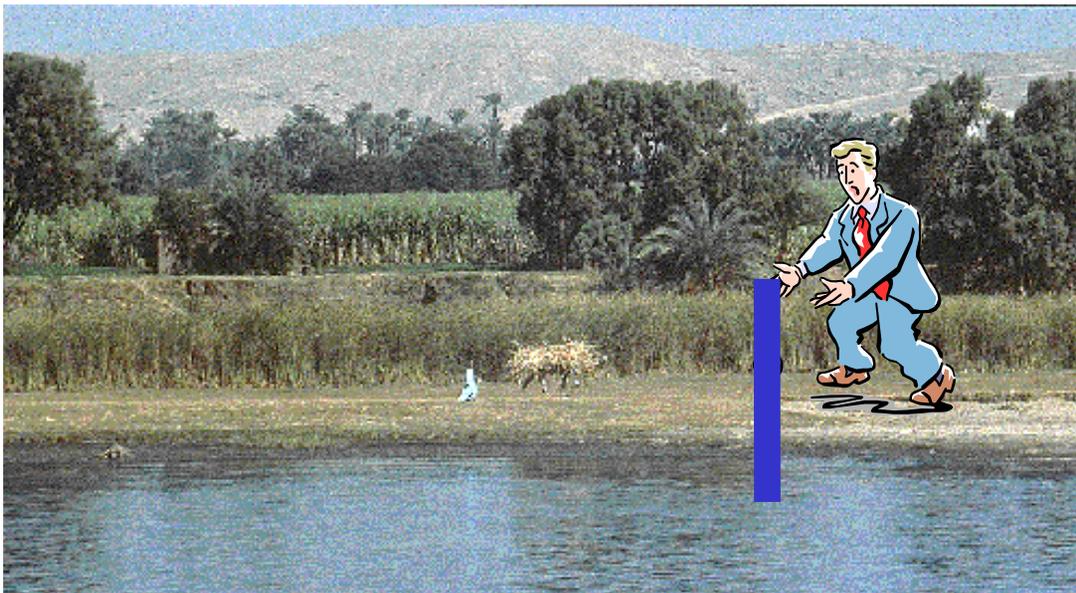


# Monte Carlo

**Solving problems using random number generation.**

**Extremely useful in solving noisy or analytically intractable problems!**



measuring the  
volume of the  
Nile

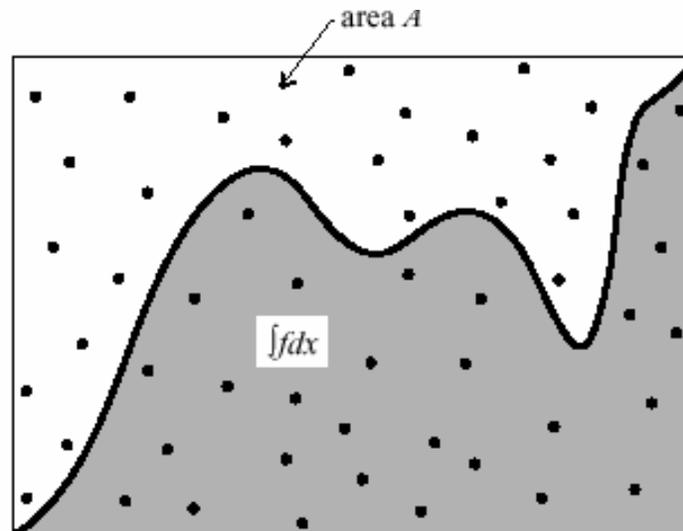


# Useful information you can get after doing many trials of your system:

Mean:  $\mu = \sum_i x_i P(x_i)$ .  $P(x_i)$  estimated from your trials as:  
#sightings of  $x_i$  / total # trials

Standard Deviation:  $s_N = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}$ .  
num trials (pointing to  $N$ )  
mean (pointing to  $\bar{x}$ )

Integral evaluation:



Other stuff...

# Get information about biological systems!

Molecular Monte Carlo lets you explore many interesting properties...

- thermally-averaged structures
- molecular charge distributions
- reaction rate constants
- free energies
- dielectric constants
- heat capacities
- phase transition temperatures
- just about anything.



# Easier things to try right now

We can use Monte Carlo to estimate a whole *probability distribution function* that might otherwise be very difficult to compute

Or, we can take a predefined distribution function and randomly draw numbers from it to learn things about the distribution

# Generating random numbers from a specific distribution

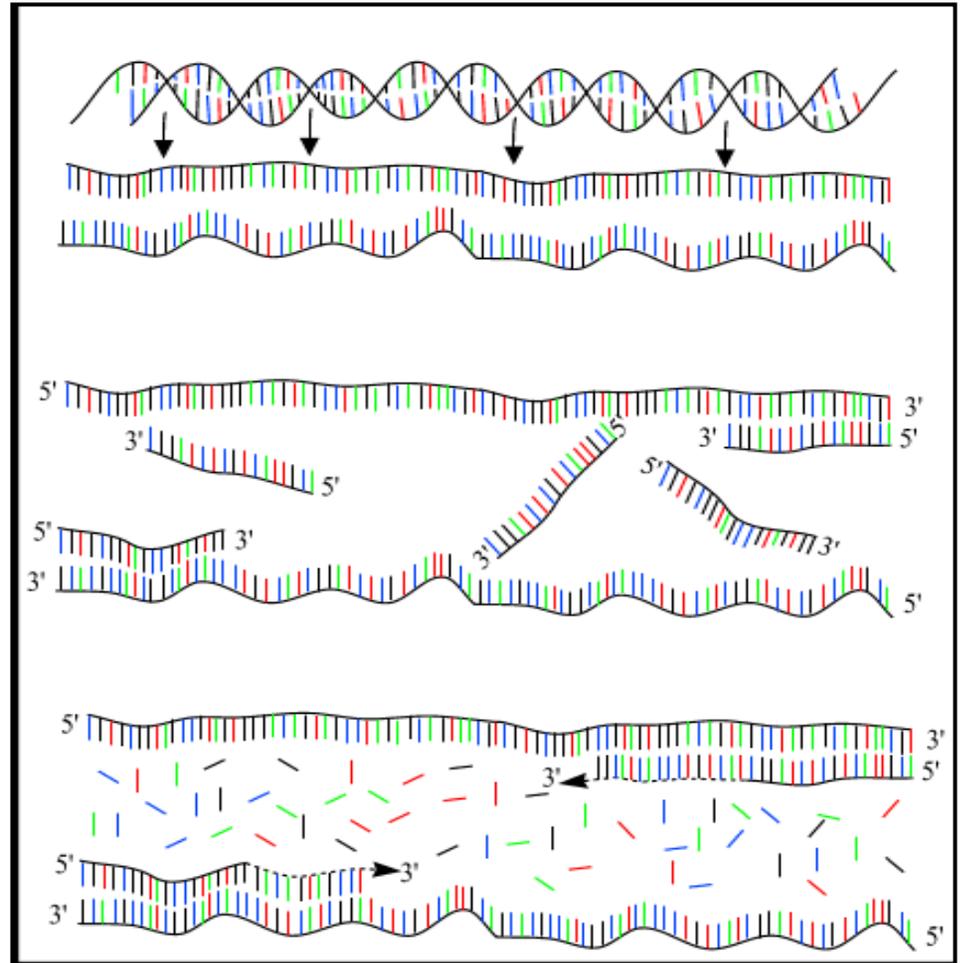
- Almost always start by drawing random numbers from a *uniform distribution* where all values between 0 and 1 are equally likely
- Many programming languages have such “random” number generators built-in

