

Homework 4

Phantom Types and GADTs

Submit your solutions by emailing them to `tkadur@andrew.cmu.edu` or bringing a printed version to class. Remember not to spend more than an hour on this assignment!

Both phantom types and GADTs are great practical tools for improving the robustness and correctness of your code at compile-time. In class, we talked about using them to enforce unit correctness. This homework will ask you to think about other ways to use these tools. Try to be creative!

1. Explain some reasons why phantom types were not as well-suited as GADTs to the unit correctness example from class.
2. Despite not being perfect for our in-class example, phantom types are useful and solve different problems than GADTs.

Come up with another potential use case for phantom types. You could come up with an entirely novel way to use them, but you could also adapt the example from class to solve a different problem.

3. Describe, in your own words, the difference between GADTs and ADTs. (In other words, what makes GADTs "generalized"?)
4. Read [this blogpost](#) describing a use for GADTs we didn't cover in class - improving performance. The code examples are in OCaml, not SML, but hopefully the main points will still be clear.
5. Come up with another potential use case for GADTs. You could come up with an entirely novel way to use them, but you could also adapt an example from class or the blogpost to solve a different problem.