Activities

For each operating system feature:

- Test the performance of basic system calls.
  - Example: read a file and measure effective bandwidth from disk.
- Perform tests to clarify incomplete or incorrect documentation.
  - Example: Can one thread read/write another thread’s stack?

Activities

For each operating system feature:

- Perform tests to probe implementation details of the OS.
  - Example: does the priority of a process decay with age?
- Stress test.
  - Example: how many concurrent processes can you spawn?

Skills

- Debugging, especially with library code.
- Experimental design.
- Presentation of technical and quantitative information.
- Reading technical papers.

Experiences

- Realize that all knowledge is tentative.
  (Don’t ask me, don’t trust the book, try it!)
- Learn to deal with that uncertainty.
  (Tame the sophomoric skeptic within.)
- See how new knowledge is created.

Why OS?

- Natural connection to performance evaluation.
- Special challenges:
  - Opaque abstraction.
  - Complex system behavior.

Other good choices: networks, databases, robotics...

Empirical methods in CS

Allen B. Downey
Wellesley College
downey@wellesley.edu
http://rocky.wellesley.edu

“Teaching Experimental Design in an Operating Systems Class”

SIGCSE ’99