# MATLAB random number generators

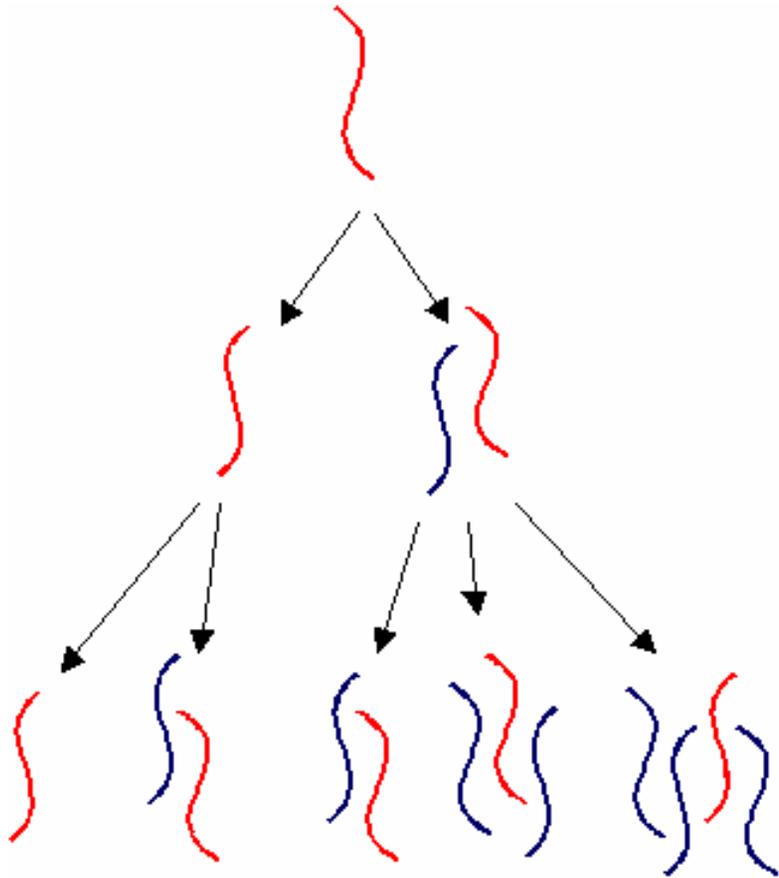
- MATLAB has a ton of built-in random number generators so that you don't have to worry about the algorithms
- A great demo of these generators is  
randtool.m
- You can try it out by typing 'randtool' at the matlab prompt in MIT server

# Polymerase Chain Reaction: A brief introduction

- Denaturation:
  - Heat sample to 94°C to denature the DNA
- Annealing
  - Add primers
  - Reduce temp. to 54°C
- Extension
  - Increase temp. to 72°C
  - Add dNTPs and thermophilic polymerase (Taq)
- And repeat!



# How can we represent PCR statistically?



PCR is an example of a branching process. During each cycle of the reaction, we can generate one or two “branches” for each existing copy of DNA (i.e. each strand either gets copied or it doesn't)

# Scary model for PCR



Probability that PCR tube contains  $n_1$  copies given that we started with  $n_0$ :

$$P(n_1 | n_0 \{\varepsilon\}) = \binom{n_0}{n_1 - n_0} \varepsilon^{n_1 - n_0} (1 - \varepsilon)^{2n_0 - n_1}$$

Apply Bayesian sum and product rule:

$$P(A | C) = \sum_B P(A | B)P(B | C)$$

And obtain an expression for the probability that after  $k$  cycles the tube contains  $n_k$  copies:

$$P(n_k | n_0 \{\varepsilon\}) = \sum_{n_{k-1}} \sum_{n_{k-2}} \cdots \sum_{n_1} P(n_k | n_{k-1} \{\varepsilon\}) P(n_{k-1} | n_{k-2} \{\varepsilon\}) \cdots P(n_1 | n_0 \{\varepsilon\})$$

# Brownian motion and diffusion

- Brownian motion: the constant and seemingly random motion of particles due to the kinetic energy imparted on one another as they collide
- Diffusion: the macroscopic phenomenon by which we observe the spreading out of particles in space due to accumulated Brownian motion

[http://www.phys.virginia.edu/classes/109N/more\\_stuff/Applets/brownian/brownian.html](http://www.phys.virginia.edu/classes/109N/more_stuff/Applets/brownian/brownian.html)

# Brownian Motion

- Not really random—if knew the location and velocity of every particle we could predict trajectories at any later time
- Realistically this is impossible for system with large number of particles
- Describe by the *random walk*