1. Zero\_last has an effect on the array despite the return type being void because in the function, we are actively changing the array. We don’t make a copy of the array and return a different array.
2. In my code for zero\_last and maxboth, I treated the parameters as their arrays, not pointers. For example, in maxboth.c, I run a for loop for array x and array y. To check if their values were equal, I just wrote x[i] == y[k]. I’m pretty sure I could keep the same algorithm while changing things slightly to use them as true pointers, instead. For example, I could have used \*(x+1) to step through the array instead of how I did it.
3. When working on fib15 and trying to reverse the array, it kept jumbling the array in a random order instead of reversing it. I wasn’t getting any errors in the terminal, but I decided to use gdb to step through that part of the program anyway. I used display to display the values of start and end through every loop of the while. What I found was the values were resetting every time instead of incrementing down or up, because I had accidentally put the initialization of these variables inside of the while loop. So, to fix this, I put the initialization of the variables outside of the while loop, and my problem was solved!