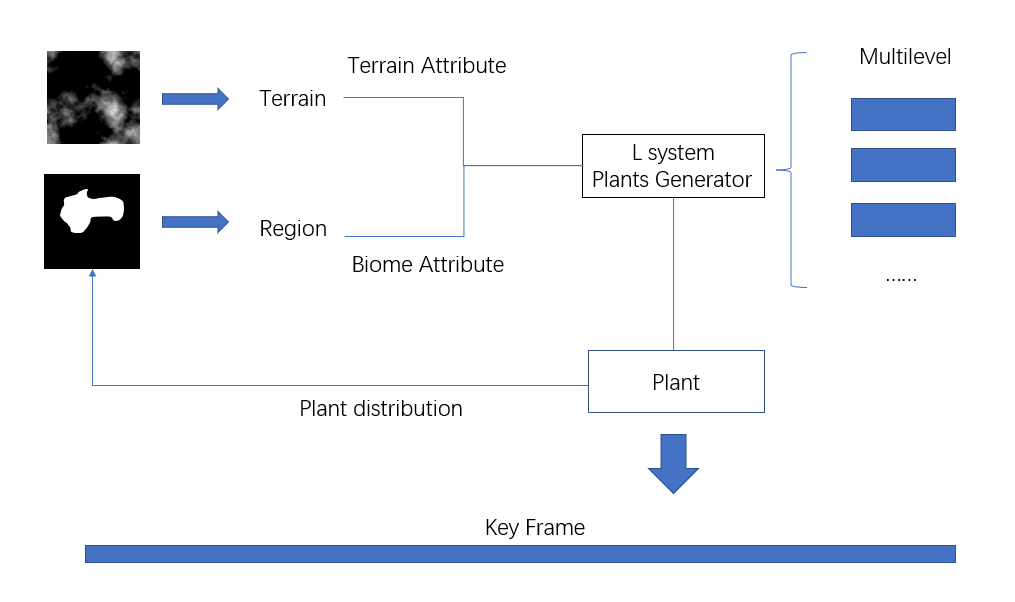
**Project report**

**Summary**

****This report will give a detailed introduction to the algorithm design and code structure of the Maya program ---- Landscape System, as well as the application instructions and operation results.

**Introduction**

Landscape System is a system that allows users to simulate the growth and distribution of plants created by L-system on the selected region of the terrain they set up. The functions of Landscape System include terrain editing, plants creation, area selection and plants filling.

**Algorithm design**

Algorithm design can be divided into four connected parts ---- Terrain generation, Region selection, Plant generation, Plant distribution, which are similar to the functional structure of Landscape System.

In the part of terrain generation, we adopt two methods: noise map and height map. The algorithm of terrain generation from noise graph is based on a random noise generating function, and the values generated by the function are assigned to the vertices of the terrain grid. The algorithm of generating terrain from height map is to gray-scale the imported height map and assign the color value of each pixel point to the grid vertices of terrain.

Region selection is based on the imported bitmap with the same size as the terrain to generate a list of the corresponding pixel length, which uses 0 and 1 to record whether each vertex of the terrain is chosen to plant trees or not.

The generation of plants We use the L-system to generate 3d plant branch models with given rules and the number of iterations. We used three rules to generate three different plant branches, and at the same time drawn two 2d leaves distributed on the branches.

Finally, the distribution of plants on the terrain is randomly distributed in the selected area. At the same time, the orientation of plants will be adjusted randomly, and the number of plants can be set by the user.

**Script structures**

The whole flow of control is sequential. And we use Class in our programming. The main GUI system and the L-system class. So we can instantiate the L-system easily. Other functions about terrain won’t be heavily used. They are more fit for single function other than class function.

The program will enter from main.py, then call other module in GUI’s drawback functions.

* main.py
  + Class LandscapeSystem

We create GUI there, and attach the functions for the drawback

These structure are sequential.

* + Lsystem.py
    - Class Lsystem

The generation of lsystem rule and the plant is a loop-structure. And the selection-structure is widely used in the different rule.

* + terrain.py

The process on the vertices of the mesh and the plants is embedded loop-structure.

* + TerrainGenerator.py

To generate the terrain from image, we should process the image by a embedded loop-structure to deal with the two dimension.

