41:14/0 = 0
42:14/0 = 0
43:14/0 = 0
44:14/0 = 0
45:14/0 = 0
46:14/0 = 0
47:14/0 = 0
48:14/0 = 0
0:15/0 = 0
1:15/0 = 0
```

```
747   2:15/0 = 0
748   3:15/0 = 0
749   4:15/0 = 0
750   5:15/0 = 0
751   6:15/0 = 0
752   7:15/0 = 0
753   8:15/0 = 0
754   9:15/0 = 0
755   10:15/0 = 0
756   11:15/0 = 0
757   12:15/0 = 0
758   13:15/0 = 0
759   14:15/0 = 0
760   15:15/0 = 0
761   16:15/0 = 0
762   17:15/0 = 0
763   18:15/0 = 0
764   19:15/0 = 0
765   20:15/0 = 0
766   21:15/0 = 0
767   22:15/0 = 0
768   23:15/0 = 0
769   24:15/0 = 0
770   25:15/0 = 0
771   26:15/0 = 0
772   27:15/0 = 0
773   28:15/0 = 0
774   29:15/0 = 0
775   30:15/0 = 0
776   31:15/0 = 0
777   32:15/0 = 0
778   33:15/0 = 0
779   34:15/0 = 0
```

```
35:15/0 = 0
36:15/0 = 0
37:15/0 = 0
38:15/0 = 0
39:15/0 = 0
40:15/0 = 0
41:15/0 = 0
42:15/0 = 0
43:15/0 = 0
44:15/0 = 0
45:15/0 = 0
46:15/0 = 0
47:15/0 = 0
48:15/0 = 0
0:16/0 = 0
1:16/0 = 0
2:16/0 = 0
3:16/0 = 0
4:16/0 = 0
5:16/0 = 0
6:16/0 = 0
7:16/0 = 0
8:16/0 = 0
9:16/0 = 0
10:16/0 = 0
11:16/0 = 0
12:16/0 = 0
13:16/0 = 0
14:16/0 = 0
15:16/0 = 0
16:16/0 = 0
17:16/0 = 0
18:16/0 = 0
```

```
19:16/0 = 0
20:16/0 = 0
21:16/0 = 0
22:16/0 = 0
23:16/0 = 0
24:16/0 = 0
25:16/0 = 0
26:16/0 = 0
27:16/0 = 0
28:16/0 = 0
29:16/0 = 0
30:16/0 = 0
31:16/0 = 0
32:16/0 = 0
33:16/0 = 0
34:16/0 = 0
35:16/0 = 0
36:16/0 = 0
37:16/0 = 0
38:16/0 = 0
39:16/0 = 0
40:16/0 = 0
41:16/0 = 0
42:16/0 = 0
43:16/0 = 0
44:16/0 = 0
45:16/0 = 0
46:16/0 = 0
47:16/0 = 0
48:16/0 = 0
0:17/0 = 0
1:17/0 = 0
2:17/0 = 0
```

```
3:17/0 = 0
4:17/0 = 0
5:17/0 = 0
6:17/0 = 0
7:17/0 = 0
8:17/0 = 0
9:17/0 = 0
10:17/0 = 0
11:17/0 = 0
12:17/0 = 0
13:17/0 = 0
14:17/0 = 0
15:17/0 = 0
16:17/0 = 0
17:17/0 = 0
18:17/0 = 0
19:17/0 = 0
20:17/0 = 0
21:17/0 = 0
22:17/0 = 0
23:17/0 = 0
24:17/0 = 0
25:17/0 = 0
26:17/0 = 0
27:17/0 = 0
28:17/0 = 0
29:17/0 = 0
30:17/0 = 0
31:17/0 = 0
32:17/0 = 0
33:17/0 = 0
34:17/0 = 0
35:17/0 = 0
```

```
36:17/0 = 0
37:17/0 = 0
38:17/0 = 0
39:17/0 = 0
40:17/0 = 0
41:17/0 = 0
42:17/0 = 0
43:17/0 = 0
44:17/0 = 0
45:17/0 = 0
46:17/0 = 0
47:17/0 = 0
48:17/0 = 0
0:18/0 = 0
1:18/0 = 0
2:18/0 = 0
3:18/0 = 0
4:18/0 = 0
5:18/0 = 0
6:18/0 = 0
7:18/0 = 0
8:18/0 = 0
9:18/0 = 0
10:18/0 = 0
11:18/0 = 0
12:18/0 = 0
13:18/0 = 0
14:18/0 = 0
15:18/0 = 0
16:18/0 = 0
17:18/0 = 0
18:18/0 = 0
19:18/0 = 0
```

```
20:18/0 = 0
21:18/0 = 0
22:18/0 = 0
23:18/0 = 0
24:18/0 = 0
25:18/0 = 0
26:18/0 = 0
27:18/0 = 0
28:18/0 = 0
29:18/0 = 0
30:18/0 = 0
31:18/0 = 0
32:18/0 = 0
33:18/0 = 0
34:18/0 = 0
35:18/0 = 0
36:18/0 = 0
37:18/0 = 0
38:18/0 = 0
39:18/0 = 0
40:18/0 = 0
41:18/0 = 0
42:18/0 = 0
43:18/0 = 0
44:18/0 = 0
45:18/0 = 0
46:18/0 = 0
47:18/0 = 0
48:18/0 = 0
0:19/0 = 0
1:19/0 = 0
2:19/0 = 0
3:19/0 = 0
```

```
4:19/0 = 0
5:19/0 = 0
6:19/0 = 0
7:19/0 = 0
8:19/0 = 0
9:19/0 = 0
10:19/0 = 0
11:19/0 = 0
12:19/0 = 0
13:19/0 = 0
14:19/0 = 0
15:19/0 = 0
16:19/0 = 0
17:19/0 = 0
18:19/0 = 0
19:19/0 = 0
20:19/0 = 0
21:19/0 = 0
22:19/0 = 0
23:19/0 = 0
24:19/0 = 0
25:19/0 = 0
26:19/0 = 0
27:19/0 = 0
28:19/0 = 0
29:19/0 = 0
30:19/0 = 0
31:19/0 = 0
32:19/0 = 0
33:19/0 = 0
34:19/0 = 0
35:19/0 = 0
36:19/0 = 0
```

```
37:19/0 = 0
38:19/0 = 0
39:19/0 = 0
40:19/0 = 0
41:19/0 = 0
42:19/0 = 0
43:19/0 = 0
44:19/0 = 0
45:19/0 = 0
46:19/0 = 0
47:19/0 = 0
48:19/0 = 0
0:20/0 = 0
1:20/0 = 0
2:20/0 = 0
3:20/0 = 0
4:20/0 = 0
5:20/0 = 0
6:20/0 = 0
7:20/0 = 0
8:20/0 = 0
9:20/0 = 0
10:20/0 = 0
11:20/0 = 0
12:20/0 = 0
13:20/0 = 0
14:20/0 = 0
15:20/0 = 0
16:20/0 = 0
17:20/0 = 0
18:20/0 = 0
19:20/0 = 0
20:20/0 = 0
```

```
21:20/0 = 0
22:20/0 = 0
23:20/0 = 0
24:20/0 = 0
25:20/0 = 0
26:20/0 = 0
27:20/0 = 0
28:20/0 = 0
29:20/0 = 0
30:20/0 = 0
31:20/0 = 0
32:20/0 = 0
33:20/0 = 0
34:20/0 = 0
35:20/0 = 0
36:20/0 = 0
37:20/0 = 0
38:20/0 = 0
39:20/0 = 0
40:20/0 = 0
41:20/0 = 0
42:20/0 = 0
43:20/0 = 0
44:20/0 = 0
45:20/0 = 0
46:20/0 = 0
47:20/0 = 0
48:20/0 = 0
0:21/0 = 0
1:21/0 = 0
2:21/0 = 0
3:21/0 = 0
4:21/0 = 0
```

```
5:21/0 = 0
6:21/0 = 0
7:21/0 = 0
8:21/0 = 0
9:21/0 = 0
10:21/0 = 0
11:21/0 = 0
12:21/0 = 0
13:21/0 = 0
14:21/0 = 0
15:21/0 = 0
16:21/0 = 0
17:21/0 = 0
18:21/0 = 0
19:21/0 = 0
20:21/0 = 0
21:21/0 = 0
22:21/0 = 0
23:21/0 = 0
24:21/0 = 0
25:21/0 = 0
26:21/0 = 0
27:21/0 = 0
28:21/0 = 0
29:21/0 = 0
30:21/0 = 0
31:21/0 = 0
32:21/0 = 0
33:21/0 = 0
34:21/0 = 0
35:21/0 = 0
36:21/0 = 0
37:21/0 = 0
```

```
38:21/0 = 0
39:21/0 = 0
40:21/0 = 0
41:21/0 = 0
42:21/0 = 0
43:21/0 = 0
44:21/0 = 0
45:21/0 = 0
46:21/0 = 0
47:21/0 = 0
48:21/0 = 0

[sub_resource type="TileSet" id="TileSet_3drs5"]
sources/0 = SubResource("TileSetAtlasSource_6h0bd")

[node name="TileMap" type="TileMap"]
tile_set = SubResource("TileSet_3drs5")
format = 2
script = ExtResource("2_lf4lw")

[node name="AcceptDialog" parent="." instance=ExtResource("3_y08lj"
    )]

[node name="WinDialog" parent="." instance=ExtResource("4_fkys0")]
```

C.3.5 tile_map.gd

```
extends TileMap

const buildings: Array[Vector2i] = [
    Vector2i(0, 19),
    Vector2i(1, 19),
```

```
    Vector2i(2, 19),
    Vector2i(3, 19),
    Vector2i(4, 19),
    Vector2i(5, 19),
    Vector2i(6, 19),
    Vector2i(7, 19),
    Vector2i(8, 20),
    Vector2i(0, 20),
    Vector2i(1, 20),
    Vector2i(2, 20),
    Vector2i(3, 20),
    Vector2i(4, 20),
    Vector2i(5, 20),
    Vector2i(6, 20),
    Vector2i(7, 20),
    Vector2i(8, 20),
    Vector2i(0, 21),
    Vector2i(1, 21),
    Vector2i(2, 21),
    Vector2i(3, 21),
    Vector2i(4, 21),
    Vector2i(5, 21),
    Vector2i(6, 21),
    Vector2i(7, 21),
    Vector2i(8, 21)
]
const trees: Array[Vector2i] = [
    Vector2i(0,1),
    Vector2i(1,1),
    Vector2i(2,1),
    Vector2i(3,1),
    Vector2i(4,1),
    Vector2i(5,1),
```

```
39      Vector2i(6,1),
40      Vector2i(7,1),
41      Vector2i(0,2),
42      Vector2i(1,2),
43      Vector2i(2,2),
44      Vector2i(3,2),
45      Vector2i(4,2)
46  ]
47  const PLAYER_SPRITE: Vector2i = Vector2i(24, 7)
48  var player_placement_cell: Vector2i
49  const rings: Array[Vector2i] = [
50      Vector2i(43, 6),
51      Vector2i(44, 6),
52      Vector2i(45, 6),
53      Vector2i(46, 6)
54  ]
55  var ring_placement_cell: Vector2i
56
57  var points: Array[Dictionary] = []
58  const EUCLIDEAN: String = "Euclidean distance"
59  const MANHATTAN: String = "Manhattan distance"
60  ## Determines whether or not the Euclidean or Manhattan distance
        formula is used for calculation of the deltas between points
        within Voronoi cells.
61  @export_enum(EUCLIDEAN, MANHATTAN) var distance: String = MANHATTAN
62  ## Determines the number of points randomly picked from at the
        start. Therefore, it also determines the number of cells in our
         Voronoi tesselation.
63  @export_range(15, 40, 1) var random_starting_points: int = 20
64  var x_tile_range: int = ProjectSettings.get_setting("display/window
        /size/viewport_width") / tile_set.tile_size.x
65  var y_tile_range: int = ProjectSettings.get_setting("display/window
        /size/viewport_height") / tile_set.tile_size.y
```

```
66
67  # Called when the node enters the scene tree for the first time.
68  func _ready() -> void:
69      randomize()
70      var start_time: float = Time.get_ticks_msec()
71      define_points(random_starting_points)
72      paint_points()
73      place_player()
74      place_ring()
75      var new_time: float = Time.get_ticks_msec() - start_time
76      print("Time taken: " + str(new_time) + "ms")
77      $AcceptDialog.dialog_text = "You're a hollow Golem who seeks the
            ultimate treasure; a ring that's got something on top of it
           . It's somewhere in this large village and barely visible to
            your naked eyes, which took us " + str(new_time) + "
           milliseconds to generate (" + str(new_time / 1000.0) + "
           seconds), but you'll stop at nothing to get what you want.
           You can chow down every tree and fauna that stands in your
           way of the ring, but your Achilles heel is any bricks and
           mortar, which WILL make you stop at your tracks. Since it's
           easy to get lost in here, we'll tell you that you're in
           position " + str(player_placement_cell) + " in this big
           village of size " + str(Vector2i(x_tile_range, y_tile_range)
           ) + ". It's also easy to get stuck here, so either press the
            G key or right click to teleport somewhere else where there
            is fauna (or even the ring!!), because this game actually
           WANTS you to win it. Ultimately, though, it is YOUR job to
           find the ring, so are you ready to attain the treasure that
           is rightfully yours?!"
78      $AcceptDialog.visible = true
79      $AcceptDialog.confirmed.connect(_on_AcceptDialog_closed)
80      $AcceptDialog.canceled.connect(_on_AcceptDialog_closed)
81      $WinDialog.confirmed.connect(_on_WinDialog_confirmed)
```

```
82      $WinDialog.canceled.connect(_on_WinDialog_canceled)
83      get_tree().paused = true
84
85  func _on_WinDialog_confirmed() -> void:
86      get_tree().reload_current_scene()
87
88  func _on_WinDialog_canceled() -> void:
89      get_tree().quit()
90
91  func _on_AcceptDialog_closed() -> void:
92      $AcceptDialog.visible = false
93      get_tree().paused = false
94
95  func _get_random_placement_cell() -> Vector2i:
96      return Vector2i(randi() % x_tile_range, randi() % y_tile_range)
97
98  func place_player() -> void:
99      player_placement_cell = _get_random_placement_cell()
100     while buildings.has(get_cell_atlas_coords(0,
            player_placement_cell)) or player_placement_cell ==
            ring_placement_cell:
101         player_placement_cell = _get_random_placement_cell()
102     set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
103
104 func place_ring() -> void:
105     ring_placement_cell = _get_random_placement_cell()
106     while buildings.has(get_cell_atlas_coords(0, ring_placement_cell
            )) or ring_placement_cell == player_placement_cell:
107         ring_placement_cell = _get_random_placement_cell()
108     set_cell(0, ring_placement_cell, 0, rings.pick_random())
109
110 func _is_not_out_of_bounds(cell: Vector2i) -> bool:
111     return cell.x >= 0 and cell.x < x_tile_range and cell.y >= 0 and
```

```
            cell.y < y_tile_range
112
113 func _physics_process(_delta) -> void:
114     var previous_cell: Vector2i = player_placement_cell
115     var direction: Vector2i = Vector2i.ZERO
116     if Input.is_action_pressed("ui_up"): direction = Vector2i.UP
117     elif Input.is_action_pressed("ui_down"): direction = Vector2i.
            DOWN
118     elif Input.is_action_pressed("ui_left"): direction = Vector2i.
            LEFT
119     elif Input.is_action_pressed("ui_right"): direction = Vector2i.
            RIGHT
120     elif Input.is_action_just_pressed("reset_position"): # Respawn
            player in a different part of the map
121         player_placement_cell = _get_random_placement_cell()
122         while buildings.has(get_cell_atlas_coords(0,
                player_placement_cell)): # This time, since we're not
                STARTING the game, we don't care whether or not the
                player magically lands on the ring
123             player_placement_cell = _get_random_placement_cell()
124         set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
125         set_cell(0, previous_cell, 0) # replace the previous sprite
126         _win_game_if_won()
127         return
128     var new_placement_cell: Vector2i = player_placement_cell +
            direction
129     if (not get_used_cells(0).has(new_placement_cell) or trees.has(
            get_cell_atlas_coords(0, new_placement_cell)) or
            new_placement_cell == ring_placement_cell) and
            _is_not_out_of_bounds(new_placement_cell):
130         player_placement_cell = new_placement_cell
131         set_cell(0, previous_cell, 0) # deletes contents of previous
                cell (atlas_coords = Vector2i(-1, -1))
```

```
132         set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
133         _win_game_if_won()
134 
135 func _win_game_if_won() -> void:
136     if player_placement_cell == ring_placement_cell:
137         $WinDialog.visible = true
138         get_tree().paused = true
139 
140 # ALGORITHM BEGINS HERE
141 
142 # Used as inspiration: http://pcg.wikidot.com/pcg-algorithm:voronoi
        -diagram
143 # (brute-force implementation in JavaScript, here adapted to
        GDScript)
144 
145 func paint_points() -> void:
146     for point in points:
147         set_cell(0, Vector2(point["x"], point["y"]), 0, point["type"
              ])
148         for citizen in point["citizens"]:
149             if _is_in_bounds(point["x"], citizen["dx"], point["y"],
                  citizen["dy"]):
150                 set_cell(0, Vector2(point["x"] + citizen["dx"], point["
                      y"] + citizen["dy"]), 0, point["type"])
151 
152 func _is_in_bounds(x: int, dx: int, y: int, dy: int) -> bool:
153     return x + dx >= 0 and x + dx < x_tile_range and y + dy >= 0 and
            y + dy < y_tile_range
154 
155 func _squared(x: int) -> int:
156     return x ** 2
157 
158 func calculate_points_delta(x: int, y: int, p: int) -> float:
```

```
159     if distance == EUCLIDEAN:
160         return sqrt(_squared(points[p]["x"] - x) + _squared(points[p
                ]["y"] - y))
161     return abs(points[p]["x"] - x) + abs(points[p]["y"] - y)
162
163 func define_points(num_points: int) -> void:
164     var types: Array[Vector2i] = trees.duplicate()
165     types.append_array(buildings)
166     for i in range(num_points):
167         var x: int = randi_range(0, x_tile_range)
168         var y: int = randi_range(0, y_tile_range)
169         var type: Vector2i = types.pick_random()
170         types.erase(type)
171         points.append(
172             {
173                 "type": type,
174                 "x": x,
175                 "y": y,
176                 "citizens": []
177             }
178         )
179     for x in range(x_tile_range):
180         for y in range(y_tile_range):
181             var lowest_delta: Dictionary = {
182                 "point_id": 0,
183                 "delta": x_tile_range * y_tile_range
184             }
185             for p in range(len(points)):
186                 var delta: float = calculate_points_delta(x, y, p)
187                 if delta < lowest_delta["delta"]:
188                     lowest_delta = {
189                         "point_id": p,
190                         "delta": delta
```

```
191                 }
192             var active_point: Dictionary = points[lowest_delta["
                    point_id"]]
193             var dx: int = x - active_point["x"]
194             var dy: int = y - active_point["y"]
195             active_point["citizens"].append(
196                 {
197                     "dx": dx,
198                     "dy": dy
199                 }
200             )
```

C.3.6 accept_dialog.tscn

```
1  [gd_scene format=3 uid="uid://cau5jgogdnf53"]
2
3  [node name="AcceptDialog" type="AcceptDialog"]
4  title = "Tree-Munching Time!"
5  position = Vector2i(326, 100)
6  size = Vector2i(500, 421)
7  mouse_passthrough = true
8  ok_button_text = "Bring it on!"
9  dialog_text = "You're a hollow Golem who seeks the ultimate
       treasure; a ring that's got something on top of it. It's
       somewhere in this large village and barely visible to your
       naked eyes, but you'll stop at nothing to get what you want.
       You can chow down every tree and fauna that stands in your way
       of the ring, but your Achilles heel is any bricks and mortar,
       which will make you stop at your tracks. Are you ready to
       attain your treasure?w Golem in a black-and-white world, in
       search for your most desired treasure. It's a ring with
       something on top of it. And you'll stop at nothing to get what
```

```
    you want. You can chow down every tree and fauna that stands in
     your way of the ring, but your Achilles heel is any bricks and
     mortar, which will make you stop at your tracks. Are you ready
     to attain the treasure that is rightfully yours?!"
dialog_autowrap = true
```

C.3.7 win_dialog.tscn

```
[gd_scene format=3 uid="uid://b5q8ovcigrvyr"]

[node name="WinDialog" type="ConfirmationDialog"]
title = "You Found the Treasure!"
position = Vector2i(326, 100)
size = Vector2i(500, 421)
mouse_passthrough = true
ok_button_text = "Get Me a New Village"
dialog_text = "You found your treasure! Well done, you!

Would you like to travel to a new village in the hopes of finding
    another ring? Or would you like to take your treasure home now?
    "
dialog_autowrap = true
cancel_button_text = "Get Me Out of Here"
```

C.3.8 icon.svg.import

```
[remap]

importer="texture"
type="CompressedTexture2D"
uid="uid://du4v6taw8ssax"
```

```
path="res://.godot/imported/icon.svg-218
    a8f2b3041327d8a5756f3a245f83b.ctex"
metadata={
"vram_texture": false
}

[deps]

source_file="res://icon.svg"
dest_files=["res://.godot/imported/icon.svg-218
    a8f2b3041327d8a5756f3a245f83b.ctex"]

[params]

compress/mode=0
compress/high_quality=false
compress/lossy_quality=0.7
compress/hdr_compression=1
compress/normal_map=0
compress/channel_pack=0
mipmaps/generate=false
mipmaps/limit=-1
roughness/mode=0
roughness/src_normal=""
process/fix_alpha_border=true
process/premult_alpha=false
process/normal_map_invert_y=false
process/hdr_as_srgb=false
process/hdr_clamp_exposure=false
process/size_limit=0
detect_3d/compress_to=1
svg/scale=1.0
editor/scale_with_editor_scale=false
```

```
editor/convert_colors_with_editor_theme=false
```

C.3.9 monochrome_packed.png.import

```
[remap]

importer="texture"
type="CompressedTexture2D"
uid="uid://cpign73sfbsrt"
path="res://.godot/imported/monochrome_packed.png-6
    b9bd1c64dd50f72acd3afd14d1ac34f.ctex"
metadata={
"vram_texture": false
}

[deps]

source_file="res://monochrome_packed.png"
dest_files=["res://.godot/imported/monochrome_packed.png-6
    b9bd1c64dd50f72acd3afd14d1ac34f.ctex"]

[params]

compress/mode=0
compress/high_quality=false
compress/lossy_quality=0.7
compress/hdr_compression=1
compress/normal_map=0
compress/channel_pack=0
mipmaps/generate=false
mipmaps/limit=-1
roughness/mode=0
```

```
27  roughness/src_normal=""
28  process/fix_alpha_border=true
29  process/premult_alpha=false
30  process/normal_map_invert_y=false
31  process/hdr_as_srgb=false
32  process/hdr_clamp_exposure=false
33  process/size_limit=0
34  detect_3d/compress_to=1
```

C.3.10 LICENSE

```
1  This work is licensed under the Creative Commons Attribution-
       ShareAlike 4.0 International License. To view a copy of this
       license, visit http://creativecommons.org/licenses/by-sa/4.0/
       or send a letter to Creative Commons, PO Box 1866, Mountain
       View, CA 94042, USA.
```

C.4 GD4PoissonRPG

These are the source code listings for the Poisson Disk Sampling implementation of the 2D tile-map RPG scenario in the Godot game engine. The link to the publicly available source code repository is here: https://github.com/Zishan-Rahman/GD4PoissonRPG

C.4.1 .gitattributes

```
1  # Normalize EOL for all files that Git considers text files.
2  * text=auto eol=lf
```

C.4.2 .gitignore

```
# Godot 4+ specific ignores
.godot/
```

C.4.3 project.godot

```
; Engine configuration file.
; It's best edited using the editor UI and not directly,
; since the parameters that go here are not all obvious.
;
; Format:
;   [section] ; section goes between []
;   param=value ; assign values to parameters

config_version=5

[application]

config/name="Poisson Sampling Project"
run/main_scene="res://tile_map.tscn"
config/features=PackedStringArray("4.0", "Forward Plus")
config/icon="res://icon.svg"

[display]

window/size/viewport_height=640

[rendering]

environment/defaults/default_clear_color=Color(0, 0, 0, 1)
```

C.4.4 tile_map.tscn

```
[gd_scene load_steps=7 format=3 uid="uid://f2kv7fettdo7"]

[ext_resource type="Texture2D" uid="uid://c3bpsm4r8t504" path="res://monochrome_packed.png" id="1_uucm3"]
[ext_resource type="Script" path="res://tile_map.gd" id="2_iyhvf"]
[ext_resource type="PackedScene" uid="uid://cau5jgogdnf53" path="res://accept_dialog.tscn" id="3_bk3rg"]
[ext_resource type="PackedScene" uid="uid://b5q8ovcigrvyr" path="res://win_dialog.tscn" id="4_4hdc7"]

[sub_resource type="TileSetAtlasSource" id="TileSetAtlasSource_j4usm"]
texture = ExtResource("1_uucm3")
0:0/0 = 0
1:0/0 = 0
2:0/0 = 0
3:0/0 = 0
4:0/0 = 0
5:0/0 = 0
6:0/0 = 0
7:0/0 = 0
8:0/0 = 0
9:0/0 = 0
10:0/0 = 0
11:0/0 = 0
12:0/0 = 0
13:0/0 = 0
14:0/0 = 0
15:0/0 = 0
16:0/0 = 0
17:0/0 = 0
18:0/0 = 0
19:0/0 = 0
```

```
20:0/0 = 0
21:0/0 = 0
22:0/0 = 0
23:0/0 = 0
24:0/0 = 0
25:0/0 = 0
26:0/0 = 0
27:0/0 = 0
28:0/0 = 0
29:0/0 = 0
30:0/0 = 0
31:0/0 = 0
32:0/0 = 0
33:0/0 = 0
34:0/0 = 0
35:0/0 = 0
36:0/0 = 0
37:0/0 = 0
38:0/0 = 0
39:0/0 = 0
40:0/0 = 0
41:0/0 = 0
42:0/0 = 0
43:0/0 = 0
44:0/0 = 0
45:0/0 = 0
46:0/0 = 0
47:0/0 = 0
48:0/0 = 0
0:1/0 = 0
1:1/0 = 0
2:1/0 = 0
3:1/0 = 0
```

```
4:1/0 = 0
5:1/0 = 0
6:1/0 = 0
7:1/0 = 0
8:1/0 = 0
9:1/0 = 0
10:1/0 = 0
11:1/0 = 0
12:1/0 = 0
13:1/0 = 0
14:1/0 = 0
15:1/0 = 0
16:1/0 = 0
17:1/0 = 0
18:1/0 = 0
19:1/0 = 0
20:1/0 = 0
21:1/0 = 0
22:1/0 = 0
23:1/0 = 0
24:1/0 = 0
25:1/0 = 0
26:1/0 = 0
27:1/0 = 0
28:1/0 = 0
29:1/0 = 0
30:1/0 = 0
31:1/0 = 0
32:1/0 = 0
33:1/0 = 0
34:1/0 = 0
35:1/0 = 0
36:1/0 = 0
```

```
37:1/0 = 0
38:1/0 = 0
39:1/0 = 0
40:1/0 = 0
41:1/0 = 0
42:1/0 = 0
43:1/0 = 0
44:1/0 = 0
45:1/0 = 0
46:1/0 = 0
47:1/0 = 0
48:1/0 = 0
0:2/0 = 0
1:2/0 = 0
2:2/0 = 0
3:2/0 = 0
4:2/0 = 0
5:2/0 = 0
6:2/0 = 0
7:2/0 = 0
8:2/0 = 0
9:2/0 = 0
10:2/0 = 0
11:2/0 = 0
12:2/0 = 0
13:2/0 = 0
14:2/0 = 0
15:2/0 = 0
16:2/0 = 0
17:2/0 = 0
18:2/0 = 0
19:2/0 = 0
20:2/0 = 0
```

```
21:2/0 = 0
22:2/0 = 0
23:2/0 = 0
24:2/0 = 0
25:2/0 = 0
26:2/0 = 0
27:2/0 = 0
28:2/0 = 0
29:2/0 = 0
30:2/0 = 0
31:2/0 = 0
32:2/0 = 0
33:2/0 = 0
34:2/0 = 0
35:2/0 = 0
36:2/0 = 0
37:2/0 = 0
38:2/0 = 0
39:2/0 = 0
40:2/0 = 0
41:2/0 = 0
42:2/0 = 0
43:2/0 = 0
44:2/0 = 0
45:2/0 = 0
46:2/0 = 0
47:2/0 = 0
48:2/0 = 0
0:3/0 = 0
1:3/0 = 0
2:3/0 = 0
3:3/0 = 0
4:3/0 = 0
```

```
5:3/0 = 0
6:3/0 = 0
7:3/0 = 0
8:3/0 = 0
9:3/0 = 0
10:3/0 = 0
11:3/0 = 0
12:3/0 = 0
13:3/0 = 0
14:3/0 = 0
15:3/0 = 0
16:3/0 = 0
17:3/0 = 0
18:3/0 = 0
19:3/0 = 0
20:3/0 = 0
21:3/0 = 0
22:3/0 = 0
23:3/0 = 0
24:3/0 = 0
25:3/0 = 0
26:3/0 = 0
27:3/0 = 0
28:3/0 = 0
29:3/0 = 0
30:3/0 = 0
31:3/0 = 0
32:3/0 = 0
33:3/0 = 0
34:3/0 = 0
35:3/0 = 0
36:3/0 = 0
37:3/0 = 0
```

```
38:3/0 = 0
39:3/0 = 0
40:3/0 = 0
41:3/0 = 0
42:3/0 = 0
43:3/0 = 0
44:3/0 = 0
45:3/0 = 0
46:3/0 = 0
47:3/0 = 0
48:3/0 = 0
0:4/0 = 0
1:4/0 = 0
2:4/0 = 0
3:4/0 = 0
4:4/0 = 0
5:4/0 = 0
6:4/0 = 0
7:4/0 = 0
8:4/0 = 0
9:4/0 = 0
10:4/0 = 0
11:4/0 = 0
12:4/0 = 0
13:4/0 = 0
14:4/0 = 0
15:4/0 = 0
16:4/0 = 0
17:4/0 = 0
18:4/0 = 0
19:4/0 = 0
20:4/0 = 0
21:4/0 = 0
```

```
22:4/0 = 0
23:4/0 = 0
24:4/0 = 0
25:4/0 = 0
26:4/0 = 0
27:4/0 = 0
28:4/0 = 0
29:4/0 = 0
30:4/0 = 0
31:4/0 = 0
32:4/0 = 0
33:4/0 = 0
34:4/0 = 0
35:4/0 = 0
36:4/0 = 0
37:4/0 = 0
38:4/0 = 0
39:4/0 = 0
40:4/0 = 0
41:4/0 = 0
42:4/0 = 0
43:4/0 = 0
44:4/0 = 0
45:4/0 = 0
46:4/0 = 0
47:4/0 = 0
48:4/0 = 0
0:5/0 = 0
1:5/0 = 0
2:5/0 = 0
3:5/0 = 0
4:5/0 = 0
5:5/0 = 0
```

```
6:5/0 = 0
7:5/0 = 0
8:5/0 = 0
9:5/0 = 0
10:5/0 = 0
11:5/0 = 0
12:5/0 = 0
13:5/0 = 0
14:5/0 = 0
15:5/0 = 0
16:5/0 = 0
17:5/0 = 0
18:5/0 = 0
19:5/0 = 0
20:5/0 = 0
21:5/0 = 0
22:5/0 = 0
23:5/0 = 0
24:5/0 = 0
25:5/0 = 0
26:5/0 = 0
27:5/0 = 0
28:5/0 = 0
29:5/0 = 0
30:5/0 = 0
31:5/0 = 0
32:5/0 = 0
33:5/0 = 0
34:5/0 = 0
35:5/0 = 0
36:5/0 = 0
37:5/0 = 0
38:5/0 = 0
```

```
39:5/0 = 0
40:5/0 = 0
41:5/0 = 0
42:5/0 = 0
43:5/0 = 0
44:5/0 = 0
45:5/0 = 0
46:5/0 = 0
47:5/0 = 0
48:5/0 = 0
0:6/0 = 0
1:6/0 = 0
2:6/0 = 0
3:6/0 = 0
4:6/0 = 0
5:6/0 = 0
6:6/0 = 0
7:6/0 = 0
8:6/0 = 0
9:6/0 = 0
10:6/0 = 0
11:6/0 = 0
12:6/0 = 0
13:6/0 = 0
14:6/0 = 0
15:6/0 = 0
16:6/0 = 0
17:6/0 = 0
18:6/0 = 0
19:6/0 = 0
20:6/0 = 0
21:6/0 = 0
22:6/0 = 0
```

```
23:6/0 = 0
24:6/0 = 0
25:6/0 = 0
26:6/0 = 0
27:6/0 = 0
28:6/0 = 0
29:6/0 = 0
30:6/0 = 0
31:6/0 = 0
32:6/0 = 0
33:6/0 = 0
34:6/0 = 0
35:6/0 = 0
36:6/0 = 0
37:6/0 = 0
38:6/0 = 0
39:6/0 = 0
40:6/0 = 0
41:6/0 = 0
42:6/0 = 0
43:6/0 = 0
44:6/0 = 0
45:6/0 = 0
46:6/0 = 0
47:6/0 = 0
48:6/0 = 0
0:7/0 = 0
1:7/0 = 0
2:7/0 = 0
3:7/0 = 0
4:7/0 = 0
5:7/0 = 0
6:7/0 = 0
```

```
7:7/0 = 0
8:7/0 = 0
9:7/0 = 0
10:7/0 = 0
11:7/0 = 0
12:7/0 = 0
13:7/0 = 0
14:7/0 = 0
15:7/0 = 0
16:7/0 = 0
17:7/0 = 0
18:7/0 = 0
19:7/0 = 0
20:7/0 = 0
21:7/0 = 0
22:7/0 = 0
23:7/0 = 0
24:7/0 = 0
25:7/0 = 0
26:7/0 = 0
27:7/0 = 0
28:7/0 = 0
29:7/0 = 0
30:7/0 = 0
31:7/0 = 0
32:7/0 = 0
33:7/0 = 0
34:7/0 = 0
35:7/0 = 0
36:7/0 = 0
37:7/0 = 0
38:7/0 = 0
39:7/0 = 0
```

```
40:7/0 = 0
41:7/0 = 0
42:7/0 = 0
43:7/0 = 0
44:7/0 = 0
45:7/0 = 0
46:7/0 = 0
47:7/0 = 0
48:7/0 = 0
0:8/0 = 0
1:8/0 = 0
2:8/0 = 0
3:8/0 = 0
4:8/0 = 0
5:8/0 = 0
6:8/0 = 0
7:8/0 = 0
8:8/0 = 0
9:8/0 = 0
10:8/0 = 0
11:8/0 = 0
12:8/0 = 0
13:8/0 = 0
14:8/0 = 0
15:8/0 = 0
16:8/0 = 0
17:8/0 = 0
18:8/0 = 0
19:8/0 = 0
20:8/0 = 0
21:8/0 = 0
22:8/0 = 0
23:8/0 = 0
```

```
24:8/0 = 0
25:8/0 = 0
26:8/0 = 0
27:8/0 = 0
28:8/0 = 0
29:8/0 = 0
30:8/0 = 0
31:8/0 = 0
32:8/0 = 0
33:8/0 = 0
34:8/0 = 0
35:8/0 = 0
36:8/0 = 0
37:8/0 = 0
38:8/0 = 0
39:8/0 = 0
40:8/0 = 0
41:8/0 = 0
42:8/0 = 0
43:8/0 = 0
44:8/0 = 0
45:8/0 = 0
46:8/0 = 0
47:8/0 = 0
48:8/0 = 0
0:9/0 = 0
1:9/0 = 0
2:9/0 = 0
3:9/0 = 0
4:9/0 = 0
5:9/0 = 0
6:9/0 = 0
7:9/0 = 0
```

```
8:9/0 = 0
9:9/0 = 0
10:9/0 = 0
11:9/0 = 0
12:9/0 = 0
13:9/0 = 0
14:9/0 = 0
15:9/0 = 0
16:9/0 = 0
17:9/0 = 0
18:9/0 = 0
19:9/0 = 0
20:9/0 = 0
21:9/0 = 0
22:9/0 = 0
23:9/0 = 0
24:9/0 = 0
25:9/0 = 0
26:9/0 = 0
27:9/0 = 0
28:9/0 = 0
29:9/0 = 0
30:9/0 = 0
31:9/0 = 0
32:9/0 = 0
33:9/0 = 0
34:9/0 = 0
35:9/0 = 0
36:9/0 = 0
37:9/0 = 0
38:9/0 = 0
39:9/0 = 0
40:9/0 = 0
```

```
41:9/0 = 0
42:9/0 = 0
43:9/0 = 0
44:9/0 = 0
45:9/0 = 0
46:9/0 = 0
47:9/0 = 0
48:9/0 = 0
0:10/0 = 0
1:10/0 = 0
2:10/0 = 0
3:10/0 = 0
4:10/0 = 0
5:10/0 = 0
6:10/0 = 0
7:10/0 = 0
8:10/0 = 0
9:10/0 = 0
10:10/0 = 0
11:10/0 = 0
12:10/0 = 0
13:10/0 = 0
14:10/0 = 0
15:10/0 = 0
16:10/0 = 0
17:10/0 = 0
18:10/0 = 0
19:10/0 = 0
20:10/0 = 0
21:10/0 = 0
22:10/0 = 0
23:10/0 = 0
24:10/0 = 0
```

```
25:10/0 = 0
26:10/0 = 0
27:10/0 = 0
28:10/0 = 0
29:10/0 = 0
30:10/0 = 0
31:10/0 = 0
32:10/0 = 0
33:10/0 = 0
34:10/0 = 0
35:10/0 = 0
36:10/0 = 0
37:10/0 = 0
38:10/0 = 0
39:10/0 = 0
40:10/0 = 0
41:10/0 = 0
42:10/0 = 0
43:10/0 = 0
44:10/0 = 0
45:10/0 = 0
46:10/0 = 0
47:10/0 = 0
48:10/0 = 0
0:11/0 = 0
1:11/0 = 0
2:11/0 = 0
3:11/0 = 0
4:11/0 = 0
5:11/0 = 0
6:11/0 = 0
7:11/0 = 0
8:11/0 = 0
```

```
9:11/0 = 0
10:11/0 = 0
11:11/0 = 0
12:11/0 = 0
13:11/0 = 0
14:11/0 = 0
15:11/0 = 0
16:11/0 = 0
17:11/0 = 0
18:11/0 = 0
19:11/0 = 0
20:11/0 = 0
21:11/0 = 0
22:11/0 = 0
23:11/0 = 0
24:11/0 = 0
25:11/0 = 0
26:11/0 = 0
27:11/0 = 0
28:11/0 = 0
29:11/0 = 0
30:11/0 = 0
31:11/0 = 0
32:11/0 = 0
33:11/0 = 0
34:11/0 = 0
35:11/0 = 0
36:11/0 = 0
37:11/0 = 0
38:11/0 = 0
39:11/0 = 0
40:11/0 = 0
41:11/0 = 0
```

```
42:11/0 = 0
43:11/0 = 0
44:11/0 = 0
45:11/0 = 0
46:11/0 = 0
47:11/0 = 0
48:11/0 = 0
0:12/0 = 0
1:12/0 = 0
2:12/0 = 0
3:12/0 = 0
4:12/0 = 0
5:12/0 = 0
6:12/0 = 0
7:12/0 = 0
8:12/0 = 0
9:12/0 = 0
10:12/0 = 0
11:12/0 = 0
12:12/0 = 0
13:12/0 = 0
14:12/0 = 0
15:12/0 = 0
16:12/0 = 0
17:12/0 = 0
18:12/0 = 0
19:12/0 = 0
20:12/0 = 0
21:12/0 = 0
22:12/0 = 0
23:12/0 = 0
24:12/0 = 0
25:12/0 = 0
```

```
26:12/0 = 0
27:12/0 = 0
28:12/0 = 0
29:12/0 = 0
30:12/0 = 0
31:12/0 = 0
32:12/0 = 0
33:12/0 = 0
34:12/0 = 0
35:12/0 = 0
36:12/0 = 0
37:12/0 = 0
38:12/0 = 0
39:12/0 = 0
40:12/0 = 0
41:12/0 = 0
42:12/0 = 0
43:12/0 = 0
44:12/0 = 0
45:12/0 = 0
46:12/0 = 0
47:12/0 = 0
48:12/0 = 0
0:13/0 = 0
1:13/0 = 0
2:13/0 = 0
3:13/0 = 0
4:13/0 = 0
5:13/0 = 0
6:13/0 = 0
7:13/0 = 0
8:13/0 = 0
9:13/0 = 0
```

```
10:13/0 = 0
11:13/0 = 0
12:13/0 = 0
13:13/0 = 0
14:13/0 = 0
15:13/0 = 0
16:13/0 = 0
17:13/0 = 0
18:13/0 = 0
19:13/0 = 0
20:13/0 = 0
21:13/0 = 0
22:13/0 = 0
23:13/0 = 0
24:13/0 = 0
25:13/0 = 0
26:13/0 = 0
27:13/0 = 0
28:13/0 = 0
29:13/0 = 0
30:13/0 = 0
31:13/0 = 0
32:13/0 = 0
33:13/0 = 0
34:13/0 = 0
35:13/0 = 0
36:13/0 = 0
37:13/0 = 0
38:13/0 = 0
39:13/0 = 0
40:13/0 = 0
41:13/0 = 0
42:13/0 = 0
```

```
43:13/0 = 0
44:13/0 = 0
45:13/0 = 0
46:13/0 = 0
47:13/0 = 0
48:13/0 = 0
0:14/0 = 0
1:14/0 = 0
2:14/0 = 0
3:14/0 = 0
4:14/0 = 0
5:14/0 = 0
6:14/0 = 0
7:14/0 = 0
8:14/0 = 0
9:14/0 = 0
10:14/0 = 0
11:14/0 = 0
12:14/0 = 0
13:14/0 = 0
14:14/0 = 0
15:14/0 = 0
16:14/0 = 0
17:14/0 = 0
18:14/0 = 0
19:14/0 = 0
20:14/0 = 0
21:14/0 = 0
22:14/0 = 0
23:14/0 = 0
24:14/0 = 0
25:14/0 = 0
26:14/0 = 0
```

```
27:14/0 = 0
28:14/0 = 0
29:14/0 = 0
30:14/0 = 0
31:14/0 = 0
32:14/0 = 0
33:14/0 = 0
34:14/0 = 0
35:14/0 = 0
36:14/0 = 0
37:14/0 = 0
38:14/0 = 0
39:14/0 = 0
40:14/0 = 0
41:14/0 = 0
42:14/0 = 0
43:14/0 = 0
44:14/0 = 0
45:14/0 = 0
46:14/0 = 0
47:14/0 = 0
48:14/0 = 0
0:15/0 = 0
1:15/0 = 0
2:15/0 = 0
3:15/0 = 0
4:15/0 = 0
5:15/0 = 0
6:15/0 = 0
7:15/0 = 0
8:15/0 = 0
9:15/0 = 0
10:15/0 = 0
```

