

# Operating Systems – Spring 2006

---

Instructor: Dave Hollinger ( *Dave* )

Web Sites:

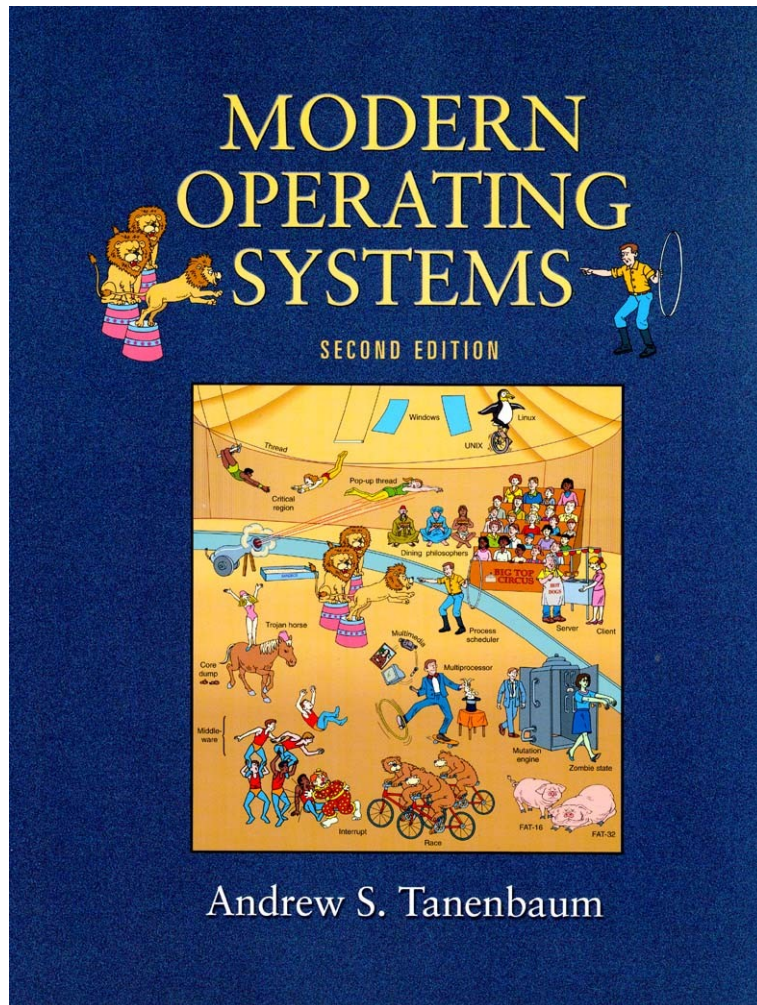
[www.cs.rpi.edu/~hollingd/opsys](http://www.cs.rpi.edu/~hollingd/opsys)

[webct.rpi.edu](http://webct.rpi.edu)

grades

homework dropbox

# TextBook



- Required!
- Great book – easy to read.
- We will also rely on information from other sources (on the www).

# Grades

---

- 55%: Projects/Homework
  - some large programming projects
  - possibly some research projects
- 45%: Tests (three – each worth 15%)
  - dates are on the syllabus.
- No Final Exam
  - the last test will be during the last class meeting.

# Projects/Homework

---

- Some large programming projects.
  - written in C (sometimes C++ is OK).
  - Unix Systems Programming.
- Some small programming projects.
  - use whatever language you want...
- Possibly some *research* projects:
  - design data structures/algorithms
  - look at open source OS code and write a report.

# Unix Programming

---

- Everyone must have a CS account.
- `ssh to freebsd.remote.cs.rpi.edu:`
  - `monica,ashley & mary-kate.cs.rpi.edu`
- Grading will be done on BSD!
- Develop code anywhere you want, but make sure you test on FreeBSD!

# CS Accounts

---

- Remote access to CS department workstations.
- New email address: *username@cs.rpi.edu*
- CS Lab documentation: [www.cs.rpi.edu/lab](http://www.cs.rpi.edu/lab)
  - Instructions on how to access machines, etc.
  - web based email management (spam filtering, forwarding, etc.)
  - Account restrictions, etc.

