Introduction to Abstraction

Jason Gross

Microteaching: Introduction to Abstraction

Microteaching Learning Goals / Intended Learning Outcomes

By the end of the 10 minute session, students will be able to...

- define "abstraction"
- give at least one example of abstraction in mathematics, and partially justify why it's an example of abstraction

1+2. Introduce a specific idea/topic & communicate the goals (mini-lecture/demonstration)

Detailed Notes (What will you actually say or do? What will the students say/do?)

> Today we begin the unit on abstraction. While we won't be using the category theory we learnt in the previous unit today, I plan to slowly integrate it into the future classes on abstraction. I'm both excited and nervous, here, because I've never taught a class on abstraction before, and I'm not aware of any existing textbook that focuses on this topic explicitly. At the same time, it underlies so much of how I see and organize the work that I've done, and so it's in some ways close to my heart. My hope and goal for today is that you'll walk away from this class with the beginnings of a solid understanding of what abstraction and abstraction barriers are, how to recognize them, and how to recognize when operations pierce them and when operations respect abstraction barriers. The category theory, by the way, will come in when we get to talking about what makes abstraction barriers *good* or *bad*.
>
> Without further ado, let's dive in!
>
> In my experience, trying to predict what the teacher is going to say helps with learning, so I'm going to ask you all to come up with a definition of abstraction before I give you my definition. Please spend one minute thinking about how you'd define abstraction, and

then type your definition into the chat; I'd prefer if you wait to send it until I say, so everyone has a chance to come up with their own definition first.

(wait one minute by my watch)

Okay, everyone send your definitions; I'm going to take about 30 seconds to read them all over.

Okay, my turn. As you may or may not recall from reading the syllabus, the essential idea behind abstraction is that, having built a compound object out of multiple components, we can throw away our knowledge of how the compound object was *constructed* without losing anything important. An abstraction is this compound-object whose details we cannot inspect, together with an *interface* for interacting with it. Those of you familiar with programming might recognize this notion of abstraction in abstract data types or APIs—application-programming interfaces.

We'll spend the rest of this class looking at examples of abstraction; I want you to have a really solid grasp of the basics of this concept, as this entire unit depends on it.

Estimated time: 3–4 minutes

Materials required: Zoom chat, a slide deck with the definition of "abstraction"

3. Micro-lecture: demonstrate/provide example(s) related to the idea/topic.

Detailed Notes (What will you actually say? What will the students say/do?)

Our first example may be familiar to any of you who have exposure to formal set theory. However, prior knowledge of set theory is not required, and please do ask any questions that you have.

Let me briefly define sets for those of you who are unfamiliar or need a refresher, so that we're all more on the same page. A *set* is an unordered collection of distinct objects. For example, there is a set containing just the number zero. There's a set called the empty set with no objects in it. There's a set containing just the colors red, green, and blue. There's also a set of all non-negative integers, called the set of natural numbers and written as a blackboard bold capital N. You can't have duplicates in a set, so it makes no sense to talk about a set containing just two copies of the number two, for example.

Set theory is the study of how to base all of mathematics on formal logic and sets. The only primitive objects, pretty much, are sets, and the only objects that these sets can contain are themselves sets. So there's the empty set, the set containing the empty set, and the set containing the set containing the empty set, among others.

Suppose the basic objects we're given are just these sets, and our task is to start talking about the natural numbers, the non-negative integers.

Well, one way to represent these numbers, in fact the standard way in set theory, is to say that "zero" will be represented by the empty set. "One" will be represented by the set containing zero. "Two" will be represented by the set containing zero and one. In general, a number will represented by the set containing all the numbers less than that number. We haven't actually defined "less than" yet, though, so we instead write:

$$0 := \emptyset$$
$$\text{successor}(n) := n \cup \{n\}$$

If you haven't seen it or have forgotten, that cup symbol is read as "union", and means that you take all of the elements of both sets being unioned, and combine them together. The elements of the union of two sets are just the elements in either one—or both—of the two sets.

