In-Class Work: Random Walks & Fractal Growth

2. Random Walk in One Dimension (continuation of Feb. 2 class)

In the following I will guide you through 2a - 2c. You will then work on 2d.

Next we will do important analysis on the random walk. To simplify the task (and yet being able to get the main concept) let us start with the random walk in one dimension.

2a. You wrote a program for a random walk in one dimension. In all following we assume $p = q = 0.5$, i.e. there is equal probability to jump to the right or to the left with jump size 1, $x(t = 0) = 0$. You printed t and x for `NSTEPS = 5000` time steps and looked at the resulting $x(t)$ with `xmgrace`.

2b. The next step is a preparation for 2c. Instead of printing every time step, print only once after $N =$NSTEPS time steps (i.e. after the time-loop) the resulting $x(N)$ and $(x(N))^2$.

2c. Now add a loop over simulation runs to your program from 2b. Each simulation run starts with $x = 0$ and gives you an $x(\mathtt{NSTEPS})$ and an $x^2(\mathtt{NSTEPS})$. First set `NSIMRUNS=10` and print for each simulation run $x(\mathtt{NSTEPS})$ and $x^2(\mathtt{NSTEPS})$. Next change your program such that it determines the average over `NSIMRUNS=10000` simulation runs of x and an x^2 and prints out the resulting averages $\langle x\rangle$ and $\langle x^2\rangle$.

2d. (START HERE) In case you had already gotten 2c finished in our Feb.2 class, you may continue with your program. In case you had not yet finished 2c, use the following copy command:

```
cp ~kvollmay/share.dir/inclass.dir/classrndwalk2c.py  inclassrndwalk2d.py
```

Modify `inclassrndwalk2d.py` such that the number of steps `NSTEPS` is no longer a constant but instead add a loop over N=`nsteps` to your program of 2c. 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 `./inclassrndwalk2d.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)
```

In-Class Work: Fractal Growth

1. 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/inclass.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: Background (if time)

Read in our Gould & Tobochnik textbook §13.3 Kinetic Growth Processes (text only, no problems, no JAVA code) Epidemic model, Eden model, and diffusion limited aggregation. (for a link to the book see our webpage) Please get me, when you get here. I will give a mini-intro about cellular automata and about fractals.