Alexa Williams

CPSC 326

HW4 Writeup

The most challenging aspect of this assignment was debugging. I was able to work through getting all of the visit functions written fairly quickly, but it took a very long time to debug them all. I relied heavily on the unit tests in order to figure out what was not working, and I did a lot of printing things out to find exactly where things were going wrong.

The fist unit test I designed is testing that you can create a variable and set its value to another variable that was made above. I wanted to specifically test this with array values to make sure my checker was handling the isArray properly.

The second unit test I designed is testing that you can make a new struct. This test is very simple, but it is very important that users can create a struct and later on make objects using that data structure.

The third unit test I designed is testing that the checker correctly catches a bad binary expression. In this case I wanted it to catch that the user is not able to do a not equals between two array type variables. It happens often that a user means to do a not equals on a specific index in the array, but they forgot to actually grab that index. This test is like a reminded to the user that they cannot do this.

The fourth unit test I designed is testing that a user is not able to call a user defined function that does not yet exist. Often times a user might think they wrote a function, so they use it in their code, but they actually have not written it yet. This reminds the user that they must define that function before trying to use it.

Finally, the fifth unit test I designed is testing that a user is not able to perform subtraction between two boolean values. This is illegal in MyPl, so it is important that the checker catches this issue before the program is run.