Estimated time: 2.5–3.5 minutes

Materials required: Slide deck

4. Provide an opportunity for students to apply and integrate new information

a. Support retrieval practice

b. Provide opportunities for students to apply new info

c. Support students' integration of new knowledge and prior knowledge

Detailed Notes (What will you actually say? What will the students say/do? What are some anticipated issues and/or pitfalls? Where might students have particular difficulty? What questions might they have? How will you address these issues/ questions?)

How is this definition of natural numbers an example of abstraction? I want each of you to spend thirty seconds thinking about what makes this an abstraction. Consider writing it down, since humans have limited ability to hold things mentally.

(wait 30 seconds by my watch)

Okay, hold onto those thoughts—I hope you wrote them down!—now can someone share their recollection of the definition of abstraction?

(listen to student, perhaps point out any missing pieces or misconceptions)

(bring up the definition of abstraction in the slide deck)

Now, in order to justify something as an example of abstraction, we need to point at (a) the compound objects we built; (b) the components we built it from; (c) what it means to throw away our knowledge of how the compound object was constructed; and (d) how to justify that we haven't lost anything important. This is a lot. We'll be saving (c) for future examples, and (d) for future classes, but keep these in the back of your mind if you have the bandwidth for it.

Now, I want you all to spend another 30 seconds finding (a) and (b) for this example.

(wait 30 seconds)

Okay, everyone please enter your (a) and (b) into the chat.

(wait 30 seconds)

As *(some/all/none)* of you got, the compound objects we built are 0, successor, and all of the natural numbers. The components we built it from are sets. And note that we don't say anything about sets in naming 0, successor, nor any natural number. We use sets to *define* these objects, but not in naming them, and eventually not even in making use of these objects. This is what I mean when I say that we throw away the details of how these objects are constructed. It's okay if this seems confusing; I hope that the other examples I've prepared will help to resolve some confusion about what exactly I mean by "throwing away the details of how these objects are constructed." And eventually, once I've built up enough examples and machinery, I'll try to give a precise definition.

We're nearly out of time, but I want to note that this interface isn't actually complete. I want to shelve the notion of completeness, for now, though; I think it's worth going through a couple of examples of abstraction before diving more fully into the complete details.

Estimated time: 4–5.25 minutes

Materials required: Slide deck, zoom chat, students with keyboards and a place to write or type

What are some anticipated issues and/or pitfalls? Where might students have particular difficulty? What questions might they have? How will you address these issues/ questions?

- Students might be confused by the definition of natural numbers as sets. (Solution: Spend more time answering confusions, but this is hard in a very time-limited microteaching)
- Students might be unfamiliar with set theory in general. (Solution: Spend more time answering confusions, but this is hard in a very time-limited microteaching)
- Students might struggle to remember the recent definition of abstraction. (Solution: Provide the definition again after letting students struggle to recall it—after all, one of the papers we read says that attempting to recall, even if the recall is unsuccessful, still boosts learning.)
- I might not be giving enough time for students to think and share answers. (Not sure how to solve this one. Give more time? But time is limited in microteaching.)
- It might be hard to hold all four components of abstraction mentally at the same time. (Solution: I explicitly tell them to focus on the first two components for now.)
- Students might be reluctant to participate. (Solution: Establish the norm of participation early and often, and act as if them participating is natural and expected. Maybe tease gently if no-one is participating.)
- Students are likely to be confused about the "throwing away the details" bit, which for me is very natural and intuitive and sharp, but which I imagine will be fuzzy for the students. (Solution: The real class will have two more examples, and we'll be coming back to each of the three examples a second time to dive deeper, and I hope that things will become more clear by the time we look at piercing and respecting abstraction barriers. I also plan to make confusion welcomed here, so that students don't give up because they're confused. Perhaps I should be soliciting questions here instead? But there isn't time in the microteaching session.)

5. A *brief* summary/wrap-up

Detailed Notes: What will you actually say or do to wrap up the session?

> In the actual class, we'd spend the rest of the time covering two more examples, and then going back to all three examples and looking at what it looks like to pierce the abstraction barriers, and what it looks like to respect them. I hope this microteaching session has whet your appetite for the concept of abstraction, for building compound objects out of simpler ones in a way that lets us throw away the details of how we built them. I'd be grateful for any feedback you have, and thanks for your attention and engagement!

Estimated time (plan on (∼30 seconds): 32–42 seconds