# Cryptographic Random Numbers

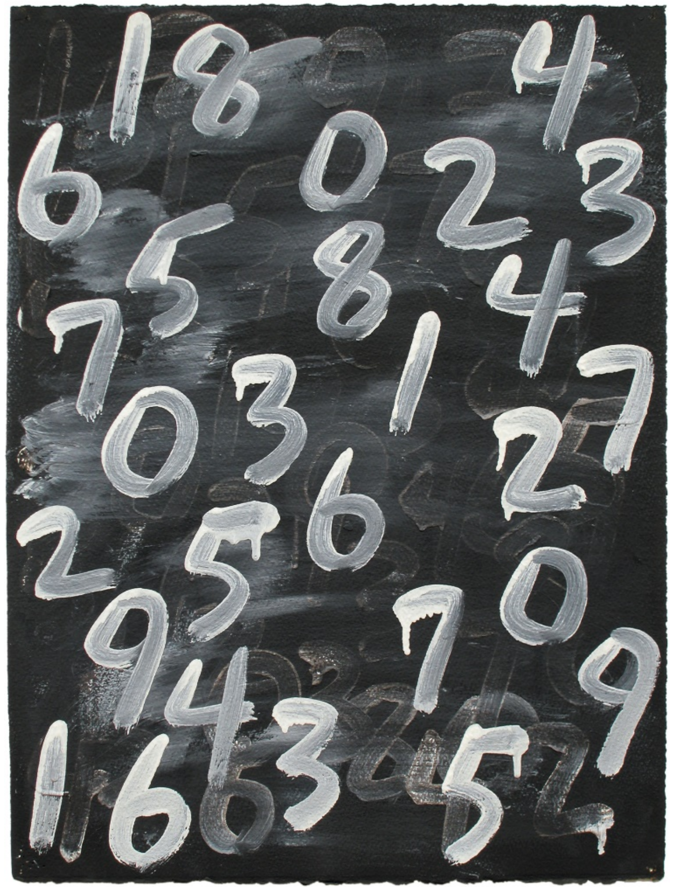
Stephen Haunts

@stephenhaunts | www.stephenhaunts.com

# Overview

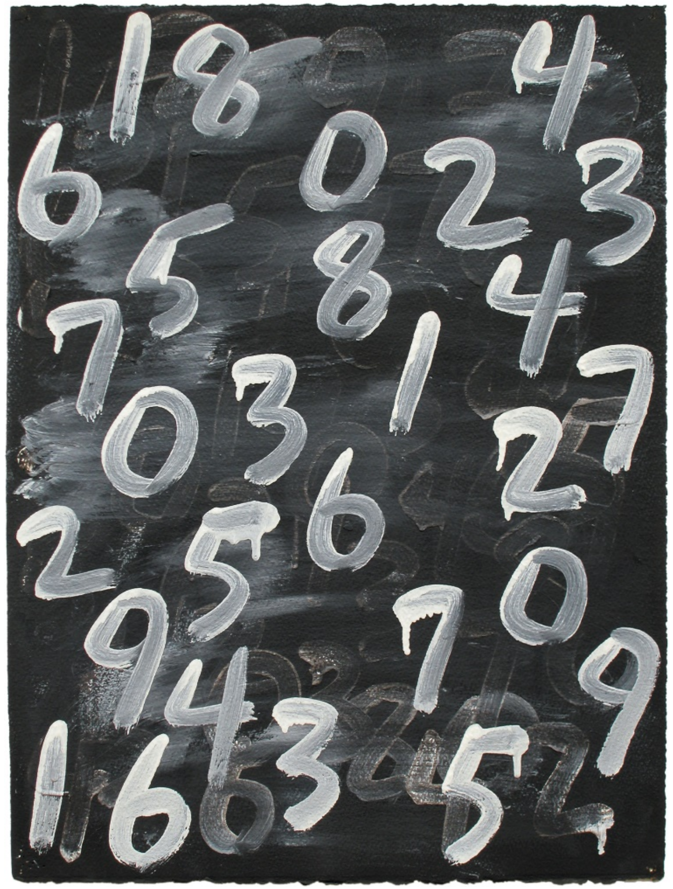
- Why are random numbers important?
- ***System.Random*** and its problems
- Secure random numbers with ***RNGCryptoServiceProvider***
- Code Demo

# Why Are Random Numbers Important?



- Used for generating encryption keys
- Software based random numbers are not always truly random
- Randomness can be created from human interaction
- Not practical for server applications

