**Water Cells**

Maintain a list of water cells. Each has the following data:

X coordinate in the grid

Y coordinate in the grid

Distance left to go ( I thought this could be derived from velocity)

Height

**Sand Cells**

Maintain a list of sand cells. Each has the following data:

X coordinate in the grid

Y coordinate in the grid

Height

Wetness

Height of both the sand and the water as well as the wetness value are in the same units. Say we have determined that each grid square represents one square centimeter, and we are using a 10 by 10 grid. We could have height range from 0 to 1000, where a height of 1000 corresponds to a height of 10 cm.

Note: I assume a standard coordinate system with the origin in the bottom left corner

**Initial Set Up**

For each shape, we ensure that the sum of all heights of the sand cells is the same to start off with, ensuring the same volume of sand per castle.

The sand cells have coordinates anywhere in the grid, start with a given height, and start with 0 wetness. The water cells start off in a line at the bottom of the grid, each with a specified height and distance to go for that trial. (We’ll assume that on any given “wave” all the water cells in that wave have the same starting distance to go and the same starting height). Water cells will move forward in each time step, losing some height to the sand that they get wet. The cells will be removed once they have no height left, or once they have no distance left to go. The sand cells will get some wetness from the water cells. Once the sand is “fully wet” and unstable, meaning that the wetness of the sand is greater than some amount, any additional wetness will cause the height to decrease. The sand will readjust itself as well to fall into place to maintain a max angle of 45 degrees between any neighboring cells. This means the sand can fall outside the edges of the castle too, and if the sand escapes a square region that is larger than the sand castle shape, it will be removed.

The model could be driven by the water cell iteration function, which could call a sand readjustment function after each “wave”

**Water Cell Iteration Pseudocode**

For as many iterations as we choose

For each Water cell

Increment the y coordinate of the cell

If the cell’s coordinates match the coordinates of some sand cell

Increase the Sand’s wetness by the height of the water

Decrease the water cell’s height by the amount of wetness it gave to the sand cell

Decrease the distance the water has to go by some amount (this amount will be greater than if the water wasn’t moving through sand)

If the cell’s coordinates do not match any coordinates of a sand cell

Decrease the distance the water has to go by 1

If the distance for the water cell to go is <= 0, remove it

If the height of the water cell is <= 0, remove it from the list

Call the function to let the sand cells readjust themselves.

Finished simulation

**Sand Readjustment Pseudocode**

For each sand cell

If the wetness is greater than some fixed amount, take loss = wetness – fixed amount

Decrease the sand height by the loss as determined above

Set the wetness to the fixed amount

If the height of the sand is <= 0, remove it from the list

For each sand cell

For each of the cell’s 8 neighbor possibilities (At each bordering edge or corner)

If there is a sand cell bordering in that spot

Compare the heights of the two cells, if the difference warrants an exchange of sand, perform the exchange

If the spot is off of the pre-determined square grid

Look at the height of the sand, and if the height warrants rearrangement just decrease the sand height by that amount

If there is not a sand cell at that spot, and it is within the grid

Look at the height of the sand, and if the height warrants rearrangement add a new sand cell in that spot with some height. Remove that height from the current sand cell.

**Specified amount of sand to Exchange**

If the difference in sand columns warrants an exchange, give the neighboring sand column 1/3 of the sand required to make the angle in between them 45 degrees. This will help ensure that sand can be spread evenly. The value of the exchange can be calculated as follows:

First, we compute the maximum height difference:

If the cells are each X cm^2, and we choose a height such that Y height units measures Z cm, then the maximum distance between sandpiles to achieve a maximum angle of 45 degrees is m = XY/Z .

For each sandpile, to figure out how many height units to donate we take. ((Height of center pile – Height of bordering pile) – m)/6

(We divide by 2 and then 3)