Nuggets of Pseudorandomness
Cynthia Dwork, Omer Reingold (MSR-SVC, {dwork, omreing}@microsoft.com)

1. Cryptography: Good Pseudorandom Generators are Crucial
   - With them, we have one-time pad (and more):
     - short key $K_s = 110$
     - derived key $K = 01101011$
   - Without, keys are bad, algorithms are worthless (theoretical & practical)

2. Data Structures & Hash Functions
   - Linear Probing:
     - If $F$ is random then insertion time and query time are $O(1)$ (in expectation).
     - But where do you store a random function?!? Derandomize!
     - Heuristic: use SHA1, MD4, ...
     - Recently (2007): 5-wise independent functions are sufficient*
     - Similar considerations all over: bloom filters, cuckoo hashing, bit-vectors, ...

3. Weak Sources & Randomness Extractors
   - Available random bits are biased and correlated
   - Von Neumann sources:
     - $b_1 b_2 \ldots b_i \ldots$ are i.i.d. 0/1 variables and $b_i = 1$ with some probability $p < 1$ then translate
       - 01 $\rightarrow$ 1
       - 10 $\rightarrow$ 0
   - Randomness Extractors produce randomness from general weak sources, many other applications

4. Algorithms: Can Randomness Save Time or Memory?
   - Conjecture - No* (*moderate overheads may still apply)
   - Examples of derandomization:
     - Primality Testing in Polynomial Time
     - Graph Connectivity logarithmic Memory
     - Holdouts: Identity testing, approximation algorithms, ...

5. Expander Graphs
   - Sparse Graphs that are highly connected:
     - $|S| \leq N/10$
     - $|\Gamma(S)| \geq \frac{1}{2} D |S|$
   - Some useful properties:
     - Random walk rapidly mixing
     - Most outgoing edges are “unique” (great for routing)
     - Very fault tolerant
     - Many applications: networking and sorting networks, cryptography, data structures, complexity & proof theory, ...

6. Expander Codes
   - Minimum distance $\geq K$
   - Constant rate and relative distance (can be made very good)
   - Simple Decoding Algorithm in Linear Time & log n parallel phases