```
11:15/0 = 0
12:15/0 = 0
13:15/0 = 0
14:15/0 = 0
15:15/0 = 0
16:15/0 = 0
17:15/0 = 0
18:15/0 = 0
19:15/0 = 0
20:15/0 = 0
21:15/0 = 0
22:15/0 = 0
23:15/0 = 0
24:15/0 = 0
25:15/0 = 0
26:15/0 = 0
27:15/0 = 0
28:15/0 = 0
29:15/0 = 0
30:15/0 = 0
31:15/0 = 0
32:15/0 = 0
33:15/0 = 0
34:15/0 = 0
35:15/0 = 0
36:15/0 = 0
37:15/0 = 0
38:15/0 = 0
39:15/0 = 0
40:15/0 = 0
41:15/0 = 0
42:15/0 = 0
43:15/0 = 0
```

```
44:15/0 = 0
45:15/0 = 0
46:15/0 = 0
47:15/0 = 0
48:15/0 = 0
0:16/0 = 0
1:16/0 = 0
2:16/0 = 0
3:16/0 = 0
4:16/0 = 0
5:16/0 = 0
6:16/0 = 0
7:16/0 = 0
8:16/0 = 0
9:16/0 = 0
10:16/0 = 0
11:16/0 = 0
12:16/0 = 0
13:16/0 = 0
14:16/0 = 0
15:16/0 = 0
16:16/0 = 0
17:16/0 = 0
18:16/0 = 0
19:16/0 = 0
20:16/0 = 0
21:16/0 = 0
22:16/0 = 0
23:16/0 = 0
24:16/0 = 0
25:16/0 = 0
26:16/0 = 0
27:16/0 = 0
```

```
28:16/0 = 0
29:16/0 = 0
30:16/0 = 0
31:16/0 = 0
32:16/0 = 0
33:16/0 = 0
34:16/0 = 0
35:16/0 = 0
36:16/0 = 0
37:16/0 = 0
38:16/0 = 0
39:16/0 = 0
40:16/0 = 0
41:16/0 = 0
42:16/0 = 0
43:16/0 = 0
44:16/0 = 0
45:16/0 = 0
46:16/0 = 0
47:16/0 = 0
48:16/0 = 0
0:17/0 = 0
1:17/0 = 0
2:17/0 = 0
3:17/0 = 0
4:17/0 = 0
5:17/0 = 0
6:17/0 = 0
7:17/0 = 0
8:17/0 = 0
9:17/0 = 0
10:17/0 = 0
11:17/0 = 0
```

```
12:17/0 = 0
13:17/0 = 0
14:17/0 = 0
15:17/0 = 0
16:17/0 = 0
17:17/0 = 0
18:17/0 = 0
19:17/0 = 0
20:17/0 = 0
21:17/0 = 0
22:17/0 = 0
23:17/0 = 0
24:17/0 = 0
25:17/0 = 0
26:17/0 = 0
27:17/0 = 0
28:17/0 = 0
29:17/0 = 0
30:17/0 = 0
31:17/0 = 0
32:17/0 = 0
33:17/0 = 0
34:17/0 = 0
35:17/0 = 0
36:17/0 = 0
37:17/0 = 0
38:17/0 = 0
39:17/0 = 0
40:17/0 = 0
41:17/0 = 0
42:17/0 = 0
43:17/0 = 0
44:17/0 = 0
```

```
45:17/0 = 0
46:17/0 = 0
47:17/0 = 0
48:17/0 = 0
0:18/0 = 0
1:18/0 = 0
2:18/0 = 0
3:18/0 = 0
4:18/0 = 0
5:18/0 = 0
6:18/0 = 0
7:18/0 = 0
8:18/0 = 0
9:18/0 = 0
10:18/0 = 0
11:18/0 = 0
12:18/0 = 0
13:18/0 = 0
14:18/0 = 0
15:18/0 = 0
16:18/0 = 0
17:18/0 = 0
18:18/0 = 0
19:18/0 = 0
20:18/0 = 0
21:18/0 = 0
22:18/0 = 0
23:18/0 = 0
24:18/0 = 0
25:18/0 = 0
26:18/0 = 0
27:18/0 = 0
28:18/0 = 0
```

```
29:18/0 = 0
30:18/0 = 0
31:18/0 = 0
32:18/0 = 0
33:18/0 = 0
34:18/0 = 0
35:18/0 = 0
36:18/0 = 0
37:18/0 = 0
38:18/0 = 0
39:18/0 = 0
40:18/0 = 0
41:18/0 = 0
42:18/0 = 0
43:18/0 = 0
44:18/0 = 0
45:18/0 = 0
46:18/0 = 0
47:18/0 = 0
48:18/0 = 0
0:19/0 = 0
1:19/0 = 0
2:19/0 = 0
3:19/0 = 0
4:19/0 = 0
5:19/0 = 0
6:19/0 = 0
7:19/0 = 0
8:19/0 = 0
9:19/0 = 0
10:19/0 = 0
11:19/0 = 0
12:19/0 = 0
```

```
13:19/0 = 0
14:19/0 = 0
15:19/0 = 0
16:19/0 = 0
17:19/0 = 0
18:19/0 = 0
19:19/0 = 0
20:19/0 = 0
21:19/0 = 0
22:19/0 = 0
23:19/0 = 0
24:19/0 = 0
25:19/0 = 0
26:19/0 = 0
27:19/0 = 0
28:19/0 = 0
29:19/0 = 0
30:19/0 = 0
31:19/0 = 0
32:19/0 = 0
33:19/0 = 0
34:19/0 = 0
35:19/0 = 0
36:19/0 = 0
37:19/0 = 0
38:19/0 = 0
39:19/0 = 0
40:19/0 = 0
41:19/0 = 0
42:19/0 = 0
43:19/0 = 0
44:19/0 = 0
45:19/0 = 0
```

```
46:19/0 = 0
47:19/0 = 0
48:19/0 = 0
0:20/0 = 0
1:20/0 = 0
2:20/0 = 0
3:20/0 = 0
4:20/0 = 0
5:20/0 = 0
6:20/0 = 0
7:20/0 = 0
8:20/0 = 0
9:20/0 = 0
10:20/0 = 0
11:20/0 = 0
12:20/0 = 0
13:20/0 = 0
14:20/0 = 0
15:20/0 = 0
16:20/0 = 0
17:20/0 = 0
18:20/0 = 0
19:20/0 = 0
20:20/0 = 0
21:20/0 = 0
22:20/0 = 0
23:20/0 = 0
24:20/0 = 0
25:20/0 = 0
26:20/0 = 0
27:20/0 = 0
28:20/0 = 0
29:20/0 = 0
```

```
30:20/0 = 0
31:20/0 = 0
32:20/0 = 0
33:20/0 = 0
34:20/0 = 0
35:20/0 = 0
36:20/0 = 0
37:20/0 = 0
38:20/0 = 0
39:20/0 = 0
40:20/0 = 0
41:20/0 = 0
42:20/0 = 0
43:20/0 = 0
44:20/0 = 0
45:20/0 = 0
46:20/0 = 0
47:20/0 = 0
48:20/0 = 0
0:21/0 = 0
1:21/0 = 0
2:21/0 = 0
3:21/0 = 0
4:21/0 = 0
5:21/0 = 0
6:21/0 = 0
7:21/0 = 0
8:21/0 = 0
9:21/0 = 0
10:21/0 = 0
11:21/0 = 0
12:21/0 = 0
13:21/0 = 0
```

```
14:21/0 = 0
15:21/0 = 0
16:21/0 = 0
17:21/0 = 0
18:21/0 = 0
19:21/0 = 0
20:21/0 = 0
21:21/0 = 0
22:21/0 = 0
23:21/0 = 0
24:21/0 = 0
25:21/0 = 0
26:21/0 = 0
27:21/0 = 0
28:21/0 = 0
29:21/0 = 0
30:21/0 = 0
31:21/0 = 0
32:21/0 = 0
33:21/0 = 0
34:21/0 = 0
35:21/0 = 0
36:21/0 = 0
37:21/0 = 0
38:21/0 = 0
39:21/0 = 0
40:21/0 = 0
41:21/0 = 0
42:21/0 = 0
43:21/0 = 0
44:21/0 = 0
45:21/0 = 0
46:21/0 = 0
```

```
47:21/0 = 0
48:21/0 = 0

[sub_resource type="TileSet" id="TileSet_8pb5m"]
sources/0 = SubResource("TileSetAtlasSource_j4usm")

[node name="TileMap" type="TileMap"]
tile_set = SubResource("TileSet_8pb5m")
format = 2
script = ExtResource("2_iyhvf")

[node name="AcceptDialog" parent="." instance=ExtResource("3_bk3rg"
    )]

[node name="WinDialog" parent="." instance=ExtResource("4_4hdc7")]
```

C.4.5 tile_map.gd

```
extends TileMap

const buildings: Array[Vector2i] = [
    Vector2i(0, 19),
    Vector2i(1, 19),
    Vector2i(2, 19),
    Vector2i(3, 19),
    Vector2i(4, 19),
    Vector2i(5, 19),
    Vector2i(6, 19),
    Vector2i(7, 19),
    Vector2i(8, 20),
    Vector2i(0, 20),
    Vector2i(1, 20),
```

```
15      Vector2i(2, 20),
16      Vector2i(3, 20),
17      Vector2i(4, 20),
18      Vector2i(5, 20),
19      Vector2i(6, 20),
20      Vector2i(7, 20),
21      Vector2i(8, 20),
22      Vector2i(0, 21),
23      Vector2i(1, 21),
24      Vector2i(2, 21),
25      Vector2i(3, 21),
26      Vector2i(4, 21),
27      Vector2i(5, 21),
28      Vector2i(6, 21),
29      Vector2i(7, 21),
30      Vector2i(8, 21)
31  ]
32  const trees: Array[Vector2i] = [
33      Vector2i(0,1),
34      Vector2i(1,1),
35      Vector2i(2,1),
36      Vector2i(3,1),
37      Vector2i(4,1),
38      Vector2i(5,1),
39      Vector2i(6,1),
40      Vector2i(7,1),
41      Vector2i(0,2),
42      Vector2i(1,2),
43      Vector2i(2,2),
44      Vector2i(3,2),
45      Vector2i(4,2)
46  ]
47  const PLAYER_SPRITE: Vector2i = Vector2i(24, 7)
```

```
48  var player_placement_cell: Vector2i
49  const rings: Array[Vector2i] = [
50      Vector2i(43, 6),
51      Vector2i(44, 6),
52      Vector2i(45, 6),
53      Vector2i(46, 6)
54  ]
55  var ring_placement_cell: Vector2i
56
57  var x_tile_range: int = ProjectSettings.get_setting("display/window
        /size/viewport_width") / tile_set.tile_size.x
58  var y_tile_range: int = ProjectSettings.get_setting("display/window
        /size/viewport_height") / tile_set.tile_size.y
59
60  var cell_points: Array[Vector2]
61  ## The probability that a building gets painted at a cell in lieu
        of a tree. The higher this probability, the more likely a
        building tile gets painted instead of a tree tile.
62  @export_range(0.0, 1.0) var paint_building_probability: float =
        0.125
63  ## The radius value used to measure distances between points for
        the algorithm. The longer the radius, the further apart points
        are during the algorithm's processing, and the further apart
        painted cells are in the game.
64  @export_range(0.5, 2.5) var point_radius: float = 1.0
65  ## The size of the region in which the algorithm is performed. Set
        to the "default" tile map size (72, 40) in the script, shown as
         (0, 0) in the Godot editor. Can be changed to use a smaller
        region for the algorithm itself, of course resulting in less
        cells covered within the boundaries set for this game, though
        the algorithm will perform faster due to less cells being
        checked.
66  @export var region_size: Vector2 = Vector2(x_tile_range,
```

```
        y_tile_range)
67  ## The maximum number of times a cell is checked before it is
        ignored. A cell can be accepted and painted on before this
        maximum number is reached. The higher this value, the more
        times a cell is checked, therefore the higher the algorithm's
        processing time.
68  @export_range(1, 50, 1) var rejection_samples: int = 8
69
70  # Called when the node enters the scene tree for the first time.
71  func _ready():
72      randomize()
73      var start_time: float = Time.get_ticks_msec()
74      cell_points = generate_points(point_radius, region_size,
            rejection_samples)
75      paint_points()
76      place_player()
77      place_ring()
78      var new_time: float = Time.get_ticks_msec() - start_time
79      print("Time taken: " + str(new_time) + "ms")
80      $AcceptDialog.dialog_text = "You're a hollow Golem who seeks the
            ultimate treasure; a ring that's got something on top of it
            . It's somewhere in this large village and barely visible to
            your naked eyes, which took us " + str(new_time) + "
            milliseconds to generate (" + str(new_time / 1000.0) + "
            seconds), but you'll stop at nothing to get what you want.
            You can chow down every tree and fauna that stands in your
            way of the ring, but your Achilles heel is any bricks and
            mortar, which WILL make you stop at your tracks. Since it's
            easy to get lost in here, we'll tell you that you're in
            position " + str(player_placement_cell) + " in this big
            village of size " + str(Vector2i(x_tile_range, y_tile_range)
            ) + ". However, it is YOUR job to find the ring, so are you
            ready to attain the treasure that is rightfully yours?!"
```

```
81      $AcceptDialog.visible = true
82      $AcceptDialog.confirmed.connect(_on_AcceptDialog_closed)
83      $AcceptDialog.canceled.connect(_on_AcceptDialog_closed)
84      $WinDialog.confirmed.connect(_on_WinDialog_confirmed)
85      $WinDialog.canceled.connect(_on_WinDialog_canceled)
86      get_tree().paused = true
87
88  func _on_WinDialog_confirmed() -> void:
89      get_tree().reload_current_scene()
90
91  func _on_WinDialog_canceled() -> void:
92      get_tree().quit()
93
94  func _on_AcceptDialog_closed() -> void:
95      $AcceptDialog.visible = false
96      get_tree().paused = false
97
98  func paint_points() -> void:
99      for point in cell_points:
100         var cell_point: Vector2i = Vector2i(roundi(point.x), roundi(
              point.y))
101         if randf() < paint_building_probability:
102             set_cell(0, cell_point, 0, buildings.pick_random())
103         else:
104             set_cell(0, cell_point, 0, trees.pick_random())
105
106 func _get_random_placement_cell() -> Vector2i:
107     return Vector2i(randi() % x_tile_range, randi() % y_tile_range)
108
109 func place_player() -> void:
110     player_placement_cell = _get_random_placement_cell()
111     while get_used_cells(0).has(player_placement_cell):
112         player_placement_cell = _get_random_placement_cell()
```

