In-Class Work: Random Walks & Fractal Growth

2. Random Walk in One Dimension

2c. We will all start today in class with the python program for 2c from last class:
`~kvollmay/share.dir/inclass2019.dir/classrndwalk2c.py`
If you had finished in last class the task for 2c already, just have a quick look at this program. You may use either your finished program from last class or this program, or if you had not finished in last class, then copy this program into your working directory and continue working with for the following tasks.

Draw a flow chart for this program.

2d. Next no longer use the number of steps `NSTEPS` as constant but instead add a loop over N=`nsteps` to your program of 2c. Add this loop to your flow chart. Loop `nsteps` from `100` to 2000 in steps of `100`. For each `nstep` print out N=`nstep` and $\langle x^2 \rangle$. Look at the resulting $\langle x(N)$ and also $\langle x^2(N) \rangle$. Please get me when you got this done, so that we can interpret your results with the class. Try if you can derive a theoretical prediction.

2e. In the following steps we will use `gnuplot` to fit the resulting $\langle x^2(N) \rangle$ simulation data. First save your data with `./classrndwalk2d.py > dat2d`, Then type the command

```
gnuplot
```

This will start a session in the graphics-tool `gnuplot`. To do a power law fit and to look at the comparison of the fitted line and your simulation data type

```
a=1;b=1;f(x)=a*x**b;fit f(x) "dat2d" using 1:3 via a,b;
plot "dat2d" using 1:3,f(x)
```

1. Fractal Growth: Random Walk in Two Dimensions

1a. For the fractal growth DLA model we will need a random walk in two dimensions. Write a python-program for a random walker on a two dimensional lattice (all four directions being equally likely), starting at $x = 0$ and $y = 0$ and (print and) look at $x(t)$ and $y(t)$. You may use the solution program `~kvollmay/share.dir/inclass2019.dir/classrndwalk2a.py`.
For looking at $x(t)$ and $y(t)$ in the same figure, you can use the command (assuming that your program is called `classfractal1a.py`)

```
./classfractal1a.py > j; xmgrace -block j -bxy 1:2 -bxy 1:3
```

1b. Movie

Next let's make a movie of your random walk. Define a lattice (`lattice`) of size 100x100 and initialize it for all sites equal to zero. Put your initial walker at site $x = 50$ and $y = 50$. We want to make a movie of the random walker where we mark on the lattice the current random walker site with the lattice value 2 and we mark any previously visited site with `1` (This is just for our fun.). To make a movie we first make an image for every random-walk step. (**So please use only `NSTEPS=50` random walk steps!**) To see how to make these pictures

see the example
`~kvollmay/share.dir/pythonsamples.dir/sample_latticemovie.py` Once you have all pictures in `frame*` you can run the movie with
`animate -delay 10 -pause 5 frame*`

2. Fractal Growth: DLA

2a. You may use google to get an idea about the main concepts of the following topics:

- Cellular Automata
- Fractal Growth
- Diffusion-Limited Aggregation

2b. Read in our Gould & Tobochnik textbook about the model we will use, that is about Diffusion-Limited Aggregation (DLA): chapter 13, bottom of page 529 and page 532 Problem a. (For a link to this book, see our course webpage.) Try to write a flow chart for this DLA model.
Please get me, when you get here. I will give a mini-intro about cellular automata and about fractals.

2c. Fractal Growth: Background (if time)

Read in our Gould & Tobochnik textbook §13.3 Kinetic Growth Processes (pages 519ff, text only, no problems, no JAVA code) Epidemic model, Eden model, and diffusion limited aggregation.