Random Number Generation

11-5-2003

Opening Discussion

- What did we talk about last class?
- Discussion of the minute essays from last time.
- What do we mean when we talk about random numbers on a computer? What does it means for a number to be random?

File Code From Last Class

- Let's now go to the code for files that we started last time. It was a simple line editor that we wanted to have the ability to read from a file and write to a file.

Random Numbers on Deterministic Machines

- You have already seen a use of the rand() function if you did the encrypt program. In reality that gives you a sequence of pseudo-random numbers.
- Computers are deterministic, given the same initial conditions and instructions you get identical behavior. As such, nothing is truly random on a computer. Instead we make a sequence where elements don't seem closely related.

The Method

- The rand() function uses what is called a linear congruential uniform generator. This uses a simple formula to get a sequence of numbers that can have a long periodicity.

$$x_{n+1}=(a*x_n+c) \bmod m$$

- Sequence depends on a, c, m, and x_0. The last one is the "seed".

The Details

- m is generally chosen to be a power of 2 to make the math faster because the modulo operator can be done by preserving the lower bits.
- The period for that sequence can be m iff, c is relatively prime to m, a%p=1 for every prime factor p of m, and a%4=1 of m is divisible by 4.

Code

- First let’s work through an example of this method on the board. Then we can write some code to implement a slightly larger random number generator and look at the rand and srand functions in stdlib.h.

Minute Essay

- There is no class on Friday because I’ll be driving to Baton Rouge for the regional ACM programming competition.
- Quiz #5 is on Monday and assignment #7 is due a week from Friday.