Date: Sep. 24 2020
To: CS4500 Staff
From: Jake Hansen and Nick Thompson
Subject: TAHBPL Racket language assessment

Overall, both of us have had a very positive experience with Racket in our TAHBPL assignments. We loved the standard library (and its great documentation), and we didn't need to resolve any dependency/installation issues on either of our machines or on the Khoury servers. Both of us have experience with the Racket teaching languages from TAing CS2500, so we were excited to see what we could do with the full power of Racket. RackUnit is fantastic for testing with having the ability to write inline tests and working out of the box with `raco` to run tests from the CLI. We think contracts are really cool, and we even used them with random generation to create dummy JSON data. Although it took us a little while to work out their functionality, we found ports to be elegant and it made reading/writing/testing to tcp/string/standard ports incredibly simple. We didn't end up using threads, but when combined with custodians we thought they had a great API. Our complaints so far have been pretty minimal, our biggest being CLI parsing. We were both slightly let down by `racket/cmdline` and wish it had the ability to give more specific requirements for args themselves, such as contract validation or the ability to transform them from strings. We are uncertain how easy it will be to keep our code simple and maintainable as we scale to a large codebase. The only other issue we have seen is the language ecosystem. DrRacket is fine, but general tooling for other editors (besides emacs) is pretty minimal. We are excited to work with a primarily functional language this semester, and are really happy with our choice of Racket.