

John Mazanec  
Theory of Computing  
Project 4: Combinators  
12/11/2017

*1. Time Spent:* 15 hours

*2. Code Development and Testing:*

To manage the code for this project, I used Github. Although I worked on it alone, I thought it would be a nice project to showcase. I felt it also would be useful to have a version control system. To test the code, I used the test files provided and checked the output against the bottom of those files. I also worked out a few of the reductions on my own. I also created a few test files as well.

*3. Language and Libraries:*

I used Python2.7 and used the sys and copy libraries.

*4. I/O Formatting:*

I use the same input formatting as the test scripts. For the output, I printed each reduction as instructed and also printed the final string after that. Then, I printed the dashed line and the expected output for easy testing and grading.

*5. Executable Combinators:*

My code can handle the S, K, I simple combinators and the W, B and Y double, boolean and recursive combinators.

*6. Key Data Structures and Reduction*

To represent the caf's, I used a simple list of strings. I had functions that would remove unnecessary parentheses and turn strings into more elements in the list(s). To determine which element to reduce, I looped through all of them and kept an index. I worked left to right.

*7. Testing and Functions*

As mentioned above, to test my code, I compared the output against the expected output. I also went through several reductions on my own to make sure it made sense. I wrote five functions. A main function read in the input, an evaluate function recursively reduced the caf, a split\_expr function turned strings into extra list elements, a remove\_p function that removed unnecessary parenthesis, and a p\_out function which formatted the printable reduction recursively.