Final Review

Details

- December 15 at 8:00-10:00am, same class
- 120 minutes
- close everything
- comprehensive
  - but biased heavily towards material after midterm

Topics

- core location
- persistence
  - core data
  - cloudkit
  - file systems / sandboxes
  - firebase
- concurrency
  - URLRequests
  - queues
    - ios
    - sync/async ops
    - serial/concurrent
    - dispatch groups
    - barriers
  - semaphores
  - problems
    - deadlock
    - priority inversion
- Memory conflicts / memory safety
  - what is a conflict?
  - how do they arise?
    - under what circumstances?

new stuff

    - ARKit
        - basics of how one creates a scene
        - understand the demo
        - how objects created, specified
        - how objects recognized: horizontal vs vertical
    - Core ML
        - generalities: what kind of learning are we talking about?
- Big things from before
    - how does SwiftUI work?
        - how does communication in SwiftUI work
            - how do you wire in a model?
            - how do you cause the UI to be updated?

There Will Be Coding

- work examples queue / concurrency section / memory safety in detail

Different ways to look at the exam:

- problem type:
    - 95 points short answer
    - 55 points coding
- subjects (very rough estimate)
    - how does swift work: 30 pts
    - persistence: 45 pts
    - concurrency: 50 pts
    - AR/ML: 25 pts

Grades

We currently have 60% of the grades in, i.e. 60 pts.

- Your "overall" column on grades.cs tells you how many points you have accumulated.
- Current stats on grades.cs.umd.edu reflect students who have dropped (I have no control over this). Ignoring them, the grades right now show that ***there will be no***

curve. So for the 60 pts currently on grades.cs (graded so far), this implies:
  - A: 54
  - B: 48
  - C: 42
- Compare this to your current total.
- I expect averages to go down and stdev to go up after the final, so there ***may*** be a curve at the end