Listing 1: Python example

```
N_in = 1000
N_out = 500
p_in = .4
p_out = .6
numsteps = 1000
T = np.arange(150)
p_T = np.zeros(len(T))

# Simulate realizations compared against each threshhold T
for k in np.arange(0, len(T)):
    spike = 0.0

    # Simulate lots of realizations of N
    for i in np.arange(0, numsteps):
        # Generates a coin toss vector for the inward and outward channels
        # in which 'open' corresponds to 1. Sums the vector to ascertain
        # the number of open channels in a realization.
        N_in_open = np.sum(np.random.choice([1, 0], size = N_in, p = [p_in, 1 -
        N_out_open = np.sum(np.random.choice([1,0], size = N_out, p = [p_out, 1
        # The net current, open inward channels minus open outward.
        net_current = N_in_open - N_out_open
        # IF the net current is more than T, a spike occurs
        if net_current > k:
            spike = spike + 1.0
    # Once numsteps realizations have been simulated, divides the number
    # of spikes by the total number of realizations to get the probability
    # that a spike will occur with a given threshhold T
    p_T[k] = spike/numsteps
```