Project 8.3 Growing dendrites

In this project, you will simulate a phenomenon known as *diffusion-limited aggregation* (DLA). DLA models the growth of clusters of particles such as snowflakes, soot, and some mineral deposits known as dendrites. In this type of system, we have some particles (or molecules) of a substance in a low concentration in a fluid (liquid or gas). The particles move according to Brownian motion (i.e., a random walk) and "stick" to each other when they come into contact. In two dimensions, this process creates patterns like the one in Figure 1. These structures are known as *Brownian trees* and they are *fractals* in a mathematical sense.

To simplify matters, we will model the fluid as a two dimensional grid with integer coordinates. The lower left-hand corner will have coordinates (0,0) and the upper right-hand corner will have coordinates $(199, 199)$. We will place an initial seed particle in the middle at $(100, 100)$, which we will refer to as the origin. We will only let particles occupy discrete positions in this grid.

In the simulation, we will introduce new particles into the system, one at a time, some distance from the growing cluster. Each of these new particles will follow a random walk on the grid (as in Section 5.1), either eventually sticking to a fixed particle in the cluster or walking far enough away that we abandon it and move on to the next particle. If the particle sticks, it will remain in that position for the remainder of the simulation.

To keep track of the position of each fixed particle in the growing cluster, we will need

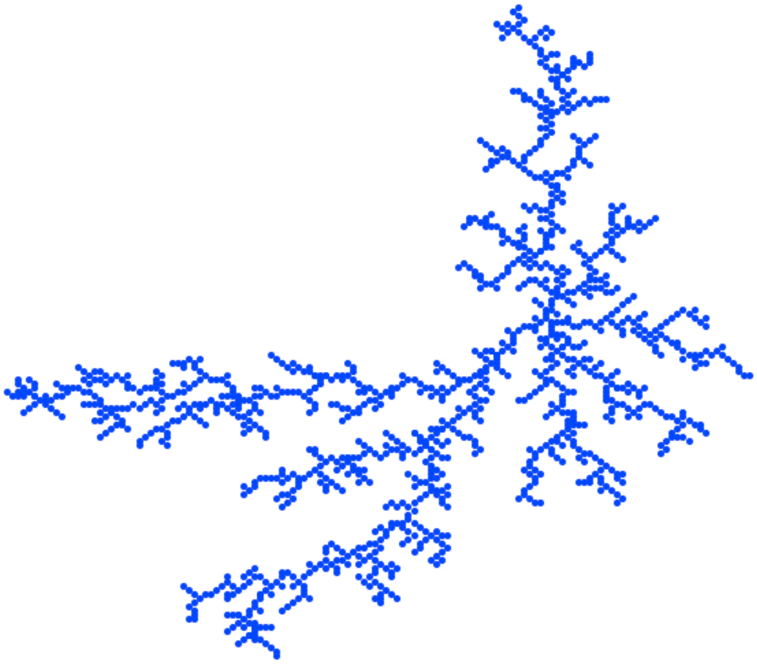

Figure 1 A two-dimensional Brownian tree.

a 200 × 200 grid. Each element in the grid will be either zero or one, depending upon whether there is a particle fixed in that location. To visualize the growing cluster, you will also set up a turtle graphics window with world coordinates matching the dimensions of your grid. When a particle sticks, place a dot in that location. Over time, you should see the cluster emerge.

The simulation can be described in five steps:

1. Initialize your turtle graphics window and your grid.
2. Place a seed particle at the origin (both in your grid and graphically), and set the radius R of the cluster to be 0. As your cluster grows, R will keep track of the maximum distance of a fixed particle from the origin.
3. Release a new particle in a random position at a distance of $R + 1$ from the origin. (You will need to choose a random angle relative to the origin and then use trigonometry to find the coordinates of the random particle position.)
4. Let the new particle perform a random walk on the grid. If, at any step, the particle comes into contact with an existing particle, it "sticks." Update your grid, draw a dot at this location, and update R if this particle is further from the origin than the previous radius. If the particle does not stick within 200 moves, abandon it and create a new particle. (In a more realistic simulation, you would only abandon the particle when it wanders outside of a circle with a radius that is some function of R.)
5. Repeat the previous two steps for `particles` particles.

Write a program that implements this simulation.

Additional challenges

There are several ways in which this model can be enhanced and/or explored further. Here are some ideas:

- Implement a three-dimensional DLA simulation. A video illustrating this can be found on the book website. Most aspects of your program will need to be modified to work on a three-dimensional grid. To visualize the cluster in three dimensions, you can use a module called `vpython`.

  For instructions on how to install VPython, visit `http://vpython.org`. Once you have `vpython` installed, you can import it with

  ```
  import vpython as vp
  ```

  Placing a sphere at a particle location looks like this:

  ```
  vp.sphere(pos = vp.vector(0, 0, 0), radius = 1, color = vp.color.blue)
  ```

  This particular call draws a blue sphere with radius 1 at the origin.

  At any time, you can zoom the image by dragging while holding down the Option key. You can rotate the image by dragging while holding down the

Control key. There is very good documentation for `vpython` available on their website.

- Investigate what happens if the particles are “stickier” (i.e., can stick from further away).
- What if you start with a line of seed particles across the bottom instead of a single particle? Or a circle of seed particles and grow inward? (Your definition of distance or radius will need to change, of course, in these cases.)