```
113     set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
114
115 func place_ring() -> void:
116     ring_placement_cell = _get_random_placement_cell()
117     while get_used_cells(0).has(ring_placement_cell):
118         ring_placement_cell = _get_random_placement_cell()
119     set_cell(0, ring_placement_cell, 0, rings.pick_random())
120
121 func _is_not_out_of_bounds(cell: Vector2i) -> bool:
122     return cell.x >= 0 and cell.x < x_tile_range and cell.y >= 0 and
          cell.y < y_tile_range
123
124 func _physics_process(_delta) -> void:
125     var previous_cell: Vector2i = player_placement_cell
126     var direction: Vector2i = Vector2i.ZERO
127     if Input.is_action_pressed("ui_up"): direction = Vector2i.UP
128     elif Input.is_action_pressed("ui_down"): direction = Vector2i.
          DOWN
129     elif Input.is_action_pressed("ui_left"): direction = Vector2i.
          LEFT
130     elif Input.is_action_pressed("ui_right"): direction = Vector2i.
          RIGHT
131     var new_placement_cell: Vector2i = player_placement_cell +
          direction
132     if (not get_used_cells(0).has(new_placement_cell) or trees.has(
          get_cell_atlas_coords(0, new_placement_cell)) or
          new_placement_cell == ring_placement_cell) and
          _is_not_out_of_bounds(new_placement_cell):
133         player_placement_cell = new_placement_cell
134         set_cell(0, previous_cell, 0) # deletes contents of previous
              cell (atlas_coords = Vector2i(-1, -1))
135         set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
136         if player_placement_cell == ring_placement_cell:
```

```
137             $WinDialog.visible = true
138             get_tree().paused = true
139 
140 # ALGORITHM BEGINS HERE
141 
142 func generate_points(radius: float, sample_region_size: Vector2,
        number_of_samples_before_rejection: int = 30) -> Array[Vector2
        ]:
143     var cell_size: float = radius / sqrt(2)
144     var grid: Array[Array] = []
145     var points: Array[Vector2] = []
146     var spawn_points: Array[Vector2] = []
147     var grid_x_axis_size: int = ceili(sample_region_size.x/cell_size
            )
148     var grid_y_axis_size: int = ceili(sample_region_size.y/cell_size
            )
149 
150     for i in range(grid_x_axis_size):
151         grid.append([])
152         for j in range(grid_y_axis_size):
153             grid[i].append(0)
154 
155     spawn_points.append(sample_region_size/2)
156 
157     while spawn_points.size() > 0:
158         var spawn_index: int = randi_range(0, spawn_points.size() -
                1)
159         var spawn_centre: Vector2 = spawn_points[spawn_index]
160         var candidate_accepted: bool = false
161 
162         for i in range(number_of_samples_before_rejection):
163             var angle: float = randf() * TAU # TAU = PI * 2
164             var direction: Vector2 = Vector2(sin(angle), cos(angle))
```

```
165             var candidate: Vector2 = spawn_centre + direction *
                    randf_range(radius, 2 * radius)
166             if is_valid(candidate, sample_region_size, cell_size,
                    radius, points, grid, grid_x_axis_size,
                    grid_y_axis_size):
167                 points.append(candidate)
168                 spawn_points.append(candidate)
169                 grid[int(candidate.x/cell_size)][int(candidate.y/
                        cell_size)] = len(points)
170                 candidate_accepted = true
171                 break
172
173         if not candidate_accepted:
174             spawn_points.remove_at(spawn_index)
175
176     return points
177
178 func is_valid(candidate: Vector2, sample_region_size: Vector2,
        cell_size: float, radius: float, points: Array[Vector2], grid:
        Array[Array], grid_x_axis_size: int, grid_y_axis_size: int) ->
        bool:
179     if candidate.x >= 0 and candidate.x < sample_region_size.x and
            candidate.y >= 0 and candidate.y < sample_region_size.y:
180         var cell_x: int = roundi(candidate.x / cell_size)
181         var cell_y: int = roundi(candidate.y / cell_size)
182         var search_start_x: int = max(0, cell_x - 2)
183         var search_end_x: int = min(cell_x + 2, grid_x_axis_size - 1)
184         var search_start_y: int = max(0, cell_y - 2)
185         var search_end_y: int = min(cell_y + 2, grid_y_axis_size - 1)
186         for x in range(search_start_x, search_end_x):
187             for y in range(search_start_y, search_end_y):
188                 var point_index: int = grid[x][y] - 1
189                 if point_index != -1:
```

```
190                 var distance: float = (candidate - points[
                        point_index]).length_squared()
191                 if distance < radius:
192                     return false
193         return true
194     return false
```

C.4.6 accept_dialog.tscn

```
1   [gd_scene format=3 uid="uid://cau5jgogdnf53"]
2
3   [node name="AcceptDialog" type="AcceptDialog"]
4   title = "Tree-Munching Time!"
5   position = Vector2i(326, 100)
6   size = Vector2i(500, 421)
7   mouse_passthrough = true
8   ok_button_text = "Bring it on!"
9   dialog_text = "You're a hollow Golem who seeks the ultimate
        treasure; a ring that's got something on top of it. It's
        somewhere in this large village and barely visible to your
        naked eyes, but you'll stop at nothing to get what you want.
        You can chow down every tree and fauna that stands in your way
        of the ring, but your Achilles heel is any bricks and mortar,
        which will make you stop at your tracks. Are you ready to
        attain your treasure?w Golem in a black-and-white world, in
        search for your most desired treasure. It's a ring with
        something on top of it. And you'll stop at nothing to get what
        you want. You can chow down every tree and fauna that stands in
         your way of the ring, but your Achilles heel is any bricks and
         mortar, which will make you stop at your tracks. Are you ready
         to attain the treasure that is rightfully yours?!"
10  dialog_autowrap = true
```

C.4.7 win_dialog.tscn

```
[gd_scene format=3 uid="uid://b5q8ovcigrvyr"]

[node name="WinDialog" type="ConfirmationDialog"]
title = "You Found the Treasure!"
position = Vector2i(326, 100)
size = Vector2i(500, 421)
mouse_passthrough = true
ok_button_text = "Get Me a New Village"
dialog_text = "You found your treasure! Well done, you!

Would you like to travel to a new village in the hopes of finding another ring? Or would you like to take your treasure home now?"
dialog_autowrap = true
cancel_button_text = "Get Me Out of Here"
```

C.4.8 icon.svg.import

```
[remap]

importer="texture"
type="CompressedTexture2D"
uid="uid://uotfe6soknht"
path="res://.godot/imported/icon.svg-218a8f2b3041327d8a5756f3a245f83b.ctex"
metadata={
"vram_texture": false
}

[deps]
```

```
12
13  source_file="res://icon.svg"
14  dest_files=["res://.godot/imported/icon.svg-218
        a8f2b3041327d8a5756f3a245f83b.ctex"]
15
16  [params]
17
18  compress/mode=0
19  compress/high_quality=false
20  compress/lossy_quality=0.7
21  compress/hdr_compression=1
22  compress/normal_map=0
23  compress/channel_pack=0
24  mipmaps/generate=false
25  mipmaps/limit=-1
26  roughness/mode=0
27  roughness/src_normal=""
28  process/fix_alpha_border=true
29  process/premult_alpha=false
30  process/normal_map_invert_y=false
31  process/hdr_as_srgb=false
32  process/hdr_clamp_exposure=false
33  process/size_limit=0
34  detect_3d/compress_to=1
35  svg/scale=1.0
36  editor/scale_with_editor_scale=false
37  editor/convert_colors_with_editor_theme=false
```

C.4.9 monochrome_packed.png.import

```
1   [remap]
2
```

```
importer="texture"
type="CompressedTexture2D"
uid="uid://c3bpsm4r8t504"
path="res://.godot/imported/monochrome_packed.png-6
    b9bd1c64dd50f72acd3afd14d1ac34f.ctex"
metadata={
"vram_texture": false
}

[deps]

source_file="res://monochrome_packed.png"
dest_files=["res://.godot/imported/monochrome_packed.png-6
    b9bd1c64dd50f72acd3afd14d1ac34f.ctex"]

[params]

compress/mode=0
compress/high_quality=false
compress/lossy_quality=0.7
compress/hdr_compression=1
compress/normal_map=0
compress/channel_pack=0
mipmaps/generate=false
mipmaps/limit=-1
roughness/mode=0
roughness/src_normal=""
process/fix_alpha_border=true
process/premult_alpha=false
process/normal_map_invert_y=false
process/hdr_as_srgb=false
process/hdr_clamp_exposure=false
process/size_limit=0
```

```
34  detect_3d/compress_to=1
```

C.4.10 LICENSE

```
1  This work is licensed under the Creative Commons Attribution-
       ShareAlike 4.0 International License. To view a copy of this
       license, visit http://creativecommons.org/licenses/by-sa/4.0/
       or send a letter to Creative Commons, PO Box 1866, Mountain
       View, CA 94042, USA.
```

C.5 GD4NoiseRPG

These are the source code listings for the Perlin/Simplex Noise implementation of the 2D tile-map RPG scenario in the Godot game engine. The link to the publicly available source code repository is here: https://github.com/Zishan-Rahman/GD4NoiseRPG

C.5.1 .gitattributes

```
1  # Normalize EOL for all files that Git considers text files.
2  * text=auto eol=lf
```

C.5.2 .gitignore

```
1  # Godot 4+ specific ignores
2  .godot/
```

C.5.3 project.godot

```
1  ; Engine configuration file.
```

```
; It's best edited using the editor UI and not directly,
; since the parameters that go here are not all obvious.
;
; Format:
;   [section] ; section goes between []
;   param=value ; assign values to parameters

config_version=5

[application]

config/name="Noise Demo"
run/main_scene="res://tile_map.tscn"
config/features=PackedStringArray("4.0", "Forward Plus")
config/icon="res://icon.svg"

[display]

window/size/viewport_height=640

[rendering]

environment/defaults/default_clear_color=Color(0, 0, 0, 1)
```

C.5.4 tile_map.tscn

```
[gd_scene load_steps=7 format=3 uid="uid://d4jdcavluwx6s"]

[ext_resource type="Texture2D" uid="uid://m662wwd4prmn" path="res
    ://monochrome_packed.png" id="1_ld7xx"]
[ext_resource type="Script" path="res://tile_map.gd" id="2_o1dn1"]
[ext_resource type="PackedScene" uid="uid://cau5jgogdnf53" path="
```

```
    res://accept_dialog.tscn" id="3_e0ur6"]
[ext_resource type="PackedScene" uid="uid://b5q8ovcigrvyr" path="
    res://win_dialog.tscn" id="4_ecfaa"]

[sub_resource type="TileSetAtlasSource" id="
    TileSetAtlasSource_1e80b"]
texture = ExtResource("1_ld7xx")
0:0/0 = 0
1:0/0 = 0
2:0/0 = 0
3:0/0 = 0
4:0/0 = 0
5:0/0 = 0
6:0/0 = 0
7:0/0 = 0
8:0/0 = 0
9:0/0 = 0
10:0/0 = 0
11:0/0 = 0
12:0/0 = 0
13:0/0 = 0
14:0/0 = 0
15:0/0 = 0
16:0/0 = 0
17:0/0 = 0
18:0/0 = 0
19:0/0 = 0
20:0/0 = 0
21:0/0 = 0
22:0/0 = 0
23:0/0 = 0
24:0/0 = 0
25:0/0 = 0
```

```
26:0/0 = 0
27:0/0 = 0
28:0/0 = 0
29:0/0 = 0
30:0/0 = 0
31:0/0 = 0
32:0/0 = 0
33:0/0 = 0
34:0/0 = 0
35:0/0 = 0
36:0/0 = 0
37:0/0 = 0
38:0/0 = 0
39:0/0 = 0
40:0/0 = 0
41:0/0 = 0
42:0/0 = 0
43:0/0 = 0
44:0/0 = 0
45:0/0 = 0
46:0/0 = 0
47:0/0 = 0
48:0/0 = 0
0:1/0 = 0
1:1/0 = 0
2:1/0 = 0
3:1/0 = 0
4:1/0 = 0
5:1/0 = 0
6:1/0 = 0
7:1/0 = 0
8:1/0 = 0
9:1/0 = 0
```

```
10:1/0 = 0
11:1/0 = 0
12:1/0 = 0
13:1/0 = 0
14:1/0 = 0
15:1/0 = 0
16:1/0 = 0
17:1/0 = 0
18:1/0 = 0
19:1/0 = 0
20:1/0 = 0
21:1/0 = 0
22:1/0 = 0
23:1/0 = 0
24:1/0 = 0
25:1/0 = 0
26:1/0 = 0
27:1/0 = 0
28:1/0 = 0
29:1/0 = 0
30:1/0 = 0
31:1/0 = 0
32:1/0 = 0
33:1/0 = 0
34:1/0 = 0
35:1/0 = 0
36:1/0 = 0
37:1/0 = 0
38:1/0 = 0
39:1/0 = 0
40:1/0 = 0
41:1/0 = 0
42:1/0 = 0
```

```
43:1/0 = 0
44:1/0 = 0
45:1/0 = 0
46:1/0 = 0
47:1/0 = 0
48:1/0 = 0
0:2/0 = 0
1:2/0 = 0
2:2/0 = 0
3:2/0 = 0
4:2/0 = 0
5:2/0 = 0
6:2/0 = 0
7:2/0 = 0
8:2/0 = 0
9:2/0 = 0
10:2/0 = 0
11:2/0 = 0
12:2/0 = 0
13:2/0 = 0
14:2/0 = 0
15:2/0 = 0
16:2/0 = 0
17:2/0 = 0
18:2/0 = 0
19:2/0 = 0
20:2/0 = 0
21:2/0 = 0
22:2/0 = 0
23:2/0 = 0
24:2/0 = 0
25:2/0 = 0
26:2/0 = 0
```

