

*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 03: Virtual Memory**

- What is virtualization?
- When we talk about “memory”, what do we mean? What is the difference between memory and storage?
- Why does an OS use page tables to virtualize memory? What problem do they solve?
- Using page tables, how is a virtual address translated to a physical address?
  - In other words: you should be able to explain the examples we did in class
- What happens if there is not enough memory to store all of our programs' instructions and data?
- What is the kernel's job?
- What are the other bits in page-table entries (PTEs) for? Specifically the P bit, the U/K bit, and the R/W bit. What problems does each of them solve?
  - After this lecture, you should understand what it means when each of those bits is set.
- How is a virtual address translated to a physical address using multilevel page tables?
- Why do multilevel page tables save space compared to “normal” page tables?