**User manual**

**Preparation**

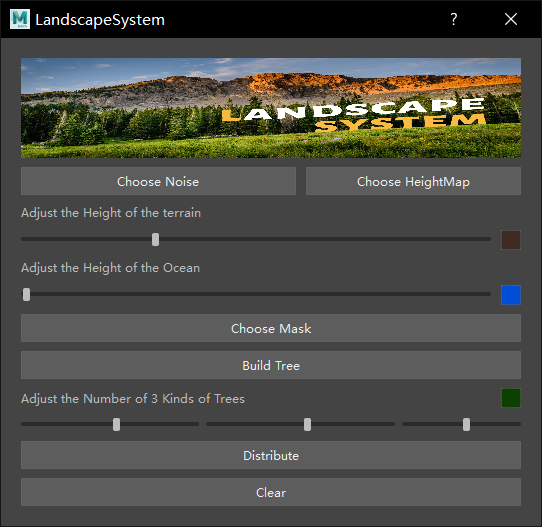
The following external modules are used in the program, which need to be installed in advance:

pymel

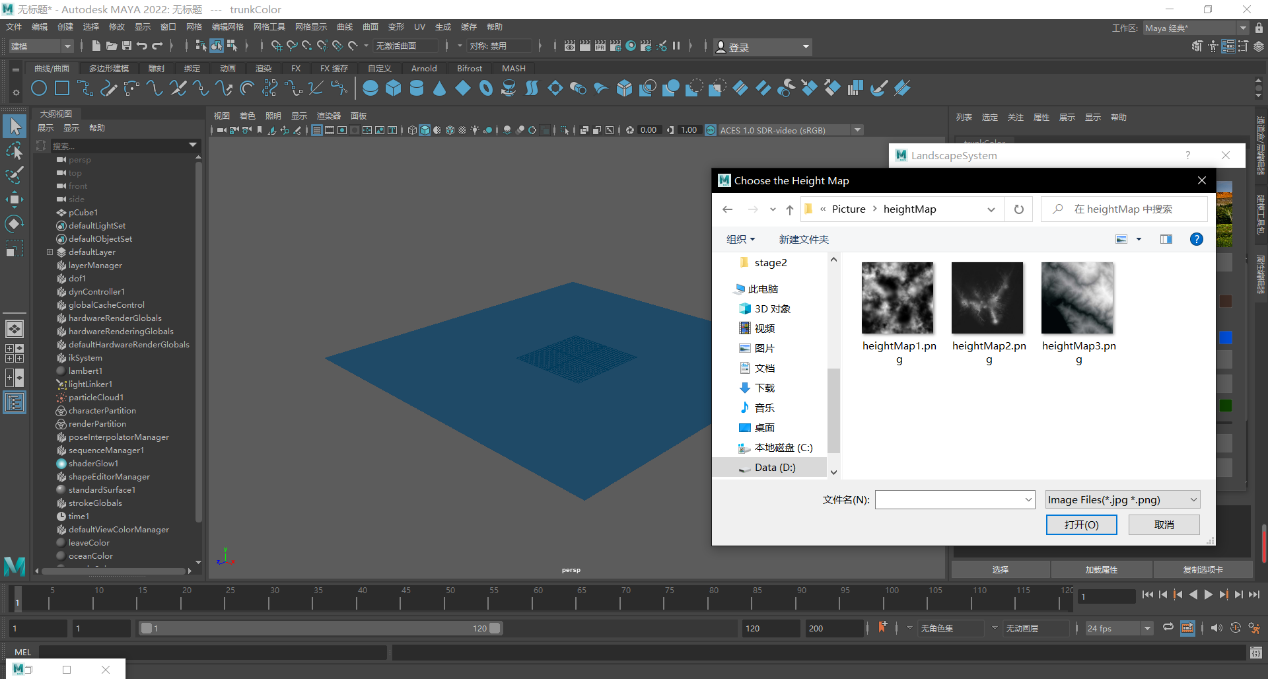
Pyside2 (no need to install if you use Maya2022 because Maya2022 has had this modules)

**Run script**

Copy all the code in the file named "main.py" to the python editor in maya and run it. The system will pop up a window asking to select the working environment, just select the entire folder named "Landscape System". Then the user interface will appear.

**User interface**

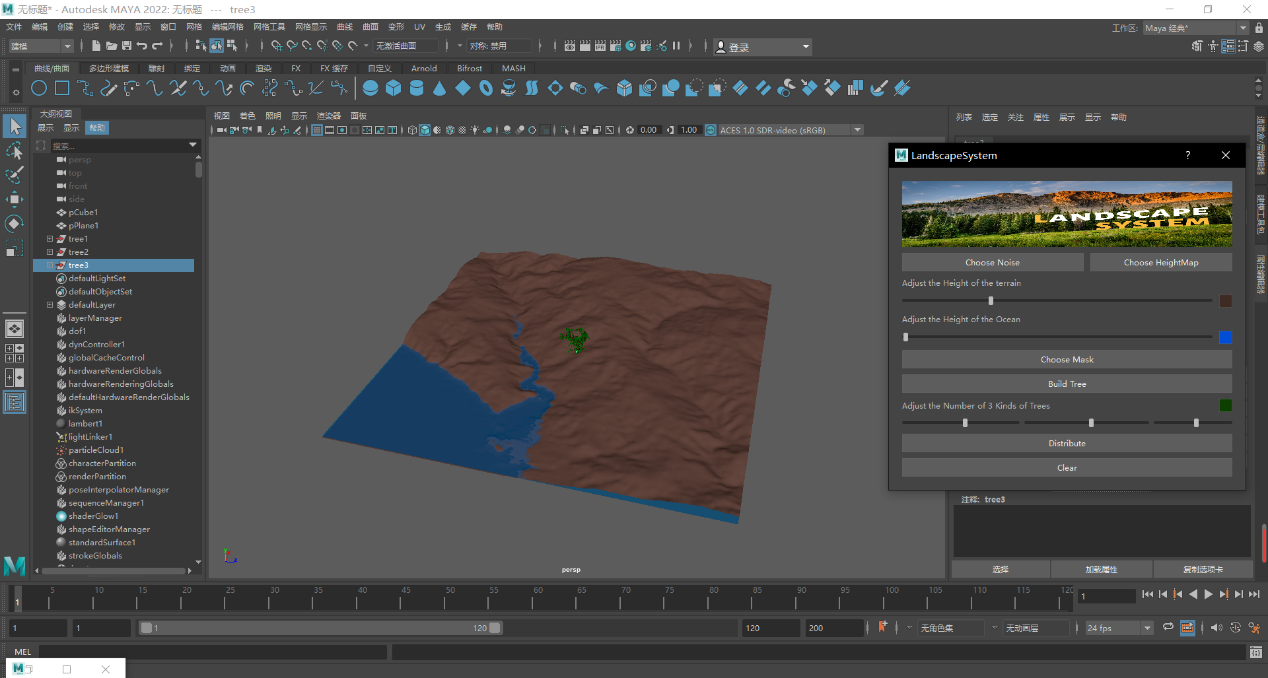
As shown in the figure, the user must first choose a way to generate terrain. Selecting the noise button will directly generate the terrain based on the noise map. Selecting the heightmap button will pop up a window asking to select a picture, and select a square whose pixels are less than 256\*256. The image (which can be a color image) will generate the terrain based on the heightmap. You can find the heightmap we prepared for you in the folder "Landscape System\artefacts\Picture\heightMap"



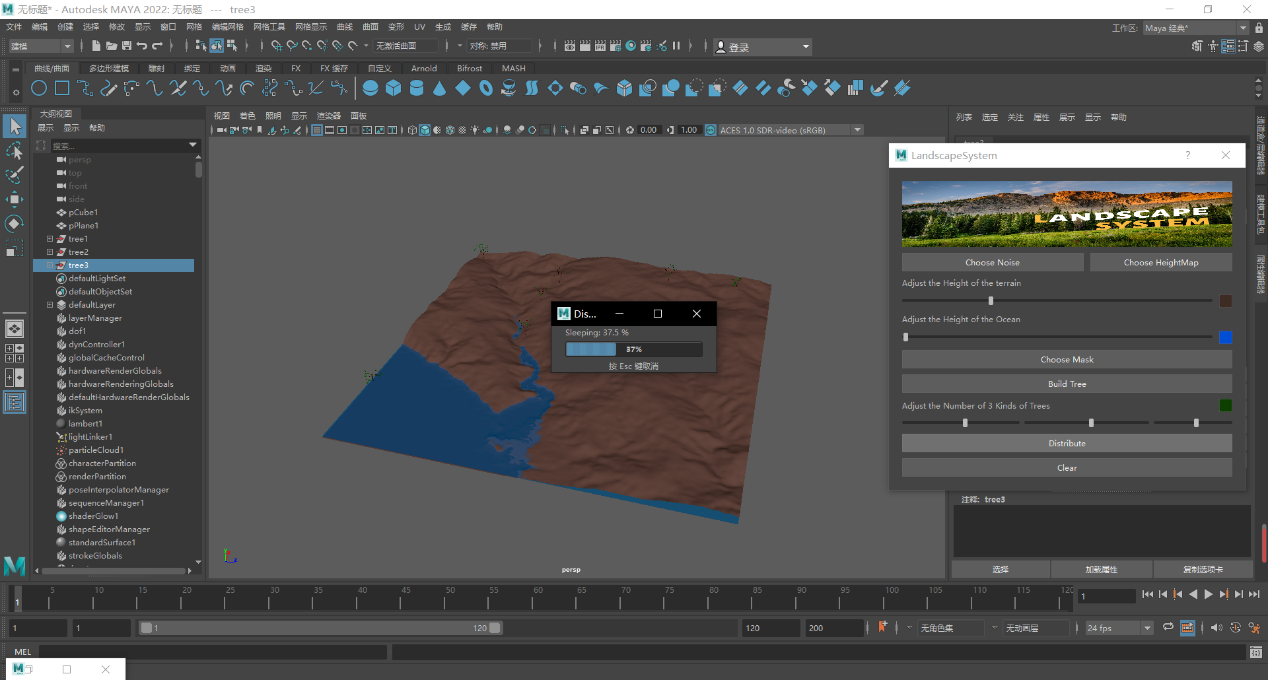
You can adjust the overall height of the terrain in the slider below, and you can change the color at the same time. We also specially designed a sea level, you can also adjust its color and height

Next, you need to use mask for region selection. Here you need to select a black and white bitmap, black represents the selected area. It is worth mentioning that if noise is used, the pixel size of the image is required to be 200\*200. If heightmap is used, the pixel size of the image needs to be the same as the imported heightmap. You can find the mask image we prepared for you in the folder with the path "Landscape System\artefacts\Picture\mask"

Then generate the plant model, click the "Build Tree" button, the system will automatically generate 3 plant models according to the L-system, and then you can adjust the number of each plant distribution and the color of the leaves.



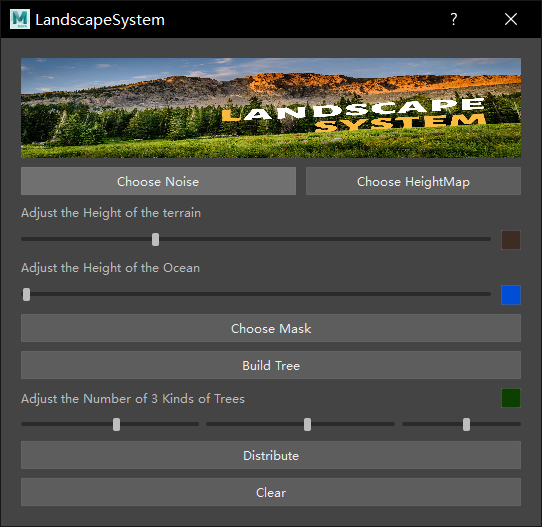
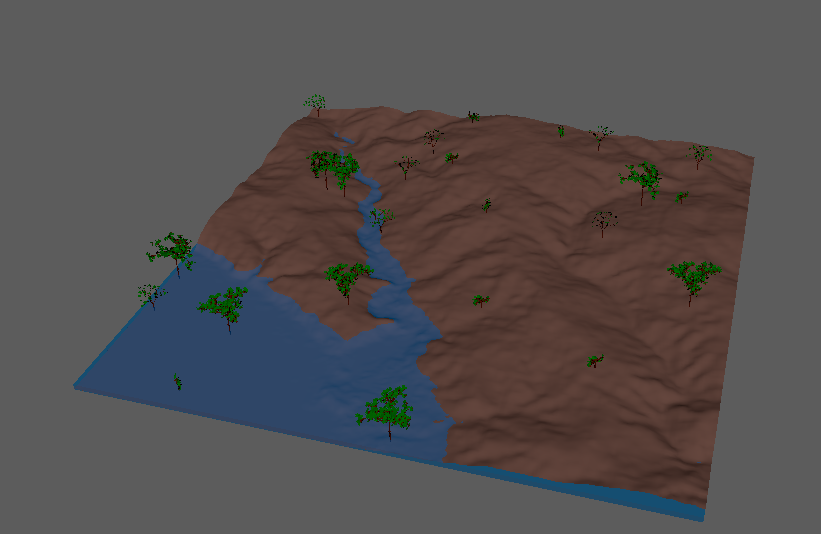
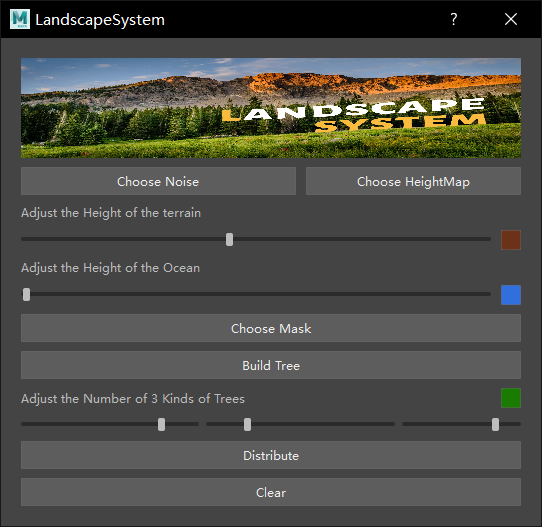
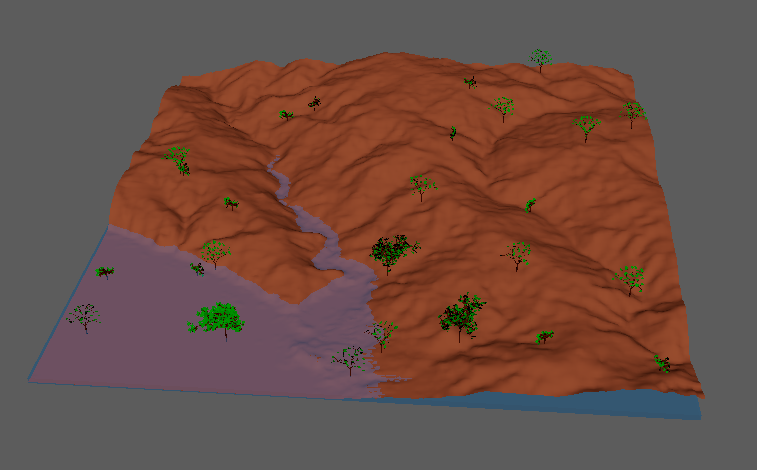
Finally, click the "Distribute" button to generate the final model. A progress bar will appear indicating your current progress.

