VJ Davey

CSci 426 Notes Section 2.1

Lehmer Random Number Generation

1. **Random Number Generation**
   Algorithmic Generators satisfy the well accepted Random Number Generation criteria:
   Randomness...Controllability...Portability...Efficiency...Documentation
   An ideal generator is a function that produces a real number between 0 and 1 where each number has an equal chance of being selected.
   We would want to be able to choose any number m in a set $x_m = \{1,2,3...m\}$.

2. **Lehmer's Algorithm**
   Is defined in terms of two fixed parameters:
   modulus m, a fixed large prime integer.
   multiplier a, a fixed integer in x_m.

   And the subsequent generation of the integer sequence $\{x_0, x_1, x_2...x_m\}$ via the iterative equation

   $$x_{i+1} = g(x_i)$$

   where $g(x)$ is defined for all $x \in x_m$ as

   $$g(x) = ax \mod m$$

   The initial seed x_0 is chosen from the set x_m.
   The modulus operation always causes the remainder to fall between 0 and $m-1$.
   If 0 is used as a seed, all subsequent values of the sequence will be 0. So, $g(x) \neq 0$ for any $x \in x_m$.
   $g : x_m \rightarrow x_m$
   There is nothing actually random about a random number generator.
   When choosing (a,m) we need the function to generate a full period sequence, and we need the sequence to appear to be random.

Full Period Multipliers
The following algorithm can be sed to determine whether a multiplier a is full period relative to the prime modulus m:

```
p = 1;
x = a;
while (x != 1) {
    p++;
    x = (a * x) % m;
}
if (p == m - 1) {
        //a is a full period multiplier
}else{
    //a is not a full period multiplier
}
```