# Why Are Random Numbers Important?



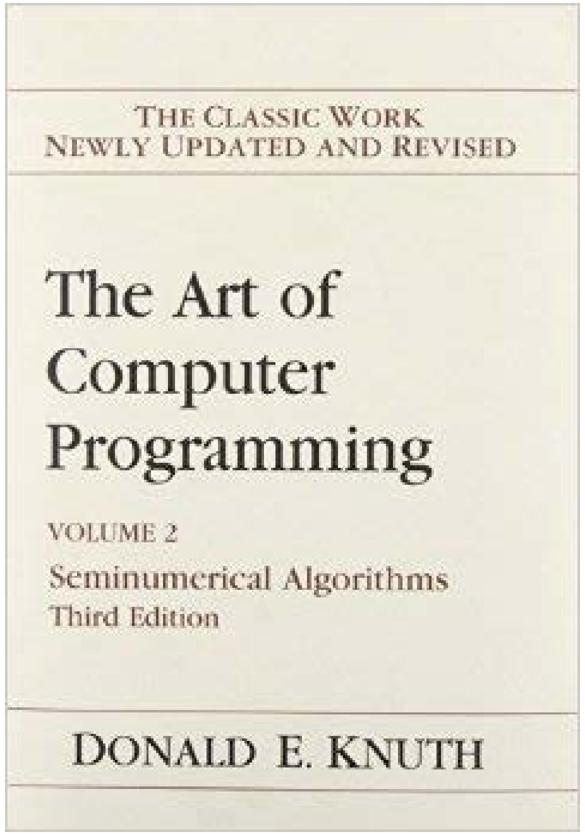
- Can use dedicated hardware or a specifically designed algorithm

# System.Random and Its Problems

- ***System.Random*** is a pseudo random number generator
- A seed value is passed into the constructor
- The seed value should be different each time
- ***System.Random*** is deterministic and predictable

# System.Random and Its Problems

- Based on the Subtractive Random Number Generator by Donald E. Knuth



THE CLASSIC WORK
NEWLY UPDATED AND REVISED

The Art of
Computer
Programming

VOLUME 2

Seminumerical Algorithms
Third Edition

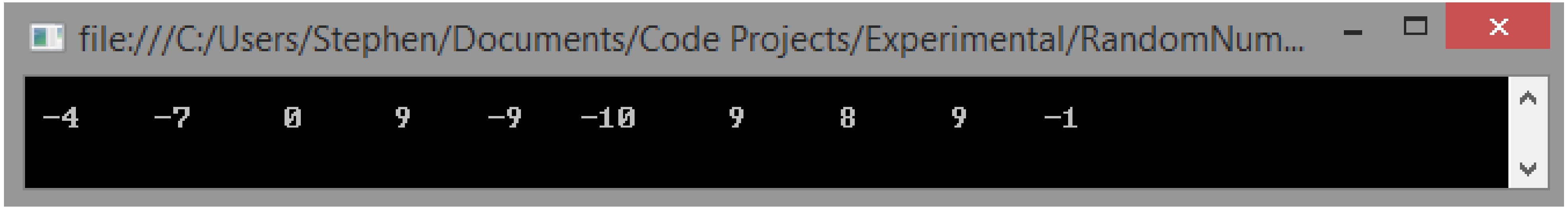
DONALD E. KNUTH

# System.Random and Its Problems

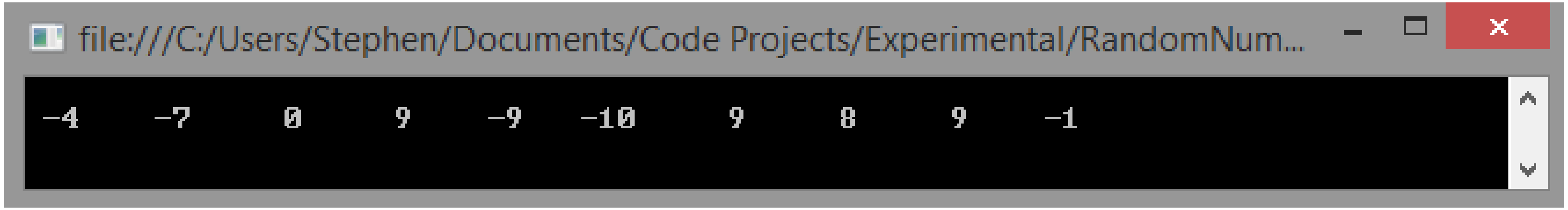
```
Random rnd = new Random(250);

for (int ctr = 0; ctr < 10; ctr++)
{
   Console.Write("{0,3}   ", rnd.Next(-10, 11));
}
```

# System.Random and Its Problems


file:///C:/Users/Stephen/Documents/Code Projects/Experimental/RandomNum...

```
-4      -7       0       9      -9     -10       9       8       9      -1
```


file:///C:/Users/Stephen/Documents/Code Projects/Experimental/RandomNum...

```
-4      -7       0       9      -9     -10       9       8       9      -1
```

# System.Random and Its Problems

- Microsoft recommends creating 1 instance of ***System.Random*** to generate numbers for your application
  - http://bit.ly/1CKgPUf
- ***System.Random*** is not thread safe

# Secure Random Numbers with RNGCryptoServiceProvider

- Good random numbers are important in Cryptography
- Random numbers used for creating encryption keys and for hashing
- ***System.Random*** is not good for non-deterministic random numbers
- ***RNGCryptoServiceProvider*** is a more secure way to generate random numbers
- ***RNGCryptoServiceProvider*** is slower to execute than ***System.Random***
- Performance is a small trade-off for generating encryption keys

# Secure Random Numbers with RNGCryptoServiceProvider

```
public static byte[] GenerateRandomNumber(int length)
{
    using (var randomNumberGenerator = new RNGCryptoServiceProvider())
    {
        var randomNumber = new byte[length];
        randomNumberGenerator.GetBytes(randomNumber);

        return randomNumber;
    }
}
```

# Code Demo

## How to Use RNGCryptoServiceProvider

# Module Summary

- ***System.Random*** is not truly random
- ***RNGCryptoServiceProvider*** is designed for cryptographic operations