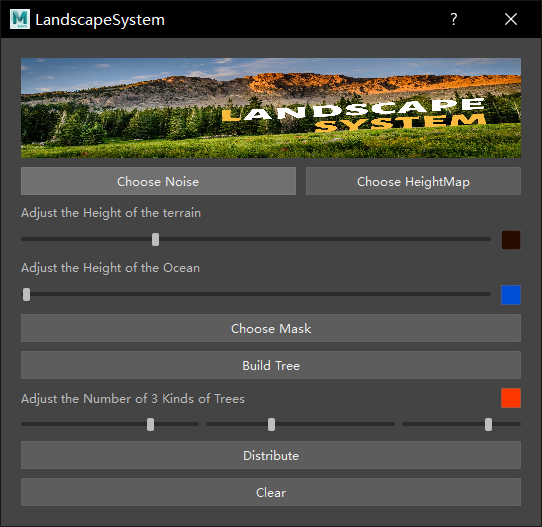
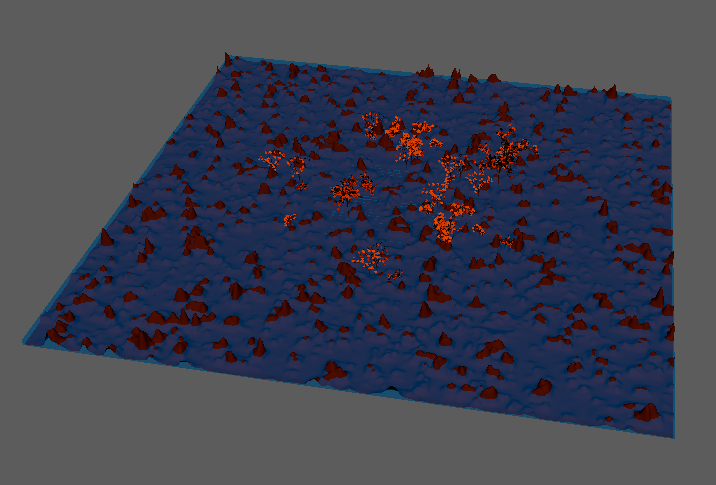


If you want to regenerate the model, you can click the "Clear" button to delete the generated model and repeat the above operation process.

**Results analysis**

We tested the effect of model simulations using different terrain generation methods and setting different variables:

1.  use height map, and other parameters are set as follows.
2.  use height map, and other parameters are set as follows.
3. use noise map, and other parameters are set as follows.

****

**Shortcomings and Future work**

Although the current project has implemented many functions, there are some flaws and some unimplemented functions that we expect, which will become our future work.

Flaws:

1.The water surface will directly submerge the plants

2.The types of L-system are not rich enough

3.The function of the noise map to generate terrain can be optimized

Unimplemented functions:

1.Plant distribution according to environmental factors

2.Animation of plant growth