Project 3 Report

Following the template provided in the project description, my design for my program used that as the foundation, as well as some other things I added for the overall functionality of my program. In addition to the four data structures needed (available, maximum, allocation, need), I also added an array to keep track of the user inputs named “commands”. I also implemented some helper functions, which are named “safety_algorithm”, “request_resource”, release_resource”, and “print_array”. The safety_algorithm function contains the main logic of the program, testing if the request inputted by the user can be satisfied depending on if the machine is in a safe state or not. If the machine is in a safe state, the program outputs the order of processes that is executed and changes the allocation/need array to the appropriate values. The program continues to run until the user decides to terminate the program.

The main struggle I had was implementing the safety algorithm. I needed to find a way to compute all of the processes in a way that the resources can be allocated to each process in an order that prevents deadlock. I used the slides for a guide on how to implement the logic, implementing a step by step check that each process has to go through in order to ensure that it is in a safe state.

Video Link: https://youtu.be/tjIV-0QP0PU