# CS Workstations

---

- You don't have a machine all to yourself!
  - learn about the `ps` command.
  - make sure you know how to `kill` a process gone bad.
  - pay attention to how much disk space you use!
    - `df` and `du` can be helpful.
  - learn to annoy people!:

```
for user in `who -q | grep -v \#`; do echo "Hello" | write $user; done
```

# Tentative Schedule

---

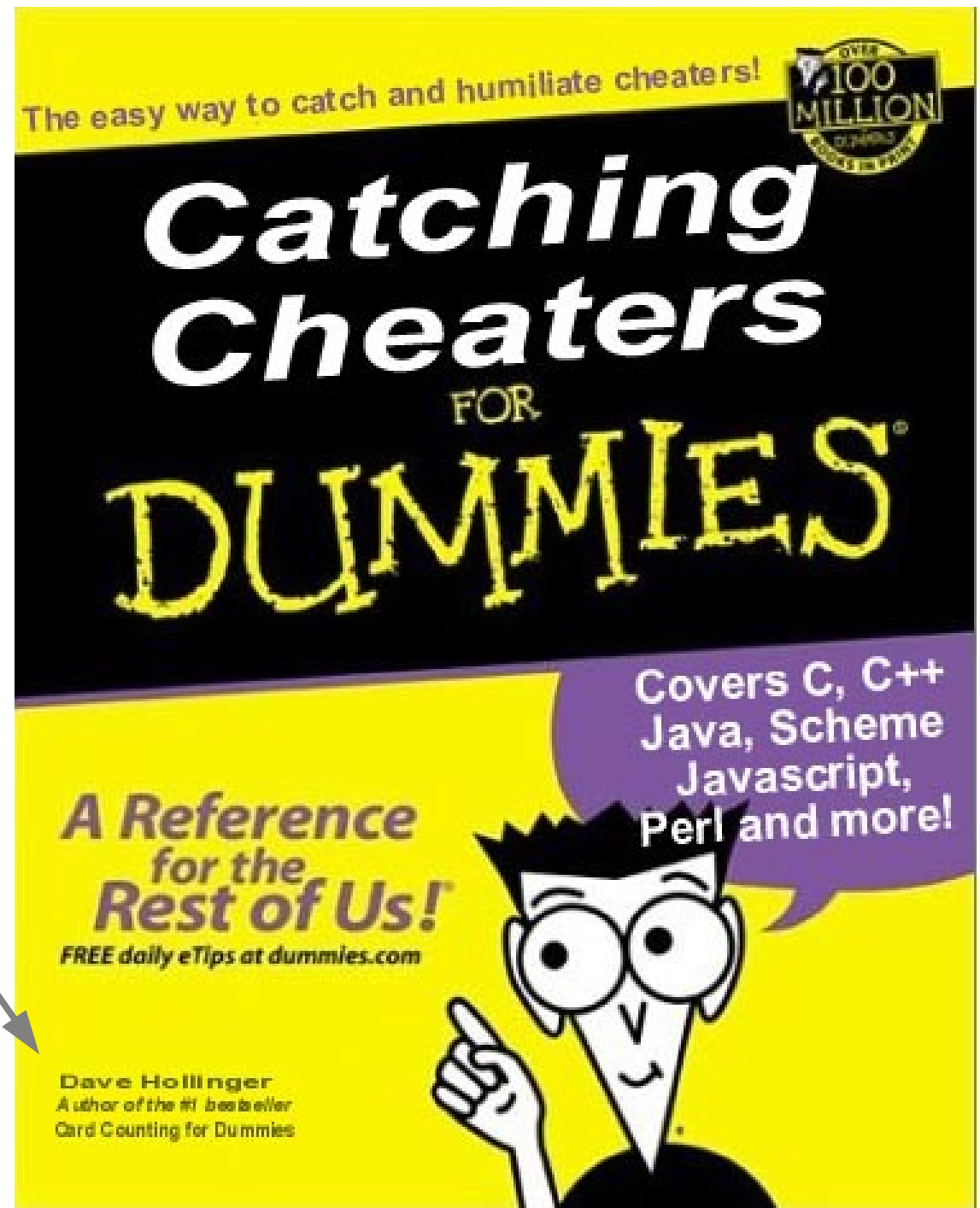
- The Syllabus includes tentative reading schedule, this will certainly change...
  - The test dates will not change!
- We will follow the book fairly closely, although we will also look at other sources of information.
  - The book is great for “the big picture”, but we need to look elsewhere for details...

# Academic Integrity

---

- Any duplicate or near duplicate homework submissions will result in a minimum of a 2 letter grade drop for the final course grade for all students involved and may result in a failure for the entire course.
- For programming projects, you may *discuss* homework with other students (this includes WebCT discussion boards) but sharing of code in any form is not acceptable.
- Looking at another student's code or showing your code to another student is **not** permitted.

Dave is a world renowned cheater catcher and author!



# Programming Project Requirements

---

- Every file must include a comment at the top of the file including your name.
- There must be a file named README that includes your name, a one-line description of each file submitted, instructions for building, etc.
- You must comment your code – if we can't easily follow your code you lose points.

# Course Topics

---

- History
- Systems Programming
- Deadlock
- Memory
- I/O
- File Systems
- Security
- Networking
- Multimedia ?
- Multiple Processors
- Case Studies
  - Linux, FreeBSD
  - Windows