```
27:2/0 = 0
28:2/0 = 0
29:2/0 = 0
30:2/0 = 0
31:2/0 = 0
32:2/0 = 0
33:2/0 = 0
34:2/0 = 0
35:2/0 = 0
36:2/0 = 0
37:2/0 = 0
38:2/0 = 0
39:2/0 = 0
40:2/0 = 0
41:2/0 = 0
42:2/0 = 0
43:2/0 = 0
44:2/0 = 0
45:2/0 = 0
46:2/0 = 0
47:2/0 = 0
48:2/0 = 0
0:3/0 = 0
1:3/0 = 0
2:3/0 = 0
3:3/0 = 0
4:3/0 = 0
5:3/0 = 0
6:3/0 = 0
7:3/0 = 0
8:3/0 = 0
9:3/0 = 0
10:3/0 = 0
```

```
11:3/0 = 0
12:3/0 = 0
13:3/0 = 0
14:3/0 = 0
15:3/0 = 0
16:3/0 = 0
17:3/0 = 0
18:3/0 = 0
19:3/0 = 0
20:3/0 = 0
21:3/0 = 0
22:3/0 = 0
23:3/0 = 0
24:3/0 = 0
25:3/0 = 0
26:3/0 = 0
27:3/0 = 0
28:3/0 = 0
29:3/0 = 0
30:3/0 = 0
31:3/0 = 0
32:3/0 = 0
33:3/0 = 0
34:3/0 = 0
35:3/0 = 0
36:3/0 = 0
37:3/0 = 0
38:3/0 = 0
39:3/0 = 0
40:3/0 = 0
41:3/0 = 0
42:3/0 = 0
43:3/0 = 0
```

```
44:3/0 = 0
45:3/0 = 0
46:3/0 = 0
47:3/0 = 0
48:3/0 = 0
0:4/0 = 0
1:4/0 = 0
2:4/0 = 0
3:4/0 = 0
4:4/0 = 0
5:4/0 = 0
6:4/0 = 0
7:4/0 = 0
8:4/0 = 0
9:4/0 = 0
10:4/0 = 0
11:4/0 = 0
12:4/0 = 0
13:4/0 = 0
14:4/0 = 0
15:4/0 = 0
16:4/0 = 0
17:4/0 = 0
18:4/0 = 0
19:4/0 = 0
20:4/0 = 0
21:4/0 = 0
22:4/0 = 0
23:4/0 = 0
24:4/0 = 0
25:4/0 = 0
26:4/0 = 0
27:4/0 = 0
```

```
28:4/0 = 0
29:4/0 = 0
30:4/0 = 0
31:4/0 = 0
32:4/0 = 0
33:4/0 = 0
34:4/0 = 0
35:4/0 = 0
36:4/0 = 0
37:4/0 = 0
38:4/0 = 0
39:4/0 = 0
40:4/0 = 0
41:4/0 = 0
42:4/0 = 0
43:4/0 = 0
44:4/0 = 0
45:4/0 = 0
46:4/0 = 0
47:4/0 = 0
48:4/0 = 0
0:5/0 = 0
1:5/0 = 0
2:5/0 = 0
3:5/0 = 0
4:5/0 = 0
5:5/0 = 0
6:5/0 = 0
7:5/0 = 0
8:5/0 = 0
9:5/0 = 0
10:5/0 = 0
11:5/0 = 0
```

```
12:5/0 = 0
13:5/0 = 0
14:5/0 = 0
15:5/0 = 0
16:5/0 = 0
17:5/0 = 0
18:5/0 = 0
19:5/0 = 0
20:5/0 = 0
21:5/0 = 0
22:5/0 = 0
23:5/0 = 0
24:5/0 = 0
25:5/0 = 0
26:5/0 = 0
27:5/0 = 0
28:5/0 = 0
29:5/0 = 0
30:5/0 = 0
31:5/0 = 0
32:5/0 = 0
33:5/0 = 0
34:5/0 = 0
35:5/0 = 0
36:5/0 = 0
37:5/0 = 0
38:5/0 = 0
39:5/0 = 0
40:5/0 = 0
41:5/0 = 0
42:5/0 = 0
43:5/0 = 0
44:5/0 = 0
```

```
45:5/0 = 0
46:5/0 = 0
47:5/0 = 0
48:5/0 = 0
0:6/0 = 0
1:6/0 = 0
2:6/0 = 0
3:6/0 = 0
4:6/0 = 0
5:6/0 = 0
6:6/0 = 0
7:6/0 = 0
8:6/0 = 0
9:6/0 = 0
10:6/0 = 0
11:6/0 = 0
12:6/0 = 0
13:6/0 = 0
14:6/0 = 0
15:6/0 = 0
16:6/0 = 0
17:6/0 = 0
18:6/0 = 0
19:6/0 = 0
20:6/0 = 0
21:6/0 = 0
22:6/0 = 0
23:6/0 = 0
24:6/0 = 0
25:6/0 = 0
26:6/0 = 0
27:6/0 = 0
28:6/0 = 0
```

```
29:6/0 = 0
30:6/0 = 0
31:6/0 = 0
32:6/0 = 0
33:6/0 = 0
34:6/0 = 0
35:6/0 = 0
36:6/0 = 0
37:6/0 = 0
38:6/0 = 0
39:6/0 = 0
40:6/0 = 0
41:6/0 = 0
42:6/0 = 0
43:6/0 = 0
44:6/0 = 0
45:6/0 = 0
46:6/0 = 0
47:6/0 = 0
48:6/0 = 0
0:7/0 = 0
1:7/0 = 0
2:7/0 = 0
3:7/0 = 0
4:7/0 = 0
5:7/0 = 0
6:7/0 = 0
7:7/0 = 0
8:7/0 = 0
9:7/0 = 0
10:7/0 = 0
11:7/0 = 0
12:7/0 = 0
```

```
13:7/0 = 0
14:7/0 = 0
15:7/0 = 0
16:7/0 = 0
17:7/0 = 0
18:7/0 = 0
19:7/0 = 0
20:7/0 = 0
21:7/0 = 0
22:7/0 = 0
23:7/0 = 0
24:7/0 = 0
25:7/0 = 0
26:7/0 = 0
27:7/0 = 0
28:7/0 = 0
29:7/0 = 0
30:7/0 = 0
31:7/0 = 0
32:7/0 = 0
33:7/0 = 0
34:7/0 = 0
35:7/0 = 0
36:7/0 = 0
37:7/0 = 0
38:7/0 = 0
39:7/0 = 0
40:7/0 = 0
41:7/0 = 0
42:7/0 = 0
43:7/0 = 0
44:7/0 = 0
45:7/0 = 0
```

```
46:7/0 = 0
47:7/0 = 0
48:7/0 = 0
0:8/0 = 0
1:8/0 = 0
2:8/0 = 0
3:8/0 = 0
4:8/0 = 0
5:8/0 = 0
6:8/0 = 0
7:8/0 = 0
8:8/0 = 0
9:8/0 = 0
10:8/0 = 0
11:8/0 = 0
12:8/0 = 0
13:8/0 = 0
14:8/0 = 0
15:8/0 = 0
16:8/0 = 0
17:8/0 = 0
18:8/0 = 0
19:8/0 = 0
20:8/0 = 0
21:8/0 = 0
22:8/0 = 0
23:8/0 = 0
24:8/0 = 0
25:8/0 = 0
26:8/0 = 0
27:8/0 = 0
28:8/0 = 0
29:8/0 = 0
```

```
30:8/0 = 0
31:8/0 = 0
32:8/0 = 0
33:8/0 = 0
34:8/0 = 0
35:8/0 = 0
36:8/0 = 0
37:8/0 = 0
38:8/0 = 0
39:8/0 = 0
40:8/0 = 0
41:8/0 = 0
42:8/0 = 0
43:8/0 = 0
44:8/0 = 0
45:8/0 = 0
46:8/0 = 0
47:8/0 = 0
48:8/0 = 0
0:9/0 = 0
1:9/0 = 0
2:9/0 = 0
3:9/0 = 0
4:9/0 = 0
5:9/0 = 0
6:9/0 = 0
7:9/0 = 0
8:9/0 = 0
9:9/0 = 0
10:9/0 = 0
11:9/0 = 0
12:9/0 = 0
13:9/0 = 0
```

```
14:9/0 = 0
15:9/0 = 0
16:9/0 = 0
17:9/0 = 0
18:9/0 = 0
19:9/0 = 0
20:9/0 = 0
21:9/0 = 0
22:9/0 = 0
23:9/0 = 0
24:9/0 = 0
25:9/0 = 0
26:9/0 = 0
27:9/0 = 0
28:9/0 = 0
29:9/0 = 0
30:9/0 = 0
31:9/0 = 0
32:9/0 = 0
33:9/0 = 0
34:9/0 = 0
35:9/0 = 0
36:9/0 = 0
37:9/0 = 0
38:9/0 = 0
39:9/0 = 0
40:9/0 = 0
41:9/0 = 0
42:9/0 = 0
43:9/0 = 0
44:9/0 = 0
45:9/0 = 0
46:9/0 = 0
```

```
47:9/0 = 0
48:9/0 = 0
0:10/0 = 0
1:10/0 = 0
2:10/0 = 0
3:10/0 = 0
4:10/0 = 0
5:10/0 = 0
6:10/0 = 0
7:10/0 = 0
8:10/0 = 0
9:10/0 = 0
10:10/0 = 0
11:10/0 = 0
12:10/0 = 0
13:10/0 = 0
14:10/0 = 0
15:10/0 = 0
16:10/0 = 0
17:10/0 = 0
18:10/0 = 0
19:10/0 = 0
20:10/0 = 0
21:10/0 = 0
22:10/0 = 0
23:10/0 = 0
24:10/0 = 0
25:10/0 = 0
26:10/0 = 0
27:10/0 = 0
28:10/0 = 0
29:10/0 = 0
30:10/0 = 0
```

```
31:10/0 = 0
32:10/0 = 0
33:10/0 = 0
34:10/0 = 0
35:10/0 = 0
36:10/0 = 0
37:10/0 = 0
38:10/0 = 0
39:10/0 = 0
40:10/0 = 0
41:10/0 = 0
42:10/0 = 0
43:10/0 = 0
44:10/0 = 0
45:10/0 = 0
46:10/0 = 0
47:10/0 = 0
48:10/0 = 0
0:11/0 = 0
1:11/0 = 0
2:11/0 = 0
3:11/0 = 0
4:11/0 = 0
5:11/0 = 0
6:11/0 = 0
7:11/0 = 0
8:11/0 = 0
9:11/0 = 0
10:11/0 = 0
11:11/0 = 0
12:11/0 = 0
13:11/0 = 0
14:11/0 = 0
```

```
15:11/0 = 0
16:11/0 = 0
17:11/0 = 0
18:11/0 = 0
19:11/0 = 0
20:11/0 = 0
21:11/0 = 0
22:11/0 = 0
23:11/0 = 0
24:11/0 = 0
25:11/0 = 0
26:11/0 = 0
27:11/0 = 0
28:11/0 = 0
29:11/0 = 0
30:11/0 = 0
31:11/0 = 0
32:11/0 = 0
33:11/0 = 0
34:11/0 = 0
35:11/0 = 0
36:11/0 = 0
37:11/0 = 0
38:11/0 = 0
39:11/0 = 0
40:11/0 = 0
41:11/0 = 0
42:11/0 = 0
43:11/0 = 0
44:11/0 = 0
45:11/0 = 0
46:11/0 = 0
47:11/0 = 0
```

```
48:11/0 = 0
0:12/0 = 0
1:12/0 = 0
2:12/0 = 0
3:12/0 = 0
4:12/0 = 0
5:12/0 = 0
6:12/0 = 0
7:12/0 = 0
8:12/0 = 0
9:12/0 = 0
10:12/0 = 0
11:12/0 = 0
12:12/0 = 0
13:12/0 = 0
14:12/0 = 0
15:12/0 = 0
16:12/0 = 0
17:12/0 = 0
18:12/0 = 0
19:12/0 = 0
20:12/0 = 0
21:12/0 = 0
22:12/0 = 0
23:12/0 = 0
24:12/0 = 0
25:12/0 = 0
26:12/0 = 0
27:12/0 = 0
28:12/0 = 0
29:12/0 = 0
30:12/0 = 0
31:12/0 = 0
```

```
32:12/0 = 0
33:12/0 = 0
34:12/0 = 0
35:12/0 = 0
36:12/0 = 0
37:12/0 = 0
38:12/0 = 0
39:12/0 = 0
40:12/0 = 0
41:12/0 = 0
42:12/0 = 0
43:12/0 = 0
44:12/0 = 0
45:12/0 = 0
46:12/0 = 0
47:12/0 = 0
48:12/0 = 0
0:13/0 = 0
1:13/0 = 0
2:13/0 = 0
3:13/0 = 0
4:13/0 = 0
5:13/0 = 0
6:13/0 = 0
7:13/0 = 0
8:13/0 = 0
9:13/0 = 0
10:13/0 = 0
11:13/0 = 0
12:13/0 = 0
13:13/0 = 0
14:13/0 = 0
15:13/0 = 0
```

```
16:13/0 = 0
17:13/0 = 0
18:13/0 = 0
19:13/0 = 0
20:13/0 = 0
21:13/0 = 0
22:13/0 = 0
23:13/0 = 0
24:13/0 = 0
25:13/0 = 0
26:13/0 = 0
27:13/0 = 0
28:13/0 = 0
29:13/0 = 0
30:13/0 = 0
31:13/0 = 0
32:13/0 = 0
33:13/0 = 0
34:13/0 = 0
35:13/0 = 0
36:13/0 = 0
37:13/0 = 0
38:13/0 = 0
39:13/0 = 0
40:13/0 = 0
41:13/0 = 0
42:13/0 = 0
43:13/0 = 0
44:13/0 = 0
45:13/0 = 0
46:13/0 = 0
47:13/0 = 0
48:13/0 = 0
```

```
0:14/0 = 0
1:14/0 = 0
2:14/0 = 0
3:14/0 = 0
4:14/0 = 0
5:14/0 = 0
6:14/0 = 0
7:14/0 = 0
8:14/0 = 0
9:14/0 = 0
10:14/0 = 0
11:14/0 = 0
12:14/0 = 0
13:14/0 = 0
14:14/0 = 0
15:14/0 = 0
16:14/0 = 0
17:14/0 = 0
18:14/0 = 0
19:14/0 = 0
20:14/0 = 0
21:14/0 = 0
22:14/0 = 0
23:14/0 = 0
24:14/0 = 0
25:14/0 = 0
26:14/0 = 0
27:14/0 = 0
28:14/0 = 0
29:14/0 = 0
30:14/0 = 0
31:14/0 = 0
32:14/0 = 0
```

```
33:14/0 = 0
34:14/0 = 0
35:14/0 = 0
36:14/0 = 0
37:14/0 = 0
38:14/0 = 0
39:14/0 = 0
40:14/0 = 0
41:14/0 = 0
42:14/0 = 0
43:14/0 = 0
44:14/0 = 0
45:14/0 = 0
46:14/0 = 0
47:14/0 = 0
48:14/0 = 0
0:15/0 = 0
1:15/0 = 0
2:15/0 = 0
3:15/0 = 0
4:15/0 = 0
5:15/0 = 0
6:15/0 = 0
7:15/0 = 0
8:15/0 = 0
9:15/0 = 0
10:15/0 = 0
11:15/0 = 0
12:15/0 = 0
13:15/0 = 0
14:15/0 = 0
15:15/0 = 0
16:15/0 = 0
```

```
17:15/0 = 0
18:15/0 = 0
19:15/0 = 0
20:15/0 = 0
21:15/0 = 0
22:15/0 = 0
23:15/0 = 0
24:15/0 = 0
25:15/0 = 0
26:15/0 = 0
27:15/0 = 0
28:15/0 = 0
29:15/0 = 0
30:15/0 = 0
31:15/0 = 0
32:15/0 = 0
33:15/0 = 0
34:15/0 = 0
35:15/0 = 0
36:15/0 = 0
37:15/0 = 0
38:15/0 = 0
39:15/0 = 0
40:15/0 = 0
41:15/0 = 0
42:15/0 = 0
43:15/0 = 0
44:15/0 = 0
45:15/0 = 0
46:15/0 = 0
47:15/0 = 0
48:15/0 = 0
0:16/0 = 0
```

```
1:16/0 = 0
2:16/0 = 0
3:16/0 = 0
4:16/0 = 0
5:16/0 = 0
6:16/0 = 0
7:16/0 = 0
8:16/0 = 0
9:16/0 = 0
10:16/0 = 0
11:16/0 = 0
12:16/0 = 0
13:16/0 = 0
14:16/0 = 0
15:16/0 = 0
16:16/0 = 0
17:16/0 = 0
18:16/0 = 0
19:16/0 = 0
20:16/0 = 0
21:16/0 = 0
22:16/0 = 0
23:16/0 = 0
24:16/0 = 0
25:16/0 = 0
26:16/0 = 0
27:16/0 = 0
28:16/0 = 0
29:16/0 = 0
30:16/0 = 0
31:16/0 = 0
32:16/0 = 0
33:16/0 = 0
```

```
34:16/0 = 0
35:16/0 = 0
36:16/0 = 0
37:16/0 = 0
38:16/0 = 0
39:16/0 = 0
40:16/0 = 0
41:16/0 = 0
42:16/0 = 0
43:16/0 = 0
44:16/0 = 0
45:16/0 = 0
46:16/0 = 0
47:16/0 = 0
48:16/0 = 0
0:17/0 = 0
1:17/0 = 0
2:17/0 = 0
3:17/0 = 0
4:17/0 = 0
5:17/0 = 0
6:17/0 = 0
7:17/0 = 0
8:17/0 = 0
9:17/0 = 0
10:17/0 = 0
11:17/0 = 0
12:17/0 = 0
13:17/0 = 0
14:17/0 = 0
15:17/0 = 0
16:17/0 = 0
17:17/0 = 0
```

```
18:17/0 = 0
19:17/0 = 0
20:17/0 = 0
21:17/0 = 0
22:17/0 = 0
23:17/0 = 0
24:17/0 = 0
25:17/0 = 0
26:17/0 = 0
27:17/0 = 0
28:17/0 = 0
29:17/0 = 0
30:17/0 = 0
31:17/0 = 0
32:17/0 = 0
33:17/0 = 0
34:17/0 = 0
35:17/0 = 0
36:17/0 = 0
37:17/0 = 0
38:17/0 = 0
39:17/0 = 0
40:17/0 = 0
41:17/0 = 0
42:17/0 = 0
43:17/0 = 0
44:17/0 = 0
45:17/0 = 0
46:17/0 = 0
47:17/0 = 0
48:17/0 = 0
0:18/0 = 0
1:18/0 = 0
```

```
2:18/0 = 0
3:18/0 = 0
4:18/0 = 0
5:18/0 = 0
6:18/0 = 0
7:18/0 = 0
8:18/0 = 0
9:18/0 = 0
10:18/0 = 0
11:18/0 = 0
12:18/0 = 0
13:18/0 = 0
14:18/0 = 0
15:18/0 = 0
16:18/0 = 0
17:18/0 = 0
18:18/0 = 0
19:18/0 = 0
20:18/0 = 0
21:18/0 = 0
22:18/0 = 0
23:18/0 = 0
24:18/0 = 0
25:18/0 = 0
26:18/0 = 0
27:18/0 = 0
28:18/0 = 0
29:18/0 = 0
30:18/0 = 0
31:18/0 = 0
32:18/0 = 0
33:18/0 = 0
34:18/0 = 0
```

```
35:18/0 = 0
36:18/0 = 0
37:18/0 = 0
38:18/0 = 0
39:18/0 = 0
40:18/0 = 0
41:18/0 = 0
42:18/0 = 0
43:18/0 = 0
44:18/0 = 0
45:18/0 = 0
46:18/0 = 0
47:18/0 = 0
48:18/0 = 0
0:19/0 = 0
1:19/0 = 0
2:19/0 = 0
3:19/0 = 0
4:19/0 = 0
5:19/0 = 0
6:19/0 = 0
7:19/0 = 0
8:19/0 = 0
9:19/0 = 0
10:19/0 = 0
11:19/0 = 0
12:19/0 = 0
13:19/0 = 0
14:19/0 = 0
15:19/0 = 0
16:19/0 = 0
17:19/0 = 0
18:19/0 = 0
```

```
19:19/0 = 0
20:19/0 = 0
21:19/0 = 0
22:19/0 = 0
23:19/0 = 0
24:19/0 = 0
25:19/0 = 0
26:19/0 = 0
27:19/0 = 0
28:19/0 = 0
29:19/0 = 0
30:19/0 = 0
31:19/0 = 0
32:19/0 = 0
33:19/0 = 0
34:19/0 = 0
35:19/0 = 0
36:19/0 = 0
37:19/0 = 0
38:19/0 = 0
39:19/0 = 0
40:19/0 = 0
41:19/0 = 0
42:19/0 = 0
43:19/0 = 0
44:19/0 = 0
45:19/0 = 0
46:19/0 = 0
47:19/0 = 0
48:19/0 = 0
0:20/0 = 0
1:20/0 = 0
2:20/0 = 0
```

```
3:20/0 = 0
4:20/0 = 0
5:20/0 = 0
6:20/0 = 0
7:20/0 = 0
8:20/0 = 0
9:20/0 = 0
10:20/0 = 0
11:20/0 = 0
12:20/0 = 0
13:20/0 = 0
14:20/0 = 0
15:20/0 = 0
16:20/0 = 0
17:20/0 = 0
18:20/0 = 0
19:20/0 = 0
20:20/0 = 0
21:20/0 = 0
22:20/0 = 0
23:20/0 = 0
24:20/0 = 0
25:20/0 = 0
26:20/0 = 0
27:20/0 = 0
28:20/0 = 0
29:20/0 = 0
30:20/0 = 0
31:20/0 = 0
32:20/0 = 0
33:20/0 = 0
34:20/0 = 0
35:20/0 = 0
```

```
36:20/0 = 0
37:20/0 = 0
38:20/0 = 0
39:20/0 = 0
40:20/0 = 0
41:20/0 = 0
42:20/0 = 0
43:20/0 = 0
44:20/0 = 0
45:20/0 = 0
46:20/0 = 0
47:20/0 = 0
48:20/0 = 0
0:21/0 = 0
1:21/0 = 0
2:21/0 = 0
3:21/0 = 0
4:21/0 = 0
5:21/0 = 0
6:21/0 = 0
7:21/0 = 0
8:21/0 = 0
9:21/0 = 0
10:21/0 = 0
11:21/0 = 0
12:21/0 = 0
13:21/0 = 0
14:21/0 = 0
15:21/0 = 0
16:21/0 = 0
17:21/0 = 0
18:21/0 = 0
19:21/0 = 0
```

```
20:21/0 = 0
21:21/0 = 0
22:21/0 = 0
23:21/0 = 0
24:21/0 = 0
25:21/0 = 0
26:21/0 = 0
27:21/0 = 0
28:21/0 = 0
29:21/0 = 0
30:21/0 = 0
31:21/0 = 0
32:21/0 = 0
33:21/0 = 0
34:21/0 = 0
35:21/0 = 0
36:21/0 = 0
37:21/0 = 0
38:21/0 = 0
39:21/0 = 0
40:21/0 = 0
41:21/0 = 0
42:21/0 = 0
43:21/0 = 0
44:21/0 = 0
45:21/0 = 0
46:21/0 = 0
47:21/0 = 0
48:21/0 = 0

[sub_resource type="TileSet" id="TileSet_qtrb6"]
sources/0 = SubResource("TileSetAtlasSource_1e80b")

```

```
[node name="TileMap" type="TileMap"]
texture_filter = 1
tile_set = SubResource("TileSet_qtrb6")
format = 2
script = ExtResource("2_o1dn1")

[node name="AcceptDialog" parent="." instance=ExtResource("3_e0ur6"
    )]

[node name="WinDialog" parent="." instance=ExtResource("4_ecfaa")]
title = "You Found the Treasure!"
```

C.5.5 tile_map.gd

```
extends TileMap

const buildings: Array[Vector2i] = [
    Vector2i(0, 19),
    Vector2i(1, 19),
    Vector2i(2, 19),
    Vector2i(3, 19),
    Vector2i(4, 19),
    Vector2i(5, 19),
    Vector2i(6, 19),
    Vector2i(7, 19),
    Vector2i(8, 20),
    Vector2i(0, 20),
    Vector2i(1, 20),
    Vector2i(2, 20),
    Vector2i(3, 20),
    Vector2i(4, 20),
    Vector2i(5, 20),
```

```
19      Vector2i(6, 20),
20      Vector2i(7, 20),
21      Vector2i(8, 20),
22      Vector2i(0, 21),
23      Vector2i(1, 21),
24      Vector2i(2, 21),
25      Vector2i(3, 21),
26      Vector2i(4, 21),
27      Vector2i(5, 21),
28      Vector2i(6, 21),
29      Vector2i(7, 21),
30      Vector2i(8, 21)
31  ]
32  const trees: Array[Vector2i] = [
33      Vector2i(0,1),
34      Vector2i(1,1),
35      Vector2i(2,1),
36      Vector2i(3,1),
37      Vector2i(4,1),
38      Vector2i(5,1),
39      Vector2i(6,1),
40      Vector2i(7,1),
41      Vector2i(0,2),
42      Vector2i(1,2),
43      Vector2i(2,2),
44      Vector2i(3,2),
45      Vector2i(4,2)
46  ]
47  const PLAYER_SPRITE: Vector2i = Vector2i(24, 7)
48  var player_placement_cell: Vector2i
49  const rings: Array[Vector2i] = [
50      Vector2i(43, 6),
51      Vector2i(44, 6),
```

```
52      Vector2i(45, 6),
53      Vector2i(46, 6)
54  ]
55  var ring_placement_cell: Vector2i
56
57  var noise: FastNoiseLite
58  ## Defines the type of noise generation algorithm to use. Equates
        to the noise_type property in FastNoiseLite.
59  @export_enum("Perlin", "Simplex", "Simplex Smooth", "Value", "Value
        Cubic") var noise_type: String = "Simplex Smooth"
60  ## Defines the type of method used to combine octaves of a noise
        image into a fractal. Directly equates to the FractalType enum
        in FastNoiseLite.
61  @export var fractal_type: FastNoiseLite.FractalType
62  ## Defines the function used to calculate the distance between the
        nearest/second-nearest point(s). Directly equates to the
        CellularDistanceFunction enum in FastNoiseLite.
63  @export var cellular_distance_type: FastNoiseLite.
        CellularDistanceFunction
64  ### Defines the number of noise octaves to use in the generated
        image.
65  #@export_range(1, 10, 1) var octaves: int = 5
66  ## Defines the frequency of the generated noise, the higher the
        frequency, the rougher and more granular the noise.
67  @export_range(0.0, 1.0) var noise_frequency: float = 0.894
68
69  ## Defines the upper limit to set for painting a tree tile on a
        specific noise pixel. If the value returned by the get_noise_2d
        method (in FastNoiseLite) is smaller than this, then it gets
        painted.
70  @export_range(-1.0, 1.0) var tree_cap: float = -0.048
71  ## Defines the upper limit to set for painting a building tile on a
        specific noise pixel. If the value returned by the
```

```
        get_noise_2d method (in FastNoiseLite) is smaller than this,
        then it gets painted. If the value of building_cap is smaller
        than tree_cap, then decide whether or not to paint a building
        cell there with building_overtakes_tree.
72  @export_range(-1.0, 1.0) var building_cap: float = -0.252
73  ## Only used when building_cap is smaller than tree_cap. Determines
         the probability that a building tile would be painted in a
        cell where a tree tile was, or could be, also painted. Whether
        or not the cell actually is painted over is decided on
        computation time.
74  @export_range(0.0, 0.5) var building_overtakes_tree: float = 0.12
75  var x_tile_range: int = ProjectSettings.get_setting("display/window
        /size/viewport_width") / tile_set.tile_size.x
76  var y_tile_range: int = ProjectSettings.get_setting("display/window
        /size/viewport_height") / tile_set.tile_size.y
77
78  # Called when the node enters the scene tree for the first time.
79  func _ready() -> void:
80      randomize()
81      var start_time: float = Time.get_ticks_msec()
82      set_noise()
83      paint_tiles()
84      place_player()
85      place_ring()
86      var new_time: float = Time.get_ticks_msec() - start_time
87      print("Time taken: " + str(new_time) + "ms")
88      $AcceptDialog.dialog_text = "You're a hollow Golem who seeks the
             ultimate treasure; a ring that's got something on top of it
            . It's somewhere in this large village and barely visible to
             your naked eyes, which took us " + str(new_time) + "
            milliseconds to generate (" + str(new_time / 1000.0) + "
            seconds), but you'll stop at nothing to get what you want.
            You can chow down every tree and fauna that stands in your
```

