Approach Lab 4

# First Glance at Functions & Classes

|  |  |
| --- | --- |
| Classes | Description |
| SelfOrderedList | An SOL that sorts based on comparing the amount of times its been used and setting it in greatest to least |
| MTFSelfOrderedList | An SOL that works basically like LRU for the previous lab |
| TransposeSelfOrderedList | An SOL that swaps the current object with the one “before” it |

# Notes Throughout

* 3 *different* classes
* Compares are the amount of compares when searching for the object, not when applying the heuristic
* If the object is not in the list, then append it
* Personal preference: I saw that the outputs were all the same for chars and all the same for strings. So, I made a function that formatted all the data and presented it to the screen. This way main wouldn’t be clogged up the same thing repeatedly.
* For the strings, use find to insert the objects in, NOT add

# Summary/Reflection

This was one of the harder labs for me. I’m not sure if this was because I started it well before it was even discussed or if it was just “the nature of the beast.” But it posed as a good challenge of my knowledge of Linked Lists. I can say that I learned a lot with this one.