Each 6.1800 lecture will come with an outline. You can fill this in during lecture, after lecture, or not at all — it's entirely up to you how you use it. The goal of these outlines is to help you understand the main points that you should be taking away from each lecture. In some cases we will also include examples of things you should be able to do after each lecture.

In the past, these outlines have proved to be an effective tool for studying for the exams. Note that the outlines are ***not exhaustive****; there will be topics and nuances in lecture that aren't captured by the outline.*

Lecture 1: Complexity, Modularity, and Abstraction

The outline for Lecture 1 isn't too detailed. It's an introductory lecture! Things get more technical as we progress.

- Why does complexity make building systems hard?
- How do we mitigate complexity?
- What is modularity, and what does it mean to enforce it?
- How does a client/server model help us enforce modularity?
- What makes a *remote* procedure call different from a procedure call? What problems do RPCs introduce?
- What other things (besides enforcing modularity) do we care about when designing systems?