```
            way of the ring, but your Achilles heel is any bricks and
            mortar, which WILL make you stop at your tracks. Since it's
            easy to get lost in here, we'll tell you that you're in
            position " + str(player_placement_cell) + " in this big
            village of size " + str(Vector2i(x_tile_range, y_tile_range)
            ) + ". However, it is YOUR job to find the ring, so are you
            ready to attain the treasure that is rightfully yours?!"
89      $AcceptDialog.visible = true
90      $AcceptDialog.confirmed.connect(_on_AcceptDialog_closed)
91      $AcceptDialog.canceled.connect(_on_AcceptDialog_closed)
92      $WinDialog.confirmed.connect(_on_WinDialog_confirmed)
93      $WinDialog.canceled.connect(_on_WinDialog_canceled)
94      get_tree().paused = true
95
96  func _on_WinDialog_confirmed() -> void:
97      get_tree().reload_current_scene()
98
99  func _on_WinDialog_canceled() -> void:
100     get_tree().quit()
101
102 func _on_AcceptDialog_closed() -> void:
103     $AcceptDialog.visible = false
104     get_tree().paused = false
105
106 func _get_random_placement_cell() -> Vector2i:
107     return Vector2i(randi() % x_tile_range, randi() % y_tile_range)
108
109 func place_player() -> void:
110     player_placement_cell = _get_random_placement_cell()
111     while get_used_cells(0).has(player_placement_cell):
112         player_placement_cell = _get_random_placement_cell()
113     set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
114
```

```
115 func place_ring() -> void:
116     ring_placement_cell = _get_random_placement_cell()
117     while get_used_cells(0).has(ring_placement_cell):
118         ring_placement_cell = _get_random_placement_cell()
119     set_cell(0, ring_placement_cell, 0, rings.pick_random())
120
121 func _is_not_out_of_bounds(cell: Vector2i) -> bool:
122     return cell.x >= 0 and cell.x < x_tile_range and cell.y >= 0 and
          cell.y < y_tile_range
123
124 func _physics_process(_delta: float) -> void:
125     var previous_cell: Vector2i = player_placement_cell
126     var direction: Vector2i = Vector2i.ZERO
127     if Input.is_action_pressed("ui_up"): direction = Vector2i.UP
128     elif Input.is_action_pressed("ui_down"): direction = Vector2i.
          DOWN
129     elif Input.is_action_pressed("ui_left"): direction = Vector2i.
          LEFT
130     elif Input.is_action_pressed("ui_right"): direction = Vector2i.
          RIGHT
131     var new_placement_cell: Vector2i = player_placement_cell +
          direction
132     if (not get_used_cells(0).has(new_placement_cell) or trees.has(
          get_cell_atlas_coords(0, new_placement_cell)) or
          new_placement_cell == ring_placement_cell) and
          _is_not_out_of_bounds(new_placement_cell):
133         player_placement_cell = new_placement_cell
134         set_cell(0, previous_cell, 0) # deletes contents of previous
              cell (atlas_coords = Vector2i(-1, -1))
135         set_cell(0, player_placement_cell, 0, PLAYER_SPRITE)
136         if player_placement_cell == ring_placement_cell:
137             $WinDialog.visible = true
138             get_tree().paused = true
```

```
139
140 # ALGORITHM BEGINS HERE
141
142 func _get_noise_type() -> int:
143     match noise_type:
144         "Perlin": return 3
145         "Simplex": return 0
146         "Value": return 5
147         "Value Cubic": return 4
148         _: return 1 # Return Simplex Smooth by default
149
150 func set_noise() -> void:
151     noise = FastNoiseLite.new()
152     noise.frequency = noise_frequency
153     noise.noise_type = _get_noise_type() as FastNoiseLite.NoiseType
154     noise.fractal_type = fractal_type
155     noise.cellular_distance_function = cellular_distance_type
156 #   noise.fractal_octaves = octaves
157     noise.seed = randi()
158
159 # I took inspiration from a Godot 3.1 tutorial: https://youtu.be/
        SBDs8hbs43w
160 # However, no code was taken or adapted in any way, shape or form.
161
162 func paint_tiles() -> void:
163     for x in range(x_tile_range):
164         for y in range(y_tile_range):
165             var noise_point: float = noise.get_noise_2d(x * tile_set.
                    tile_size.x, y * tile_set.tile_size.y)
166             if noise_point < tree_cap and not get_used_cells(0).has(
                    Vector2i(x, y)):
167                 set_cell(0, Vector2i(x, y), 0, trees.pick_random())
168             if ((building_cap <= tree_cap and randf() <
```

```
                building_overtakes_tree) or (building_cap > tree_cap
                and noise_point < building_cap)) and not
                get_used_cells(0).has(Vector2i(x, y)):
169             set_cell(0, Vector2i(x, y), 0, buildings.pick_random())
```

C.5.6 accept_dialog.tscn

```
1   [gd_scene format=3 uid="uid://cau5jgogdnf53"]
2
3   [node name="AcceptDialog" type="AcceptDialog"]
4   title = "Tree-Munching Time!"
5   position = Vector2i(326, 100)
6   size = Vector2i(500, 421)
7   mouse_passthrough = true
8   ok_button_text = "Bring it on!"
9   dialog_text = "You're a hollow Golem who seeks the ultimate
       treasure; a ring that's got something on top of it. It's
       somewhere in this large village and barely visible to your
       naked eyes, but you'll stop at nothing to get what you want.
       You can chow down every tree and fauna that stands in your way
       of the ring, but your Achilles heel is any bricks and mortar,
       which will make you stop at your tracks. Are you ready to
       attain your treasure?w Golem in a black-and-white world, in
       search for your most desired treasure. It's a ring with
       something on top of it. And you'll stop at nothing to get what
       you want. You can chow down every tree and fauna that stands in
        your way of the ring, but your Achilles heel is any bricks and
        mortar, which will make you stop at your tracks. Are you ready
        to attain the treasure that is rightfully yours?!"
10  dialog_autowrap = true
```

C.5.7 win_dialog.tscn

```
[gd_scene format=3 uid="uid://b5q8ovcigrvyr"]

[node name="WinDialog" type="ConfirmationDialog"]
title = "Tree-Munching Time!"
position = Vector2i(326, 100)
size = Vector2i(500, 421)
mouse_passthrough = true
ok_button_text = "Get Me a New Village"
dialog_text = "You found your treasure! Well done, you!

Would you like to travel to a new village in the hopes of finding another ring? Or would you like to take your treasure home now?"
dialog_autowrap = true
cancel_button_text = "Get Me Out of Here"
```

C.5.8 icon.svg.import

```
[remap]

importer="texture"
type="CompressedTexture2D"
uid="uid://crgf6ascxsdt0"
path="res://.godot/imported/icon.svg-218a8f2b3041327d8a5756f3a245f83b.ctex"
metadata={
"vram_texture": false
}

[deps]
```

```
12
13  source_file="res://icon.svg"
14  dest_files=["res://.godot/imported/icon.svg-218
        a8f2b3041327d8a5756f3a245f83b.ctex"]
15
16  [params]
17
18  compress/mode=0
19  compress/high_quality=false
20  compress/lossy_quality=0.7
21  compress/hdr_compression=1
22  compress/normal_map=0
23  compress/channel_pack=0
24  mipmaps/generate=false
25  mipmaps/limit=-1
26  roughness/mode=0
27  roughness/src_normal=""
28  process/fix_alpha_border=true
29  process/premult_alpha=false
30  process/normal_map_invert_y=false
31  process/hdr_as_srgb=false
32  process/hdr_clamp_exposure=false
33  process/size_limit=0
34  detect_3d/compress_to=1
35  svg/scale=1.0
36  editor/scale_with_editor_scale=false
37  editor/convert_colors_with_editor_theme=false
```

C.5.9 monochrome_packed.png.import

```
1   [remap]
2
```

```
importer="texture"
type="CompressedTexture2D"
uid="uid://m662wwd4prmn"
path="res://.godot/imported/monochrome_packed.png-6
    b9bd1c64dd50f72acd3afd14d1ac34f.ctex"
metadata={
"vram_texture": false
}

[deps]

source_file="res://monochrome_packed.png"
dest_files=["res://.godot/imported/monochrome_packed.png-6
    b9bd1c64dd50f72acd3afd14d1ac34f.ctex"]

[params]

compress/mode=0
compress/high_quality=false
compress/lossy_quality=0.7
compress/hdr_compression=1
compress/normal_map=0
compress/channel_pack=0
mipmaps/generate=false
mipmaps/limit=-1
roughness/mode=0
roughness/src_normal=""
process/fix_alpha_border=true
process/premult_alpha=false
process/normal_map_invert_y=false
process/hdr_as_srgb=false
process/hdr_clamp_exposure=false
process/size_limit=0
```

```
34  detect_3d/compress_to=1
```

C.5.10 LICENSE

```
1  This work is licensed under the Creative Commons Attribution-
       ShareAlike 4.0 International License. To view a copy of this
       license, visit http://creativecommons.org/licenses/by-sa/4.0/
       or send a letter to Creative Commons, PO Box 1866, Mountain
       View, CA 94042, USA.
```

C.6 LSystemGrammarDemo

These are the source code listings for the L-System demonstration used to generate the screenshots in Figures 3.1, 3.2, 3.3 and 3.4 (in the report body). The link to the publicly available source code repository is here: https://github.com/Zishan-Rahman/LSystemGrammarDemo

C.6.1 .gitattributes

```
1  # Normalize EOL for all files that Git considers text files.
2  * text=auto eol=lf
```

C.6.2 .gitignore

```
1  # Godot 4+ specific ignores
2  .godot/
```

C.6.3 project.godot

```
1  ; Engine configuration file.
```

```
; It's best edited using the editor UI and not directly,
; since the parameters that go here are not all obvious.
;
; Format:
;   [section] ; section goes between []
;   param=value ; assign values to parameters

config_version=5

[application]

config/name="LSystemGrammarDemo"
run/main_scene="res://DemoNode.tscn"
config/features=PackedStringArray("4.0")

[display]

window/stretch/mode="canvas_items"
window/stretch/aspect="expand"

[gui]

common/drop_mouse_on_gui_input_disabled=true

[physics]

common/enable_pause_aware_picking=true
```

C.6.4 DemoNode.tscn

```
[gd_scene load_steps=2 format=3 uid="uid://bu380we4od0ln"]

```

```
3   [ext_resource type="Script" path="res://DemoNode.gd" id="1"]
4
5   [node name="DemoNode" type="Node"]
6   script = ExtResource("1")
7   choices = "deterministic"
8
9   [node name="Timer" type="Timer" parent="."]
10
11  [node name="TextLabel" type="Label" parent="."]
12  offset_right = 1152.0
13  offset_bottom = 23.0
14  autowrap_mode = 3
15
16  [connection signal="timeout" from="Timer" to="." method="
        _on_Timer_timeout"]
```

C.6.5 DemoNode.gd

```
1   extends Node
2
3   # Basic: https://youtu.be/feNVBEPXAcE?t=77 (L = +)
4   # Choices: http://paulbourke.net/fractals/lsys/
5   # Deterministic: https://www1.biologie.uni-hamburg.de/b-online/
        e28_3/lsys.html#D0L-system
6
7   ## Allows you to decide which ruleset to use. See the script file
        for the sources of said rulesets.
8   @export_enum("basic", "choices", "deterministic") var choices:
        String = "choices"
9   var axiom: String
10  @onready var string: String
11  @onready var timer = $Timer
```

```
@onready var label = $TextLabel
@onready var rules: Array[Dictionary]

func set_values() -> void:
    match choices:
        "basic":
            rules = [
                {
                    "from": "F",
                    "to": "F+F"
                }
            ]
            axiom = "F+"
        "choices":
            rules = [
                {
                    "from": "F",
                    "to": "F+--FFFF+F+-FF"
                }
            ]
            axiom = "F+F+F+F"
        "deterministic":
            rules = [
                {
                    "from": "a",
                    "to": "ab"
                },
                {
                    "from": "b",
                    "to": "a"
                }
            ]
            axiom = "b"
```

```
45
46  func _ready() -> void:
47      set_values()
48      string = axiom
49      label.size.x = get_viewport().size.x
50      label.text = string
51      print(len(string))
52      timer.start()
53
54  # Thanks to Alexander Gillberg (Codat) for inspiration
55  # https://youtu.be/eY9XkJERiG0
56  # Code adapted with his permission
57  func get_new_replacement(character: String) -> String:
58      for rule in rules:
59          if rule["from"] == character:
60              return rule["to"]
61      return ""
62
63  func _on_Timer_timeout() -> void:
64      # Thanks to Alexander Gillberg (Codat) for inspiration
65      # https://youtu.be/eY9XkJERiG0
66      # Code adapted with his permission
67      var new_string: String = ""
68      for character in string:
69          new_string += get_new_replacement(character)
70      string = new_string
71      label.text = string
72      print(len(string))
```

C.6.6 icon.svg.import

```
1  [remap]
```

```
2
3   importer="texture"
4   type="CompressedTexture2D"
5   uid="uid://cwnnuqmejj04q"
6   path="res://.godot/imported/icon.svg-218
        a8f2b3041327d8a5756f3a245f83b.ctex"
7   metadata={
8   "vram_texture": false
9   }
10
11  [deps]
12
13  source_file="res://icon.svg"
14  dest_files=["res://.godot/imported/icon.svg-218
        a8f2b3041327d8a5756f3a245f83b.ctex"]
15
16  [params]
17
18  compress/mode=0
19  compress/high_quality=false
20  compress/lossy_quality=0.7
21  compress/hdr_compression=1
22  compress/normal_map=0
23  compress/channel_pack=0
24  mipmaps/generate=false
25  mipmaps/limit=-1
26  roughness/mode=0
27  roughness/src_normal=""
28  process/fix_alpha_border=true
29  process/premult_alpha=false
30  process/normal_map_invert_y=false
31  process/hdr_as_srgb=false
32  process/hdr_clamp_exposure=false
```

```
process/size_limit=0
detect_3d/compress_to=1
svg/scale=1.0
editor/scale_with_editor_scale=false
editor/convert_colors_with_editor_theme=false
```

C.6.7 LICENSE

```
This work is licensed under the Creative Commons Attribution-
    ShareAlike 4.0 International License. To view a copy of this
    license, visit http://creativecommons.org/licenses/by-sa/4.0/
    or send a letter to Creative Commons, PO Box 1866, Mountain
    View, CA 94042, USA.
```

C.7 README

```
I verify that I am the sole author of the programs contained in
    this folder, except where explicitly stated to the contrary.

Zishan Rahman
21st April 2023
```