Syed Haroon

CSC 33200 – S

April 15, 2020

**Lab 5 Report**

To avoid the problem's race state, I used semaphores to synchronize the three processes. Previously, either the father or one of the sons may go to the "bank" at the same time to deposit or withdraw funds. Only one process can reach the "bank" or vital section at a time after using the sem.h library to implement semaphores. I call the P() method from sem.h to decrement the semaphore value to -1 before entering the critical segment. This locks the crucial portion, preventing the other two processes from entering and altering the data while the P() process is running. When the process has completed its task, it uses the V() method from sem.h to set the semaphore value to 1. This informs all processes that the essential section is ready for use.