Engineering a Component-Based Programming Language

Frias, Diego (School: Oregon Episcopal School)

This project created a programming language with innovative features designed to mitigate the problems with dynamically typed object-oriented languages. Research languages, such as this one, are very important for improving language design in the future, which in turn enhances software all around the world. Primarily, the programming language implemented in this project does not allow for inheritance, while still keeping the intuitiveness of object-based systems. There are many other features that address issues surrounding weakly typed interpreted languages, such as heavily enforced type hints, fine-grained access control, and easier scope management. However, both the language and implementation do not have some features that improve the developer experience, like interfaces and more statically validated code. Despite these issues, the aforementioned language made for this project successfully remedies the problems that it sets out to fix by enforcing component-based programming, lessening untyped code, and giving users the ability to control the footprint of their APIs. Future iterations should look towards implementing a type-checker that has the same purpose as TypeScript and MyPy do for their respective languages.

Awards Won: